



Opswise Automation Center 5.1.1

User Guide

© 2013 by Stonebranch, Inc. All Rights Reserved.

1. Introduction to Opswise Automation Center	4
1.1 Opswise Automation Center System Overview	5
1.2 Getting Started	8
1.2.1 Setting up Opswise Automation Center	9
1.2.2 Logging In	11
1.2.3 User Interface	13
1.2.4 Navigation Pane	17
1.2.5 Using Lists	22
1.2.6 Using Forms	31
1.2.7 Using Wildcards	37
1.2.8 Naming Tips	38
1.2.9 Business Services	39
1.3 Home Page, Dashboard, and Gauges	41
1.3.1 Using the Home Page	42
1.3.2 Using the Dashboard	45
1.3.3 Creating and Deleting Gauges	48
2. Resources	52
2.1 Resources Overview	53
2.2 Agents and Connectors	54
2.2.1 Linux Unix Agent	62
2.2.2 Windows Agent	66
2.2.3 zOS Agent	70
2.2.4 Indesca Agent	74
2.3 Agent Clusters	77
2.4 Cluster Nodes	81
2.5 Virtual Resources	86
2.6 Script Library	91
2.7 Email Template	95
2.8 Email Connection	97
2.9 Database Connection	99
2.10 SAP Connection	101
2.11 SNMP Manager	103
3. Tasks and Workflows	105
3.1 Creating Tasks	106
3.1.1 Linux Unix Task	109
3.1.2 Windows Task	119
3.1.3 zOS Task	129
3.1.4 Indesca Task	148
3.1.5 SAP Task	158
3.1.6 File Transfer Task	180
3.1.7 Manual Task	199
3.1.8 Sleep Task	206
3.1.9 SQL Task	212
3.1.10 Stored Procedure Task	220
3.1.11 Email Task	230
3.1.12 Task Monitor Task	237
3.1.13 File Monitor Task	245
3.1.14 FTP File Monitor Task	252
3.1.15 System Monitor Task	262
3.1.16 Copying Tasks	268
3.1.17 Creating Email Notifications	271
3.1.18 Creating SNMP Notifications	277
3.1.19 Setting Abort Actions	281
3.1.20 Setting Variable Actions	285
3.1.21 Setting System Operations	286
3.1.22 Setting Mutually Exclusive Tasks	291
3.1.23 Creating Step Conditions	293
3.1.24 Creating Notes	299
3.2 Manually Running and Controlling Tasks	302
3.3 Creating and Maintaining Workflows	321
3.4 Monitoring Task Activity	345
3.4.1 Monitoring Activity from the Activity Screen	346
3.4.2 Monitoring Activity from the Task Instances Screen	352
3.4.3 Monitoring Activity History	358
3.4.4 Monitoring Workflows	360
3.4.5 Viewing Task Instances for a Specific Task	366
3.4.6 Displaying Task Instance Status	370
3.4.7 Retrieving Output	372
4. Variables and Functions	376
4.1 Variables and Functions Overview	377
4.2 User-Defined Variables	378
4.3 Built-In Variables	382
4.4 Launching With Variables	392
4.5 Trigger With Variables	393
4.6 Setting Variables within a Workflow	394

4.7 Listing and Setting Variables from the Command Line	398
4.8 Functions	399
5. Triggers and Calendars	415
5.1 Creating Triggers	416
5.1.1 Cron Trigger	421
5.1.2 Time Trigger	427
5.1.3 Manual Trigger	434
5.1.4 Temporary Trigger	437
5.1.5 File Monitor Trigger	440
5.1.6 Task Monitor Trigger	445
5.1.7 Enabling and Disabling Triggers	449
5.1.8 Copying Triggers	450
5.1.9 Triggering with Variables	453
5.1.10 Displaying Trigger Forecast Information	456
5.2 Calendars	466
5.2.1 Creating Custom Days	467
5.2.2 Creating Calendars	470
5.2.3 Copying Calendars	473
6. Application Monitoring and Control	476
6.1 Application Monitoring and Control Overview	477
6.2 Applications	480
6.3 Application Control Tasks	484
6.4 Application Monitor Trigger	490
7. Reports	494
7.1 Opwise Automation Center Reports	495
8. Managing Records	517
8.1 Overview of Opwise Records Management	518
8.2 Record Versioning	519
8.3 Exporting and Importing Records	522
8.4 Bundling and Promoting Records	523
8.5 Backing Up and Purging Data	535

Introduction to Opwise Automation Center



Opwise Automation Center



Getting Started

System Overview / Components

Setting up Opwise Aut



Home Page, Dashboard, and Gauges

Logging In

User Interface

Overview

Navigation Pane

Using the Home Page

Using Lists

Using the Dashboard

Using Forms

Creating and Deleting Gauges

Using Wildcards

Naming Tips

Business Services



The information on these pages also is located in the [Opwise Automation Center 5.1.1 User Guide.pdf](#).

Opwise Automation Center System Overview

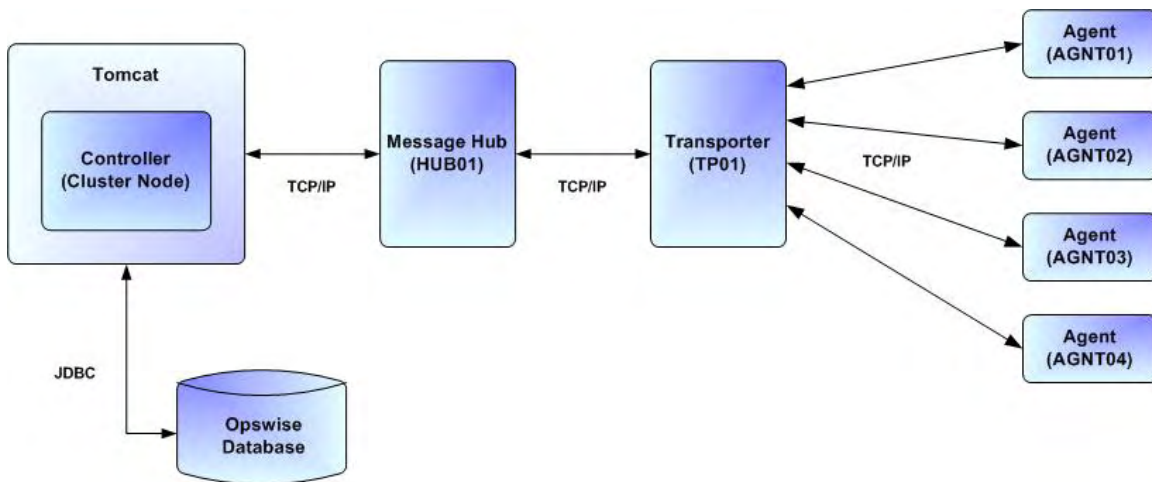
- Opwise Automation Center
- Controller
- Application Container
- Database
- Outboard
 - Message Hub (MsgHub)
 - Transporter (Transport)
 - Command Line Interface (CLI)
- Agent
- Opwise Automation Center Directory Structure
 - Controller Directories
 - Outboard Components (Message Hub, Transporter) Directories
 - Agent Directories

Opwise Automation Center

Opwise Automation Center consists of the following basic components:

- Controller (cluster node)
- Application Container
- Database
- Outboard
 - Message Hub
 - Transporter
 - Command Line Interface (CLI)
- Multiple remote Agents

The following diagram illustrates a typical configuration of Opwise Automation Center. Apache Tomcat is the application container, with the Controller running inside Tomcat; a DBMS to contain the database; the Outboard components, which include the Message Hub (HUB01) and Transporter (TP01); and one or more Agents installed on various platforms.



Each component is described in more detail, below.

Controller

The Controller (Cluster Node) provides the business logic of Opwise. The Controller is a Java web application running in a Tomcat web container. This central processing server provides components that present the user interface, handle the scheduling logic, process all messages to and from the agents, and synchronize much of the High Availability operation of the product.

For information on how multiple Controllers operate in a high availability environment, see [High Availability](#).

Application Container

The application container is third-party software that serves as a container for the Controller. Opwise Automation Center uses Apache Tomcat as the application container.

Database

The database management component supports SQL queries to a set of tables in the Opwise database.

The following databases are supported:

- Oracle 10g, 11g
- MS SQL Server 2005, 2008
- MySQL 5.1, 5.5

Outboard

The Opwise Outboard comprises three components:

- Message Hub
- Transporter
- Command Line Interface (CLI)

Message Hub (MsgHub)

The Message Hub relays messages from the outbound message queue from the Controller to the Agents, and writes messages to the inbound message queue from the Agents to the Controller.

The Message Hub communicates with the Agents using a "publish | subscribe" communications server known as the Transporter. Those messages are transformed to and from serialized message structures and moved to and from the Agents via the messaging layer.

Transporter (Transport)

The Transporter serves as the enterprise bus for the Unified Server Processes as well as the TCP/IP transport layer for the Outboard components.

The Message Hub and Agents are considered clients of the Transporter. The Transporter serves to abstract a group communications protocol referred to as "publish | subscribe." Clients subscribe to message queues to receive message traffic, and send data to subscribers by publishing messages to the Transporter. Within Opwise, the Outboard component names are normally assigned by the Controller when the clients initially register their availability at process start-up; however, the user also can specify the names (see [Agents and Connectors](#)).

Command Line Interface (CLI)

The Opwise Command Line Interface (CLI) is implemented as a set of commands that perform specific actions in an Opwise Controller. The CLI commands can execute on any system that has TCP/IP connectivity to the Opwise Transporter. The results of the action are printed to the CLI commands standard output.

Agent

The Agent, installed on each remote host system, launches and monitors applications or scripts, and returns status information and output when requested.

The Agent also performs event-oriented tasks such as file monitoring. When the Agent starts, it establishes a connection with the message transporter and registers with the Controller. When an Agent-based task is launched by the Controller, the Controller sends the Agent a set of messages that instructs the Agent to perform the requested services (see [Agents and Connectors](#)).

Opwise Automation Center Directory Structure

Shown below is the directory structure of Opwise Automation Center: Controller, Outboard, and Agent.

Controller Directories

```
/home//tomcat/  
/home//tomcat/bin  
/home//tomcat/conf  
/home//tomcat/lib  
/home//tomcat/logs  
/home//tomcat/opswise_logs  
/home//tomcat/opswise_export  
/home//tomcat/temp  
/home//tomcat/webapps  
/home//tomcat/webapps/opswise  
/home//tomcat/webapps/opswise/help  
/home//tomcat/webapps/opswise/htmlarea  
/home//tomcat/webapps/opswise/images  
/home//tomcat/webapps/opswise/META-INF  
/home//tomcat/webapps/opswise/portlet  
/home//tomcat/webapps/opswise/scripts  
/home//tomcat/webapps/opswise/styles  
/home//tomcat/webapps/opswise/WEB-INF  
/home//tomcat/webapps/opswise/WEB-INF/apps  
/home//tomcat/webapps/opswise/WEB-INF/db.index  
/home//tomcat/webapps/opswise/WEB-INF/ui.jtemplates  
/home//tomcat/webapps/opswise/WEB-INF/dict  
/home//tomcat/webapps/opswise/WEB-INF/plugins  
/home//tomcat/webapps/opswise/WEB-INF/sys.scripts  
/home//tomcat/webapps/opswise/WEB-INF/update  
/home//tomcat/webapps/opswise/WEB-INF/import.templates  
/home//tomcat/webapps/opswise/WEB-INF/lib  
/home//tomcat/webapps/opswise/WEB-INF/properties  
/home//tomcat/webapps/opswise/WEB-INF/ui.jforms  
/home//tomcat/webapps/opswise/graph  
/home//tomcat/webapps/opswise/hta  
/home//tomcat/webapps/opswise/icons  
/home//tomcat/webapps/opswise/mobile  
/home//tomcat/webapps/opswise/portal  
/home//tomcat/webapps/opswise/temp  
/home//tomcat/work
```

Outboard Components (Message Hub, Transporter) Directories

```
/opswise  
/opswise/bin  
/opswise/cache  
/opswise/etc  
/opswise/logs  
/opswise/var
```

Agent Directories

- Linux/Unix directory structure
- Windows directory structure
- z/OS directory structure

Getting Started

The following information is available for getting started with Opswise Automation Center 5.1.1:

- [Setting up Opswise Automation Center](#)
- [Logging In](#)
- [User Interface](#)
- [Navigation Pane](#)
- [Using Lists](#)
- [Using Forms](#)
- [Using Wildcards](#)
- [Naming Tips](#)
- [Business Services](#)

Setting up Opwise Automation Center

The following table provides a guideline for setting up Opwise Automation Center. It is a checklist of features and functions, including links to the detailed information and instructions on this website.

Perform Pre-installation Checks	Verify that the ports configured for all Opwise Automation Center components and prerequisites are not blocked by a firewall, and determine the space requirements for your Opwise Automation Center components and database.
Install Opwise Automation Center	If you are a new Opwise customer, you first must download and install the appropriate Opwise Automation Center software for your platform. See Installing Opwise Automation Center for help in getting Opwise Automation Center installed and verified.
Configure Opwise Automation Center	After installation, check the ports configuration of your Opwise components and the Opwise Automation Center properties . If you will be running Opwise Automation Center in a High Availability environment, you must configure Opwise Automation Center components for High Availability.
Log in to Opwise Automation Center	Once Opwise is installed, we recommend that you log in and familiarize yourself with the basic features of the user interface . See Navigation Pane for a description of each entry on the user interface navigation pane, from where you select all Opwise functions. Within Opwise, you will create all your records (for example: users, tasks, and triggers) by entering information into forms . The records are then displayed in lists of each type, which you can sort and filter and perform a wide variety of other functions.
Set up Security	Use the Opwise Automation Center Security module to create users and user groups and assign them roles and permissions . You can also define credentials that are used by Opwise to log in to remote machines. You can create Business Services that represent your organization and assign Opwise records to them, and assign permission only to users and/or user groups that belong to a specific Business Services. A complete audit history of all Opwise Automation Center activity also is available for regulatory compliance.
Define Resources	Define the types of resources that you will need in your operational database. If you have installed Opwise Agents on any machines, records for each of them are automatically created when they connect to the Opwise Controller . You also may need to define one or more resources such as email , database , or SAP connections. You also can create status-based notifications for these resources. You can set up a throttling scheme for your machines using Virtual Resources and create a library of scripts that you can execute on remote machines.
Create Tasks and Workflows	Once you have your resources in place, you can begin creating tasks . Supported task types are Workflows , Linux/Unix , Windows , z/OS , Indesca , File Transfer , Manual , Sleep , SQL , Stored Procedure , Email , Task Monitor , File Monitor , FTP File Monitor , System Monitor , SAP , and Application Control .
Create Task Triggers	To run your tasks outside of workflows, you can create task triggers , which define events, conditions, or dates/times that the tasks will run. Trigger types include Cron , Time , Temporary , Manual , File Monitor , Task Monitor , and Application Monitor . You may also need to set up one or more customized calendars that reflect your fiscal year and holiday schedules.
Create and Run Reports	Opwise provides a set of pre-defined reports and lets you create your own reports. It also lets you create a gauge from a report that you can add to your Home Page and/or Dashboard .
Monitor Operations	You will monitor your automated operations from the Activity screen , which you can customize using filters .
Manually Run and Control Tasks	You also may need to manually run and control tasks, either from the user interface or from the command line .
Home Page, Dashboard, and Gauges	You can customize the Home Page and Dashboard of your Opwise Automation Center system for each user. You can also define your own set of gauges , which are live reports that you display on your Home Page and/or Dashboard.
Monitor and Control Applications	Opwise Automation Center lets you to monitor and control all of the applications that you may have running in your entire network.
Command Line Interface	You also can monitor and control your operations and perform basic administrative functions from the command line .

Manage Data and Audit Records	Comprehensive utilities are provided that allow you to manage your Opswise records. You can view, restore, and purge old versions of Opswise records, bundle and promote records from one Opswise Automation Center server to another, and export and import records when performing an Opswise Automation Center system upgrade. You also can configure the automatic backup and purge of Opswise Automation Center data.
Help and Support	This Documentation website provides information to help you install, configure, and use Opswise Automation Center; see Help for information documentation layout and usage. Step-by-step tutorials and Instructional Videos are available for many of the features and functions described here. Troubleshooting provides a description of error messages that you could encounter, as well as potential problems and solutions. Technical Support for critical and non-critical problems is always available.

Logging In

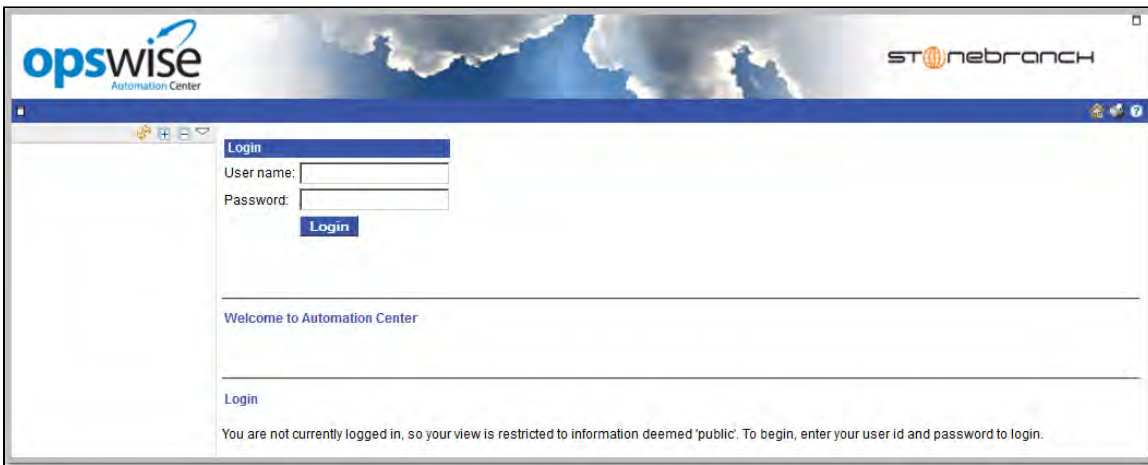
- Opswise Automation Center Login Screen
- User Name
- Password
- Password Expiration
- User Lockout
- User Restriction

Opswise Automation Center Login Screen

The Opswise Automation Center Login screen displays automatically when you bring up the Opswise system and browse to its URL.

The default URL is: **machinename:8080/opswise**

For example, if you installed Opswise locally, the URL is **localhost:8080/opswise**



User Name

The default login User name is **ops.admin**.

Password

For your initial login to Opswise, no password is required; Opswise prompts you to create a password.

Password Expiration

If the **Password Expiration Enabled** Opswise system property has been set to **true**, and you reach the maximum number of days that a user password can remain unchanged, as specified by the **Password Expiration in Days** Opswise system property, the following message displays when you enter your password at the **Password** prompt:

The system administrator requires you to change your password. Your password has expired.

You must then enter a new password.



Note

Password expiration is not applicable to LDAP authenticated users.

User Lockout

If the [Lock Account After Maximum Failed Login Attempts](#) Opswise system property has been set to **true**, and you reach the maximum number of successive failed login attempts that is allowed, as specified by the [Maximum Failed Login Attempts](#) Opswise system property, your user account in Opswise Automation Center will be locked.

(Whenever [Lock Account After Maximum Failed Login Attempts](#) is reset from **false** to **true**, the current number of failed login attempts for all users is reset to 0.)

If you attempt to log in with a locked account, the following message displays:

```
User name or password invalid.
```

To unlock a locked account, your Opswise system administrator must uncheck the **Locked out** field on the [User Definition](#) screen for that user account.

User Restriction

You can be restricted from logging in to the Opswise Automation Center user interface either of two ways:

1. The system level default for web browser access, specified by the [System Default Web Browser Access](#) Opswise system property, has been set to **No**, and the **Web Browser access** field on the [User Definition](#) screen for your user account is set to **-- System Default --**.
2. The **Web Browser access** field is set to **No**, which overrides the [System Default Web Browser Access](#) value (**Yes** or **No**).

If either restriction is in place, the following error message will display when you enter your user name at the **User name** prompt:

```
User <your user name> not permitted to login through the web browser. Please check with your administrator.
```

To remove the restriction, the system administrator must either:

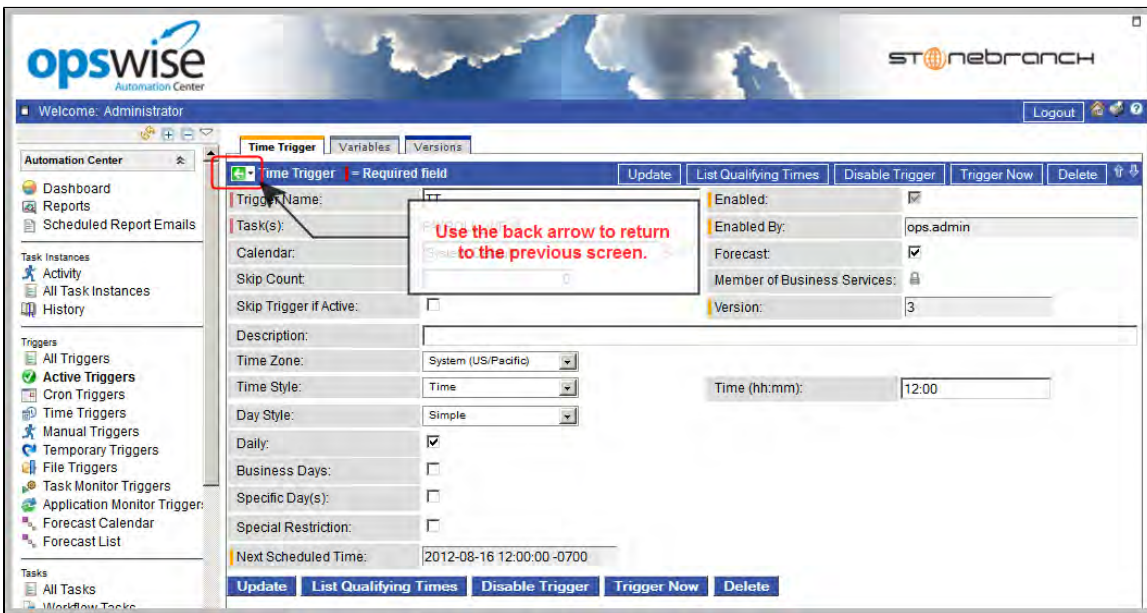
- Set the [System Default Web Browser Access](#) property to **Yes** and set the **Web Browser access** field on the [User Definition](#) screen for your user account to **-- System Default --**.
- Set the **Web Browser access** field on the [User Definition](#) screen for your user account to **Yes**.

User Interface

- Back Arrow
- Drop-Down Menus
- Context-Sensitive Menus
- Logout/Login Button
- Home Button
- Print Button
- Help Button
- Using Lists and Forms
- User Interface Properties
 - Changing Items Per Page Display for Lists

Back Arrow

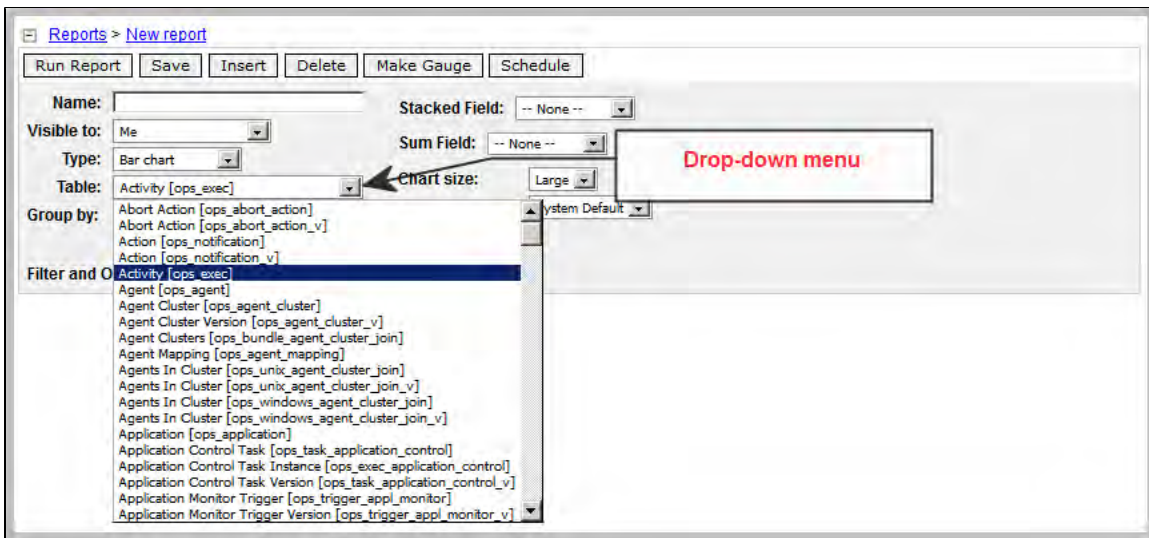
The back arrow, shown below, allows you to back up to the previously displayed screen. If you make changes to a screen and then click the back arrow, the changes are discarded.



Drop-Down Menus

Drop-down menus appear frequently in the user interface. For example, when you are creating a report, a drop-down menu displays a list of tables available for your report. Drop-down menus provide several methods that help you locate the record you are looking for:

- Use the up/down arrows and the scroll box to scroll through the list.
- Type a letter to jump to the first record in the list beginning with that letter. For example, type R to jump to the first record beginning with R. You then can continue pressing R to scroll through only those records in the list beginning with R.

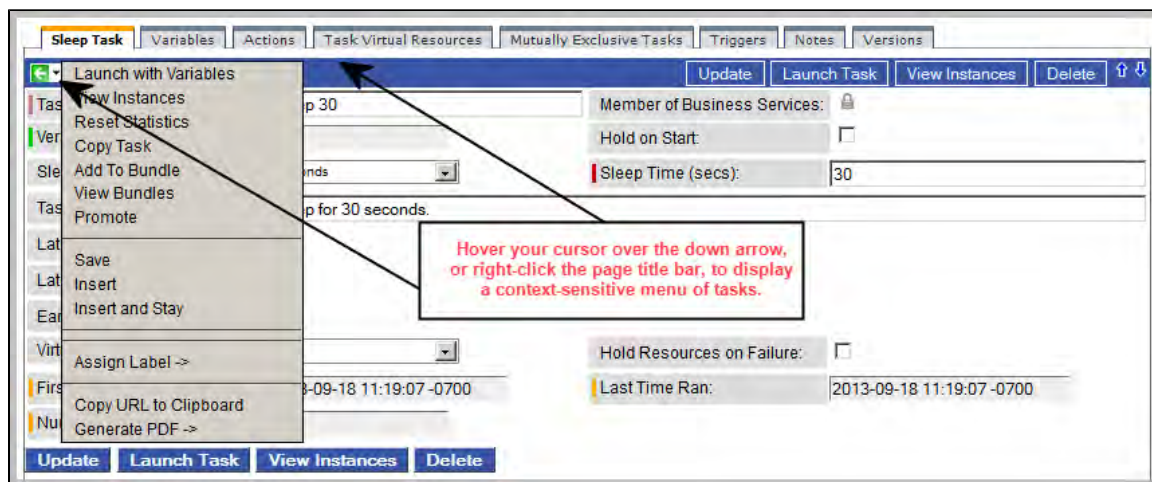


Context-Sensitive Menus

You can access context-sensitive menus from lists and record definition screens.

- From a list:
 - Right-click the blue title bar or column header to display a list of actions available for the entire list.
 - Right-click a record on the list to display a list of actions available for that record.
- From a record definition, either:
 - Hover your cursor over the down arrow on the left side of the blue title bar.
 - Right-click the blue title bar. (If a browser menu displays - back, forward, reload, and so on - right-click lower on the title bar.)

The contents of the menu varies, depending on the context and your user permissions. Shown below is a sample context-sensitive menu.



The following list identifies basic right-click menu options that appear in most contexts:

- **Save**
- **Insert**
- **Insert and Stay**

(See Saving, Updating, Deleting, and Copying Records.)

Logout/Login Button


To login and logout of Opwise, click the **Login** or **Logout** button, respectively.



Home Button

To return to your home page, click the Home  icon.

Print Button

To print the contents of the current page, click the Print  icon.

Help Button

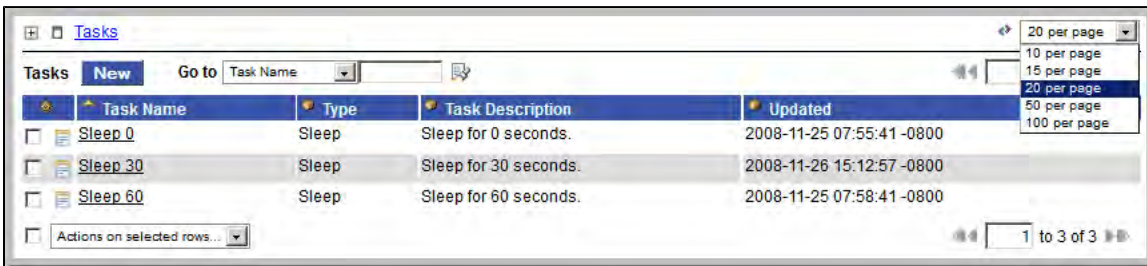
To display help, click the Help  icon.

Using Lists and Forms

A **list** is a display of records of the same type, such as a list of tasks, calendars, or users. A **form** is a screen used to enter and save a record, such as a Task definition screen.

User Interface Properties

For **lists**, Opwise provides a user interface property that lets you select, from the drop-down menu in the top right corner of any list page, the number of records that display per page.



Note

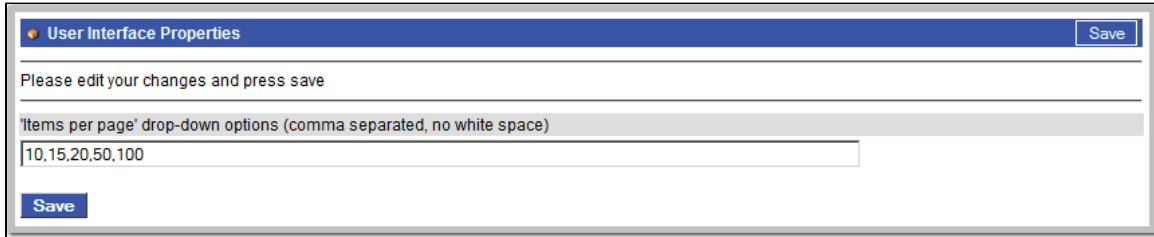
This drop-down menu is available only for lists that display when you select an item from the **Navigation Pane**.

For lists that display when you click a **tab** for an individual record, the menu is not available; 20 is the maximum number of records that can display on a tab page.

Changing Items Per Page Display for Lists

To change the selections in the **Items per page** drop-down menu:

Step 1 From the navigation pane, select **Automation Center Administration > Configuration > UI Properties**. The User Interface Properties screen displays.



Step 2 Enter a comma-separated list of selections for the number of items that can be displayed on a page.



Warning

To maintain optimal performance, we discourage the display of more than 200 items per page.

Step 3 Click the Save button.


Navigation Pane

The following table provides a quick reference and links for each item on the Opwise navigation pane.

Section	Menu Option	Description and Links
Automation Center	Dashboard	A dashboard allows you to set up a display of information that users commonly refer to throughout the day. The information is extracted from the database by way of one or more gauges . The dashboard is accessed by clicking Automation Center > Dashboard from the navigation pane. See Using the Dashboard .
	Reports	The Opwise installation includes a number of predefined reports. You can also create, save, and run your own reports as needed. The Activity screen also uses reports created using this feature to define what task instances are displayed. When you create a report for the Activity screen, you select records only from a specific table called the Activity table. When you save the Activity report, it appears automatically in the drop down-menu on the Activity screen. For normal reports, the report appears on the Reports menu when you save it. See Opwise Automation Center Reports .
	Scheduled Report Emails	This report scheduler allows you to set up a report to be run and distributed to an emailing list on specific dates and times. See Scheduling Automatic Report Distribution .
Automation Center > Task Instances	Activity	The Activity screen is a real-time display of task status and the Opwise Automation Center's central console of activity. It displays all or a selected group of task instances, as controlled by the Activity Report selected in the Activity screen drop-down menu. The selected report also defines what columns are displayed. See Monitoring Activity from the Activity Screen .
	Task Instances	Task Instances displays the same information as the Activity screen, but only for task instances for which there has been a status change or a modification to the task instance record within the last 7 days (an Updated on Last 7 Days filter has been pre-selected for this display). Also, unlike the Activity screen, the display is not automatically refreshed. This screen allows you to issue commands against multiple tasks and provides more extensive filtering capabilities. Task Instances also allows you to view details about workflow instances – information that is not available from the Activity screen. See Monitoring Activity from the Task Instances Screen .
	History	The History screen provides an historical display of all completed task activity. Only tasks with a status in an "end state" (SUCCESS, FINISHED, FAILED, CANCELLED, START FAILURE, SKIPPED) display on the History screen. This allows you to track information about a specific task instance, including multiple runs. For example, Task A may have failed and was then re-run by a user. This task instance will display twice on the History screen, first the time that it ran and failed and again for the time it was re-run to success. See Monitoring Activity History .
Automation Center > Triggers	All Triggers	Displays all triggers. A trigger specifies times or events, or both, that trigger one or more tasks. When each trigger is satisfied, Opwise Automation Center loads the task(s) into the schedule (creates a task instance for each task) and runs it. If a task has multiple triggers, Opwise Automation Center creates and runs a task instance each time a trigger is satisfied. See Creating Triggers .
	Active Triggers	Displays all enabled triggers. Opwise Automation Center only processes triggers that are flagged as Enabled. For tracking and compliance purposes, you must manually enable and disable triggers using the Enable Trigger and Disable Trigger buttons or menu options, or the command to enable and disable triggers. This process saves an audit record detailing the event. See Enabling and Disabling Triggers .
	Cron Triggers	Displays all Cron triggers. The Cron trigger uses standard Cron syntax. Once the Cron trigger is entered into the Opwise system, Opwise Automation Center interprets it and processes it as it would any other trigger. The trigger fires when the current date and time match the all the values specified in the Minutes, Hours, Day of Month, Month, and Day of Week fields. See Cron Trigger .
	Time Triggers	Displays all Time triggers. The Time trigger allows you to specify dates and times at which a task will be triggered. You can define specific dates and times, such as "March 15 at 12:00 a.m.," or a series, such as "every hour of every Friday," or a mixture, such as "9 a.m. every Monday." You can specify simple date and time selection parameters, such as "every weekday at 12:00," or create more complex formulas such as "every 3 hours on the last business day of the year." See Time Trigger .
	Manual Triggers	Displays all Manual triggers. The Manual trigger allows you to launch a task(s) immediately, while overriding one or more user-defined variables used by the task. You will use this trigger if you want to manually launch a task but cannot use the Launch Task or Trigger Now buttons because you need to override one or more variables. See Manual Trigger .
	Temporary Triggers	Displays all Temporary triggers. The Temporary trigger allows you to set up a one-time trigger for a task, based on a single date and time. You will use this trigger if you want to set up a task to run once at some time in the future. See Temporary Trigger .

	File Triggers	Displays all File triggers. The File Monitor trigger allows you to trigger a task based on the creation, deletion, change, existence or non-existence of a file on a particular machine. The trigger works by executing a File Monitor task, which specifies the remote machine (Windows, Linux, Unix, z/OS) and what kind of file event triggers the new task to run (create, delete, and so on). When the File Monitor task notifies the trigger that the File Monitor event has occurred, the trigger then runs the specified task(s). See File Trigger .
	Task Monitor Triggers	Displays all Task Monitor triggers. The Task Monitor Trigger allows you to trigger one or more tasks based on the conditions specified in an associated Task Monitor task. Each Task Monitor trigger is associated with a single Task Monitor task that monitors any number of running tasks for the specified conditions. When you enable this trigger, its associated Task Monitor task launches. When you disable this trigger, its associated Task Monitor task finishes. You can trigger any number of tasks when the conditions in the associated Task Monitor are satisfied. See Task Monitor Trigger .
	Application Monitor Triggers	Displays all Application Monitor triggers. The Application Monitor Trigger allows you to trigger tasks based on the status of one or more Application resources . See Application Monitor Trigger .
	Forecast Calendar	For Time, Temporary and Cron triggers, displays all scheduled instances for the next N days. The number (N) of days displayed in the forecast is specified using an Opwise property. See Forecast Calendar .
	Forecast List	Displays information about every task in the Forecast Calendar, including tasks within a workflow launched by a trigger. See Forecast List .
Automation Center > Tasks	All Tasks	Displays all defined tasks. An Opwise task executes some process on a machine. The process might be resident on the machine (agent-based process) or the task itself might embed the process, such as a File Monitor Task. See Tasks List .
	Workflow Tasks	Displays Workflow tasks, which are created using the Workflow definition tool. This is a graphical tool that allows you to select tasks, position them within a workflow and specify the dependency relationships between them. See Creating Workflows .
	Linux/Unix Tasks	Displays Linux/Unix tasks. These allow you to run platform-specific applications on Linux/Unix machines. See Linux/Unix Task .
	Windows Tasks	Displays Windows tasks. These allow you to run platform-specific applications on Windows machines. See Windows Task .
	z/OS Tasks	Displays z/OS tasks. These allow you to run platform-specific applications on z/OS machines. See z/OS Task .
	Indesca Tasks	Displays Indesca tasks. These allow you to run platform-specific applications on a machine where Indesca is running. See Indesca Task .
	SAP Tasks	Displays SAP tasks. These allow you to execute Stonebranch USAP Commands against an instance of SAP. See SAP Task .
	File Transfer Tasks	Displays File Transfer tasks. The File Transfer task allows you to execute an FTP, SFTP, or Infitran command on a remote machine where an FTP or Infitran server is running. To run a File Transfer task, you need an Opwise Automation Center Linux/Unix, z/OS, Windows or Indesca agent to communicate with the file transfer (FTP or Infitran) server. The agent can but does not have to be running on the same machine as the file transfer server. See File Transfer Task .
	Manual Tasks	Displays Manual tasks. Manual tasks are used to create a pause in a workflow during which the user must take some action. See Manual Task .
	Sleep Tasks	Displays Sleep tasks. The Sleep task allows you to execute a sleep command for a specified number of seconds, a different type of duration such as minutes or days, or until a specific time. This task is helpful, for example, if you need to impose a pause of a specific duration in the processing of a workflow. See Sleep Task .
	SQL Tasks	Displays SQL tasks. The SQL task allows you to execute an SQL statement against a database. To run an SQL task, you first need to create a Database Connection , which defines the information needed to locate and access the database. See SQL Task .
	Stored Procedure Tasks	Displays Stored Procedure tasks. The Stored Procedure task allows you to execute a stored procedure against a database. To run a Stored Procedure task, you first need to create a Database Connection , which defines the information needed to locate and access the database. See Stored Procedure Task .
	Email Tasks	Displays Email tasks. The Email task allows you to create and send emails. In order to execute Email tasks, you first need to define an Email Connection , which defines the server information and other pertinent information. See Email Task .

	Task Monitors	Displays Task Monitor tasks. The Task Monitor task monitors another task or tasks for one or more specific statuses. This task is used in conjunction with a Task Monitor trigger . The Task Monitor task specifies the name of the task or tasks being monitored and the conditions being monitored for. The associated Task Monitor trigger specifies what task or tasks will launch when the conditions are met. See Task Monitor Task .
	File Monitors	Displays File Monitor tasks. The File Monitor task allows you to monitor a specific remote machine's file system for the creation, deletion, change, existence, or non-existence of one or more files at a specific location. See File Monitor task .
	FTP File Monitors	Displays FTP File Monitor tasks. The FTP File Monitor task allows you to monitor for a file on a remote machine where an FTP server is running. The FTP File Monitor connects to the FTP server rather than the machine's file system to monitor for files. The FTP File Monitor can be used only within a workflow; you cannot run a FTP File Monitor task based on a trigger. To run an FTP File Monitor task, you need a Opwise Automation Center Linux/Unix, z/OS, or Windows agent to communicate with the FTP server. The agent can but does not have to be running on the same machine as the FTP server. See FTP File Monitor Task .
	System Monitors	Displays System Monitor tasks. The System Monitor task allows you to monitor a specific remote machine and check for disk space. See System Monitor Task .
	Application Control Tasks	Displays Application Control tasks. The Application Control task allows you to execute a start, stop, or query command against an application in the Opwise network. See Application Control Task .
Automation Center	Calendars	Calendars define business days, holidays, and other special days. They are used in conjunction with triggers to define when tasks are run. See Calendars .
	Custom Days	Custom days definition defines a single one-time date, a repeating date, or a list of dates. Custom days are attached to calendars. See Creating Custom Days .
	Variables	Variables is used to define global variables, which is a type of user-defined variable. See User-Defined Variables .
	Business Services	Business Services allow you to organize your data into business groups. You do so by creating Business Services that represent your organization and assigning Opwise records, such as tasks and resources, to one or more Business Services. You can then sort and filter screens based on the Business Services, as well as generate reports. You can also take advantage of Business Services when you set up security by assigning roles and permission only to specific Business Services. See Business Services .
	Credentials	Credentials are defined by the user and used by Opwise to log in to remote machines. See Credentials .
Automation Center > Support Links	Support Portal	Links to the support page on the Stonebranch website.
	Video Classroom	Links to the Opwise Video Classroom , which provides demos of Opwise features.
Automation Center Resources	All Agents	Displays a list of all Opwise agents. When you start an agent for the first time, Opwise automatically creates a database record containing details about the agent. This option displays a list of all agents that have connected to this Opwise server. See Displaying Agent Information .
	Linux/Unix Agents	Displays a list of Linux/Unix Agents. See Linux/Unix Agent .
	Linux/Unix Agent Clusters	Agent Clusters allow you to configure a cluster (or group) of agents and a selection method, which you can then specify in a task. When you specify an Agent Cluster in a task, Opwise Automation Center selects the best agent from the cluster, based on the selection method specified. If you specify both an agent and an agent cluster in a task, Opwise Automation Center first attempts to run the task on the agent; if the agent is unavailable, Opwise selects the best agent from the agent cluster. See Agent Clusters .
	Windows Agents	Displays a list of Windows Agents. See Windows Agent .
	Windows Agent Clusters	Agent Clusters allow you to configure a cluster (or group) of agents and a selection method, which you can then specify in a task. When you specify an Agent Cluster in a task, Opwise Automation Center selects the best agent from the cluster, based on the selection method specified. If you specify both an agent and an agent cluster in a task, Opwise Automation Center first attempts to run the task on the agent; if the agent is unavailable, Opwise selects the best agent from the agent cluster. See Agent Clusters .
	z/OS Agents	Displays a list of z/OS Agents. See z/OS Agent .
	Indesca/Infitrans Agents	Displays a list of Indesca Agents. See Indesca Agent .

	Connectors	Displays all Opswise connectors. When you start an agent for the first time, Opswise automatically creates database records containing details about the agent and its related components, the hub and transporter, which are also called connectors. This option displays a list of all connector records associated with agents that have connected to this Opswise server. See Displaying Connector Information .
	Cluster Nodes	Cluster Node (Controller) is an Opswise server. This option displays a list of all registered Opswise server nodes. In a high availability configuration, you will have a node operating in Active status and a second node operating in Passive/Available status. If the active node goes down, the available node takes over processing. See High Availability .
	Virtual Resources	Virtual resource allows you to set up a "throttling" scheme that will manage how many and which tasks are sent to a particular resource at a time. See Creating Virtual Resources .
	Script Library	Script Library allows you to upload scripts into the Opswise database. You can then execute them using Windows, Linux/Unix, and SAP tasks without needing the scripts to exist on remote machines. See Script Library .
	Email Templates	Email template allows you to create commonly-used emails that can be referred to in an Email task. If an Email task specifies a template, Opswise uses the information in the template to construct and execute the Email task. Any information specified in the task overrides what is specified in the template. See Email Template .
	Email Connections	<p>Email connections are used two ways within Opswise:</p> <ul style="list-style-type: none"> • The Email Task uses the email connection to generate emails independent of tasks. See Email Task. • The Email Notification uses the email connection to generate notifications related to tasks. See Email Notifications and Email Connection. <div style="background-color: #ffffcc; padding: 5px; margin-top: 10px;"> <p> Note Email Connections are not used for Emailing reports.</p> </div>
	Database Connections	Database Connection provides all the database server information necessary for Opswise to execute a SQL or Stored Procedure task. See Database Connection .
	SAP Connections	SAP Connection provides all the SAP server information necessary for Opswise to execute a USAP command against the SAP instance. See SAP Connection .
	SNMP Managers	SNMP Managers are used to generate SNMP notifications as follows: <ul style="list-style-type: none"> • When an outboard component (agent, transporter, or hub) goes down or comes back up. See Sending Notifications on Outboard Component Status. • When you want to generate a notification associated with a task. See Creating SNMP Notifications and SNMP Manager.
	Applications	Application is a record defining a specific application (for example, Tomcat or a database) that runs on a machine somewhere, that you want to control (start, stop, or query) from Opswise. See Applications .
Automation Center Bundles & Promotion	Bundles	Opswise Bundling and Promoting features allow you to select and Bundle a group of Opswise records and "promote" them from one Opswise server to another.
	Promotion Targets	Before you can promote Bundles or individual records, you must identify and create a Promotion Target record(s) for the target machine(s).
	Promotion History	Promotion process creates audit records on the source and target machines. On the target machine, Opswise also creates a Promotion History record , which is a copy of the old record. This feature also supports a Restore option.
Automation Center Administration > Configuration	Properties	Allows you to configure Opswise system properties. For details see, Opswise System Properties .
	Report Email Properties	Allows you to set up an email server that will be used to automatically distribute reports. See Scheduling Automatic Report Distribution .
	LDAP Properties	Allows you to configure Lightweight Directory Access Protocol (LDAP) Properties. See LDAP Security .

	UI Properties	Allows you to configure properties of the Opwise user interface. See User Interface Properties .
	Data Backup / Purge	The Backup screen allows you to configure automatic backups and/or purges of Audits, recent Activity, and historical records. See Backing Up and Purging Data
	Maintenance Scripts	Maintenance scripts help you maintain and administer your Opwise installation. See Maintenance Scripts .
	Chart Colors	When you are monitoring a running workflow, the status of each task instance is color-coded. This feature allows you to customize the colors used for each status. See Changing a Task Status Color .
	Gauges	Gauges are online reports displayed on dashboards and home pages . This displays a list of all gauges defined in your system. Each gauge record displays properties including title and gauge type. See Creating and Deleting Gauges .
	Filters	Displays a list of all record filters for which the current user has permission. List filters are created using the filtering fields on the record list itself. Note that this feature is used only for record lists, not the Activity display. This feature allows you to update or delete existing filters. See Sorting and Filtering .
Automation Center Administration > Security	Users	Displays a list of users that have been defined in your system. See Adding Users .
	Groups	Displays a list of user groups that have been defined in your system. See Adding Groups .
	Audits	Opwise Automation Center audit function maintains a detailed record of all user interactions with Opwise, including before and after images related to any change and a description of the differences. See Audits .

Using Lists

- Introduction
- Sorting and Filtering
 - Sort a List of Records
 - Filter Out a Single Record from a List
 - Create a Filter
 - Run a Filter
 - Save a Filter
 - Run a Saved Filter
 - Remove Filtering from a List
 - Manage Filters
- Displaying/Rearranging Columns
- Displaying/Hiding Breadcrumbs
- Specifying the Number of Records Per Page
- Adding Records
- Deleting Records
- Updating Multiple Records
- Quickly Displaying Record Contents
- Searching for Records
- Generating Pie or Bar Charts from a List
- Exporting Records to an Output File
 - Exporting Records to Excel or CSV
 - Exporting Records to XML, XML (Export References), or Opwise Permissions for Group
 - Exporting Records to PDF

Introduction

A list is a screen display of records of the same type, such as a list of tasks, calendars, or users. When you click on an application or module in the Opwise navigation pane, Opwise displays a list of associated records in the center pane.

Sorting and Filtering

Sort a List of Records

You can sort a list of records either of two ways:

1. Click on the column name you want to sort on. A small arrow appears to the left of the name to indicate the direction of sorting. An up arrow indicates ascending alphabetical or numerical order; a down arrow indicates the opposite. Click again to reverse the direction of sorting.
2. Position the cursor to the right of the column name you want to sort on and right click. The UI displays a menu that includes the following options:
 - Click **Sort (a to z)** to sort objects in this column in ascending alphabetical or numerical order.
 - Click **Sort (z to a)** to sort objects in this column in descending order.

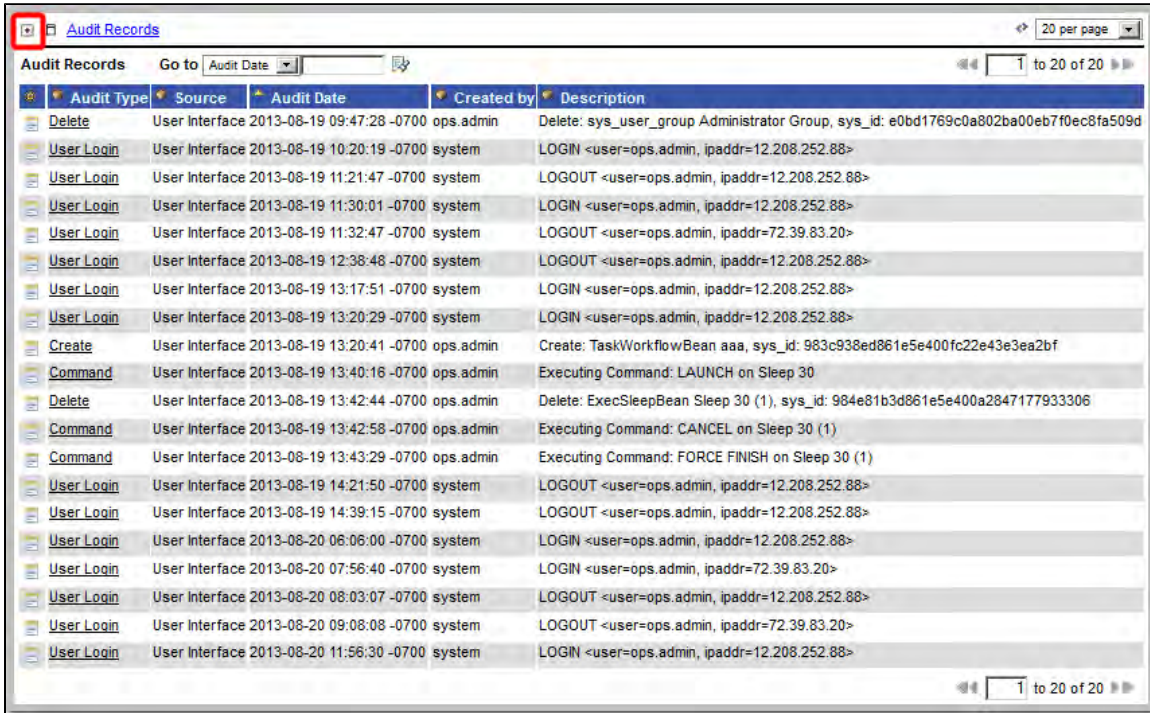
Filter Out a Single Record from a List

Step 1	Right-click on the record name.
Step 2	Select Filter Out . This quick filter is added as a breadcrumb.
Step 3	To filter the object back in, click on the prior breadcrumb.

Create a Filter

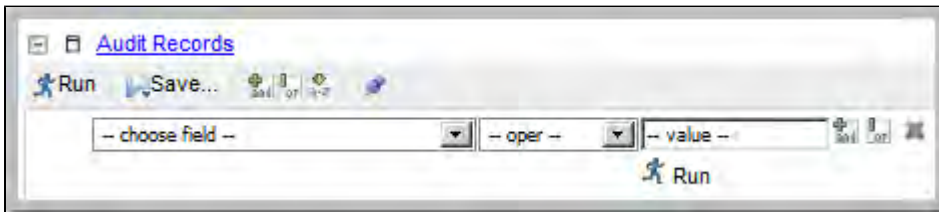
You can create a filter and either [run it](#) immediately or [save it](#) to apply it later.

Step 1 Click the plus sign in the upper left corner.



Audit Type	Source	Audit Date	Created by	Description
Delete	User Interface	2013-08-19 09:47:28 -0700	ops.admin	Delete: sys_user_group Administrator Group, sys_id: e0bd1769c0a802ba00eb7f0ec8fa509d
User Login	User Interface	2013-08-19 10:20:19 -0700	system	LOGIN <user=ops.admin, ipaddr=12.208.252.88>
User Login	User Interface	2013-08-19 11:21:47 -0700	system	LOGOUT <user=ops.admin, ipaddr=12.208.252.88>
User Login	User Interface	2013-08-19 11:30:01 -0700	system	LOGIN <user=ops.admin, ipaddr=12.208.252.88>
User Login	User Interface	2013-08-19 11:32:47 -0700	system	LOGOUT <user=ops.admin, ipaddr=72.39.83.20>
User Login	User Interface	2013-08-19 12:38:48 -0700	system	LOGOUT <user=ops.admin, ipaddr=12.208.252.88>
User Login	User Interface	2013-08-19 13:17:51 -0700	system	LOGIN <user=ops.admin, ipaddr=12.208.252.88>
User Login	User Interface	2013-08-19 13:20:29 -0700	system	LOGIN <user=ops.admin, ipaddr=12.208.252.88>
Create	User Interface	2013-08-19 13:20:41 -0700	ops.admin	Create: TaskWorkflowBean aaa, sys_id: 983c938ed861e5e400fc22e43e3ea2bf
Command	User Interface	2013-08-19 13:40:16 -0700	ops.admin	Executing Command: LAUNCH on Sleep 30
Delete	User Interface	2013-08-19 13:42:44 -0700	ops.admin	Delete: ExecSleepBean Sleep 30 (1), sys_id: 984e81b3d861e5e400a2847177933306
Command	User Interface	2013-08-19 13:42:58 -0700	ops.admin	Executing Command: CANCEL on Sleep 30 (1)
Command	User Interface	2013-08-19 13:43:29 -0700	ops.admin	Executing Command: FORCE FINISH on Sleep 30 (1)
User Login	User Interface	2013-08-19 14:21:50 -0700	system	LOGOUT <user=ops.admin, ipaddr=12.208.252.88>
User Login	User Interface	2013-08-19 14:39:15 -0700	system	LOGOUT <user=ops.admin, ipaddr=12.208.252.88>
User Login	User Interface	2013-08-20 06:06:00 -0700	system	LOGIN <user=ops.admin, ipaddr=12.208.252.88>
User Login	User Interface	2013-08-20 07:56:40 -0700	system	LOGIN <user=ops.admin, ipaddr=72.39.83.20>
User Login	User Interface	2013-08-20 08:03:07 -0700	system	LOGOUT <user=ops.admin, ipaddr=12.208.252.88>
User Login	User Interface	2013-08-20 09:08:08 -0700	system	LOGOUT <user=ops.admin, ipaddr=72.39.83.20>
User Login	User Interface	2013-08-20 11:56:30 -0700	system	LOGIN <user=ops.admin, ipaddr=12.208.252.88>

The filter dialog displays.

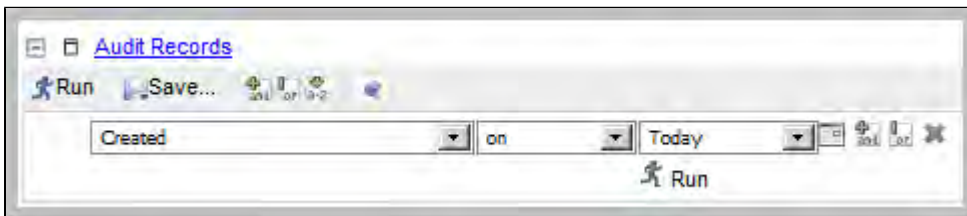


Step 2 Make a selection from the drop-down list for each filter condition:






- Field from this record type (- - choose field - -)
- Relevant comparison operator (- - oper - -)
- Relevant object or field to compare the field with (- - value - -)

For example:

- Created
- On
- Today



Step 3 Optionally, you can enhance your filters using any of the following:

- To add an AND condition, click .
- To add an OR condition, click .
- To sort the results alphabetically by a field for this record type, click .
- To pin the selected filter conditions to the top of the screen after you **Run** or **Save** the filter, click .
- To select a specific date for a date-related field for this record type, click .

Run a Filter

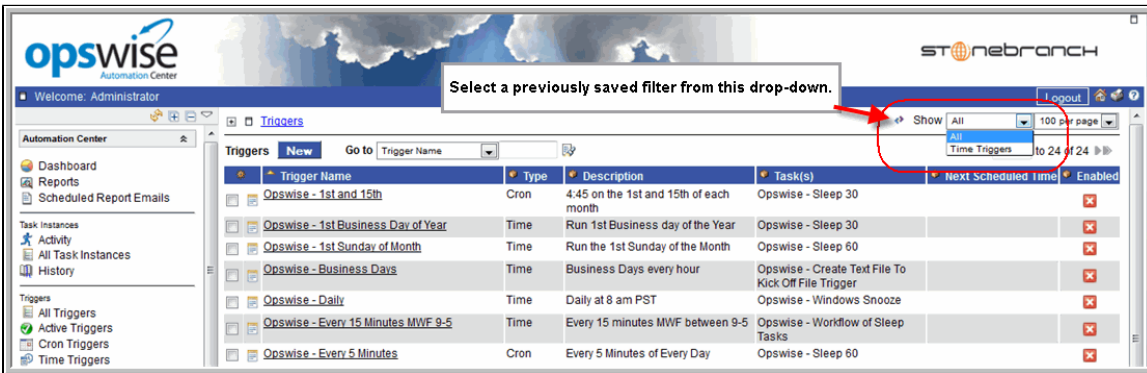
To apply the filter to the list, click **Run**. Note that the filter name is added to the breadcrumb.

Save a Filter

Step 1	Click Save .
Step 2	Specify the filter name.
Step 3	Specify which users are allowed to use the filter.
Step 4	Click the Save button.

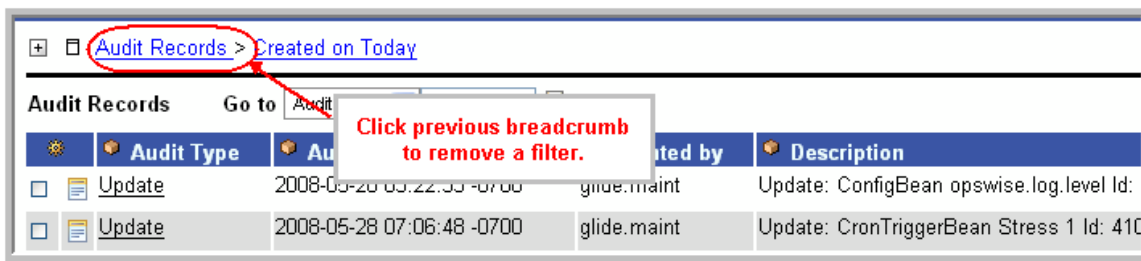
Run a Saved Filter

Select the saved filter you want to run from the **Show** drop down just below the title bar. The UI applies the filter and adds the filter name to the breadcrumbs. The Show drop-down only appears if there are saved filters associated with this list.



Remove Filtering from a List

Click on the previous breadcrumb, as shown on the example below.



Manage Filters

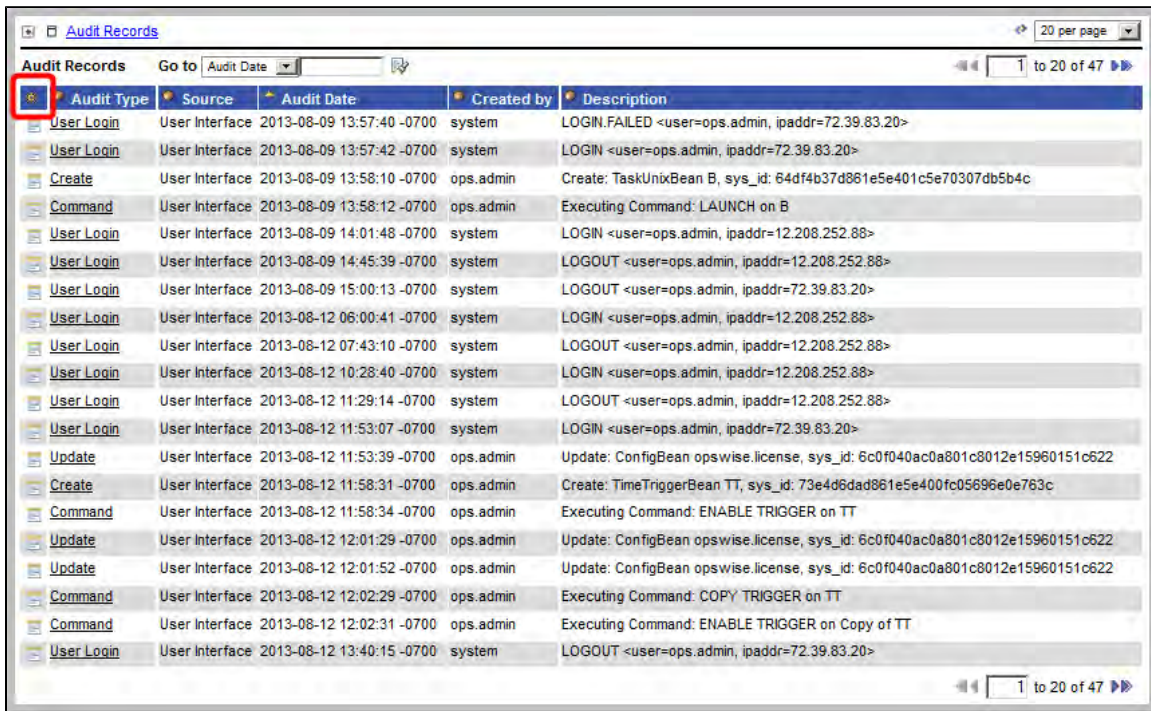
From the navigation pane, select **Automation Center Administration > Filters**. Opswise displays all filters for which you have permission. This feature allows you to modify or delete filters. To open a filter record, click on the record name. You can make changes and click **Update** or click **Delete** to remove a filter from Opswise.

Displaying/Rearranging Columns

All lists display default columns of information in a default sequence.

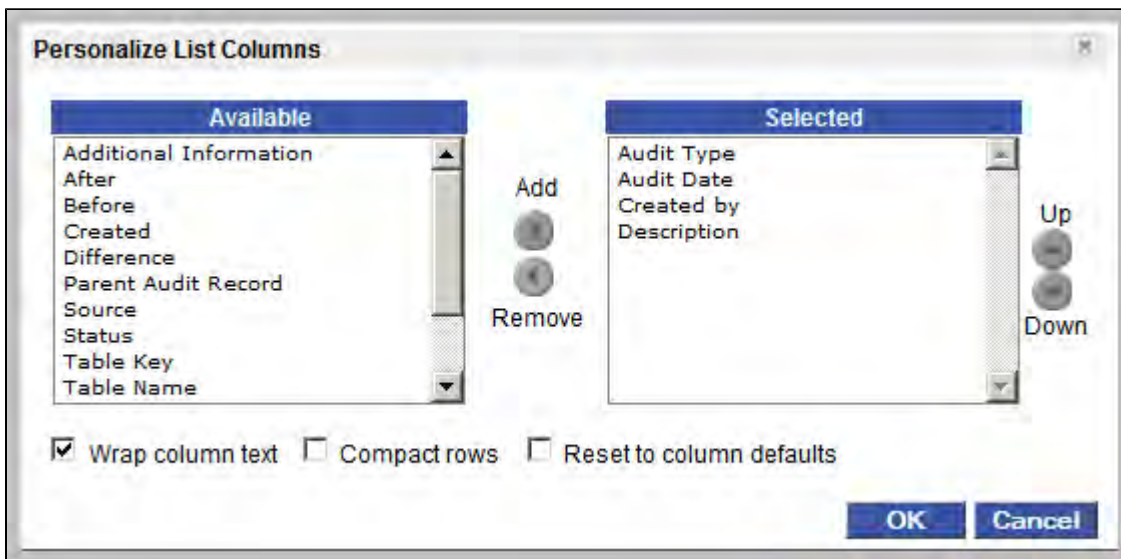
You can change the number of columns that displays and rearrange the sequence in which they display.

Step 1 Click the Personalize defaults icon that displays at the top of the left-most column in the list.



Audit Type	Source	Audit Date	Created by	Description
User Login	User Interface	2013-08-09 13:57:40 -0700	system	LOGIN.FAILED <user=ops.admin, ipaddr=72.39.83.20>
User Login	User Interface	2013-08-09 13:57:42 -0700	system	LOGIN <user=ops.admin, ipaddr=72.39.83.20>
Create	User Interface	2013-08-09 13:58:10 -0700	ops.admin	Create: TaskUnixBean B, sys_id: 64df4b37d881e5e401c5e70307db5b4c
Command	User Interface	2013-08-09 13:58:12 -0700	ops.admin	Executing Command: LAUNCH on B
User Login	User Interface	2013-08-09 14:01:48 -0700	system	LOGIN <user=ops.admin, ipaddr=12.208.252.88>
User Login	User Interface	2013-08-09 14:45:39 -0700	system	LOGOUT <user=ops.admin, ipaddr=12.208.252.88>
User Login	User Interface	2013-08-09 15:00:13 -0700	system	LOGOUT <user=ops.admin, ipaddr=72.39.83.20>
User Login	User Interface	2013-08-12 06:00:41 -0700	system	LOGIN <user=ops.admin, ipaddr=12.208.252.88>
User Login	User Interface	2013-08-12 07:43:10 -0700	system	LOGOUT <user=ops.admin, ipaddr=12.208.252.88>
User Login	User Interface	2013-08-12 10:28:40 -0700	system	LOGIN <user=ops.admin, ipaddr=12.208.252.88>
User Login	User Interface	2013-08-12 11:29:14 -0700	system	LOGOUT <user=ops.admin, ipaddr=12.208.252.88>
User Login	User Interface	2013-08-12 11:53:07 -0700	system	LOGIN <user=ops.admin, ipaddr=72.39.83.20>
Update	User Interface	2013-08-12 11:53:39 -0700	ops.admin	Update: ConfigBean opswise.license, sys_id: 6c0f040ac0a801c8012e15960151c622
Create	User Interface	2013-08-12 11:58:31 -0700	ops.admin	Create: TimeTriggerBean TT, sys_id: 73e4d6dad861e5e400fc05696e0e763c
Command	User Interface	2013-08-12 11:58:34 -0700	ops.admin	Executing Command: ENABLE TRIGGER on TT
Update	User Interface	2013-08-12 12:01:29 -0700	ops.admin	Update: ConfigBean opswise.license, sys_id: 6c0f040ac0a801c8012e15960151c622
Update	User Interface	2013-08-12 12:01:52 -0700	ops.admin	Update: ConfigBean opswise.license, sys_id: 6c0f040ac0a801c8012e15960151c622
Command	User Interface	2013-08-12 12:02:29 -0700	ops.admin	Executing Command: COPY TRIGGER on TT
Command	User Interface	2013-08-12 12:02:31 -0700	ops.admin	Executing Command: ENABLE TRIGGER on Copy of TT
User Login	User Interface	2013-08-12 13:40:15 -0700	system	LOGOUT <user=ops.admin, ipaddr=72.39.83.20>

Step 2 On the Personalize List Columns dialog, select the columns that you want to display and the sequence in which you want them displayed.



Step 3 Click **OK** to change the display of columns as selected.

Displaying/Hiding Breadcrumbs

All lists display breadcrumbs. Breadcrumbs keep track of your location within the UI, and what, if any, filters you have applied. The following example shows breadcrumbs that represent the application being displayed (Audits), along with a filter (Audit Date on Last month) that has been applied to the list.

A screenshot of the 'Audit Records' interface. At the top, there is a breadcrumb trail: 'Audit Records > Audit Date on Last month'. The 'Audit Date on Last month' breadcrumb is highlighted with a red rectangular box. Below the breadcrumb, there is a search bar with 'Go to' and a dropdown menu for 'Audit Type'. The main content is a table with the following columns: 'Audit Type', 'Audit Date', 'Created by', and 'Description'. The table contains 17 rows of audit records. At the bottom right, there is a pagination control showing '1 to 17 of 17'.

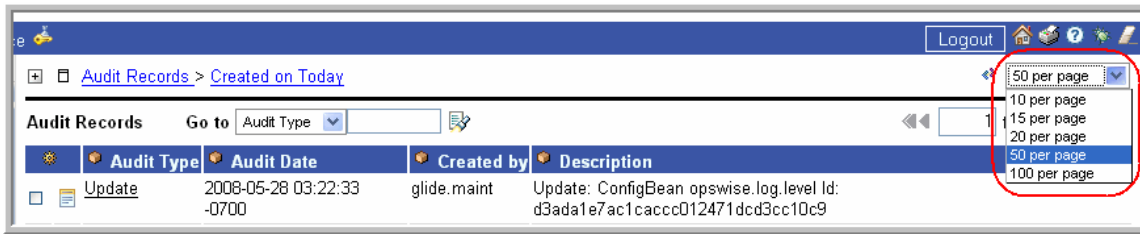
Audit Type	Audit Date	Created by	Description
User Login	2012-09-28 07:46:04 -0700	system	LOGIN.FAILED <user=ops.admin>
User Login	2012-09-28 07:46:08 -0700	system	LOGIN <user=ops.admin, ipaddr=72.39.88.171>
Create	2012-09-28 07:46:37 -0700	ops.admin	Create: GenericGroupBean HR, sys_id: 0d57ed85d861e5e4006a8f3713c1265a
Update	2012-09-28 07:46:51 -0700	ops.admin	Update: TaskSleepBean Sleep 0, sys_id: 410d6c0bc0a801c901838d8ac43b3279
User Login	2012-09-28 07:47:54 -0700	system	LOGOUT <user=ops.admin>
User Login	2012-09-28 08:38:27 -0700	system	LOGIN <user=ops.admin, ipaddr=72.39.88.171>
System Script	2012-09-28 08:39:20 -0700	ops.admin	Running System Script: maintenance_updates.js
User Login	2012-09-28 08:39:31 -0700	system	LOGIN <user=ops.admin, ipaddr=72.39.88.171>
User Login	2012-09-28 08:39:31 -0700	system	LOGOUT <user=ops.admin>
User Login	2012-09-28 08:39:45 -0700	system	LOGIN <user=ops.admin, ipaddr=72.39.88.171>
User Login	2012-09-28 09:22:38 -0700	system	LOGIN <user=ops.admin, ipaddr=72.39.88.171>
User Login	2012-09-28 09:48:36 -0700	system	LOGOUT <user=ops.admin>
User Login	2012-09-28 10:23:36 -0700	system	LOGOUT <user=ops.admin>
User Login	2012-09-28 10:55:28 -0700	system	LOGIN <user=ops.admin, ipaddr=72.39.88.171>
User Login	2012-09-28 13:34:39 -0700	system	LOGOUT <user=ops.admin>
User Login	2012-09-30 11:06:55 -0700	system	LOGIN <user=ops.admin, ipaddr=24.7.24.160>
User Login	2012-09-30 12:07:19 -0700	system	LOGOUT <user=ops.admin>

To toggle breadcrumbs, click on the icon next to the topmost breadcrumb, as shown in the following example.

A screenshot of the 'Audit Records' interface, similar to the one above. In this version, the icon next to the topmost breadcrumb 'Audit Records' is highlighted with a red rectangular box. The rest of the interface, including the search bar, table of audit records, and pagination, is identical to the previous screenshot.

Specifying the Number of Records Per Page

From the drop-down menu in the upper right corner, select the number of records per page you want to display.



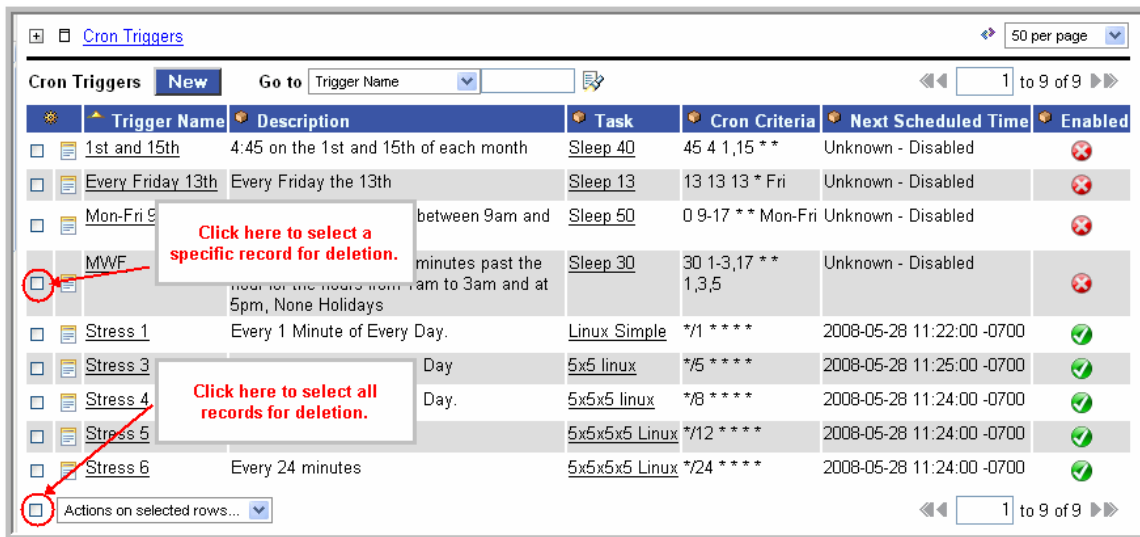
Adding Records

Adding a new record is a simple uniform procedure through the environment. From any list, click the **New** button.

Deleting Records

To delete one or more specific records, click on the box associated with those records and select **Delete** from the **Actions on selected rows...** menu.

To delete all records currently being displayed on the list, click on the box next to the **Actions on selected rows...** menu and select **Delete**.



Updating Multiple Records



Warning!

Exercise great caution when using this powerful feature.

Two menu options available from any list of records allow you to make updates to multiple records:

- **Update Entire List** - Allows you to make updates to all records that match your selection filter (not just those on the current display). If you specify no filter, Opwise selects all records of that type.
- **Update Selected** - Allows you to select specific records to update by clicking the box to the left of the record name.

You can make these "global" changes to the list for any field that is common to every record in the list or every selected record. For example, if your list includes multiple Cron triggers, you can make "global" changes to all fields on a Cron trigger. If your list includes all task types, you can make "global" changes to any field that is common to all task types. The more uniform your list is, the greater the number of fields you can change.

In order to use these features, your user ID must have the `list_updater` role.

Step 1	Optionally, filter the list so that it selects only the records you want to update.
Step 2	If you only want to update some of the records in the displayed list, select those records by clicking the box to the left of the name.
Step 3	Right-click on the blue title bar. A menu displays.

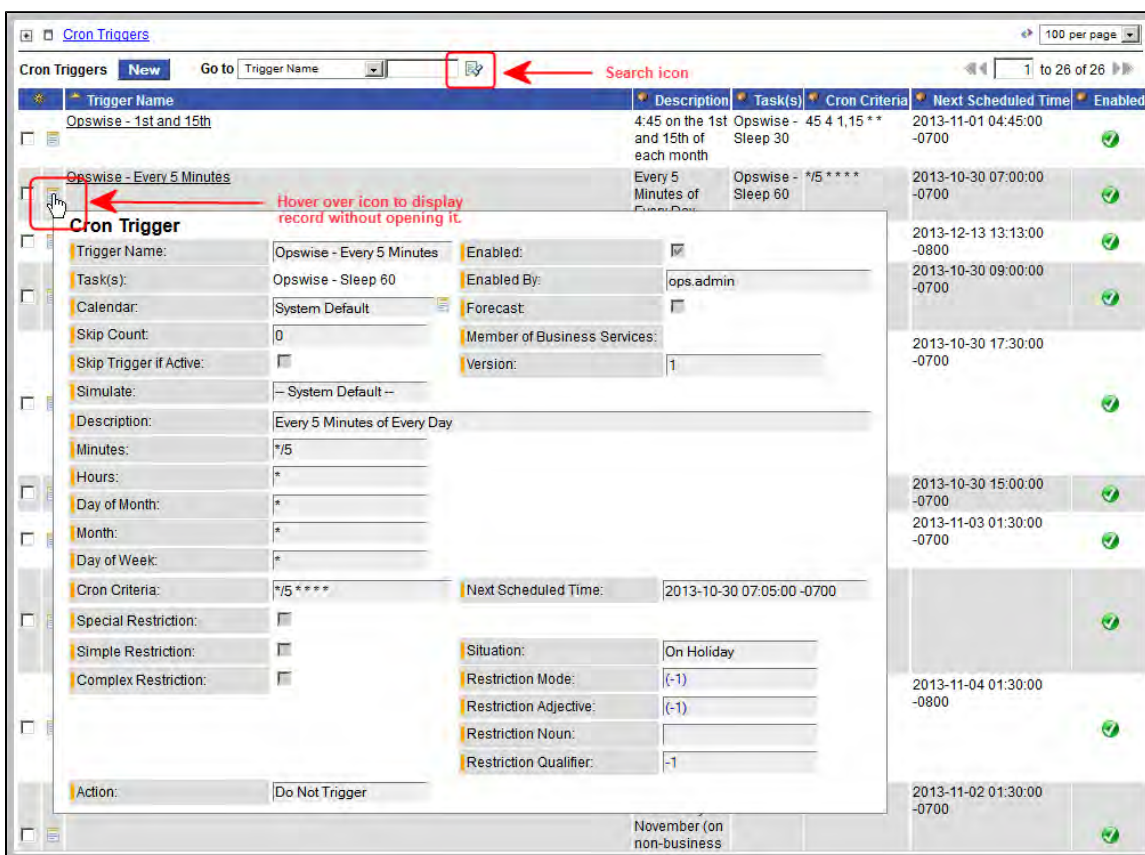
Step 4	Select Update Entire List or Update Selected as appropriate. A form pops up, showing which fields you will be allowed to change. These fields are common to all the records selected by your filter or manually selected by you. If you did not use a filter or select specific records, all records of this type are considered "selected."
Step 5	Enter the global changes you want to make. <div style="border: 1px solid black; background-color: #ffffcc; padding: 10px; margin: 10px 0;"> <p>Note For drop-down menus and true/false fields, the default selection None means that the field will not be changed on any records. If you select anything else in such a field, the new value will be applied to all selected records.</p> </div>
Step 5	When you are finished, click Update . Opswise displays the number of records that will be updated and asks you to confirm the operation.
Step 7	Once you confirm, Opswise applies your changes to all the selected records.

Quickly Displaying Record Contents

You can display the contents of a record without opening it. Hover over the paper icon to the left of the record name, as shown in the illustration below.

Searching for Records

You can perform searches from any list, using the **Go to** boxes, shown below.



The options in the **Go to** box vary depending on the module and the columns displaying on the list.

To perform a search:

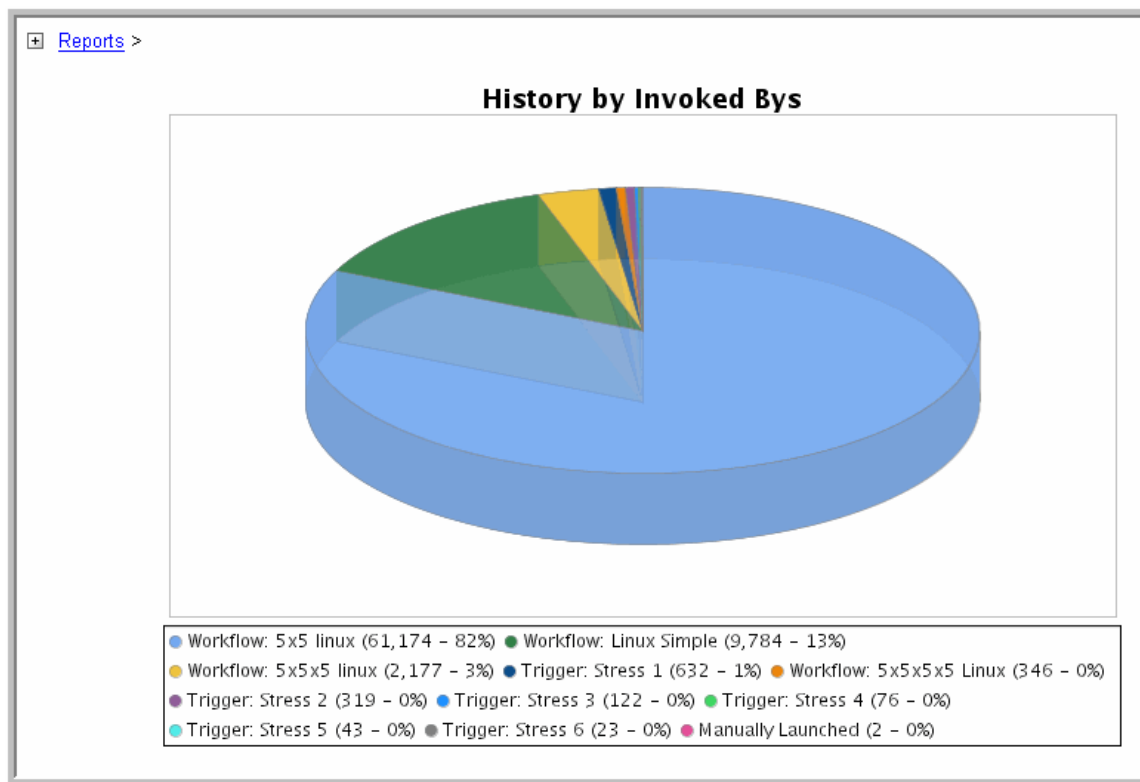
Step 1	Select the field you want to search on. For example, Name.
Step 2	Enter a value appropriate to that field. For text fields, you can search on a full or partial string.

Step 3 Click the search icon. The search fetches all records that match and places them at the top of the list.

Generating Pie or Bar Charts from a List

This feature allows you to quickly generate a pie or bar chart that sorts records in the list based on a specific column. For example, you can quickly generate a pie chart that sorts tasks by type.

In the following example, the pie chart was generated by right-clicking the **Invoked By** column on the Activity History list and selecting Pie Chart.



Step 1 Click in the column header from which you want to generate a chart.

Step 2 Right-click and select **Pie Chart** or **Bar Chart**. The UI generates the chart.

Step 3 Optionally, click the Print icon in the upper right corner to print the chart. To escape the chart, select any menu option or your browser's Back key.

Exporting Records to an Output File

This feature allows you to export record information to any of the following file types:

- Excel
- CSV (comma-separated values in an Excel file)
- XML
- XML (Export References)
- Opwise Permissions for Group (XML for Opwise Security Groups only)
- PDF

To export records, you first display the current list of those records. An export contains all records currently included on the list, even if your records displayed per page selection does not show them all. If you do not want to export all records on the list, use filtering to select the records to be exported.



Note

You cannot select records on a list to indicate which records are to be exported; you must filter the list.

When you export to Excel, CSV, or PDF, you export only the record fields currently displayed on the list. To select which fields are displayed - and

thus include them in the export - click the Personalize Default icon at the top of the first column.

When you export to XML, XML (Export References), or Opwise Permissions for Group, you export the entire record definition.

Exporting Records to Excel or CSV

To export records to Excel or CSV:

Step 1	Display the list of records.
Step 2	Right-click in any column header of the list.
Step 3	Select Export .
Step 4	Select Excel or CSV .
Step 5	When the export is complete, click Download.
Step 6	Open or save the file.

Exporting Records to XML, XML (Export References), or Opwise Permissions for Group

To export records to XML, XML (Export References), or Opwise Permissions for Group (XML for [Opwise Security Groups](#) only):

Step 1	Display the list of records.
Step 2	Right-click in any column header of the list.
Step 3	Select Export .
Step 4	<p>Select Excel, CSV, XML, or XML (Export References).</p> <ul style="list-style-type: none"> • If you select XML when exporting a Task record, you will also export local variables, Actions, Notes, and Virtual Resource requirements. • If you select XML when exporting a Workflow, you will also export all tasks in the workflow. Each task is exported to a separate XML file. • If you select XML (Export References) when exporting a Task record, you will export all of the XML export data, plus triggers, global variables, calendars, credentials, and any resources referred to. These "reference" records are saved to separate XML files.
Step 5	When the export is complete, an Exported to: message displays above the list, which identifies the name and location of the export file. (The location is configurable; see Export Path in Opwise System Properties .)

Exporting Records to PDF

To export records to a PDF file:

Step 1	Display the list of records.
Step 2	Right-click in any column header of the list.
Step 3	Select Generate PDF .
Step 4	Select a PDF layout.
Step 5	When the export is complete, click Download.
Step 6	Open or save the PDF.

Using Forms

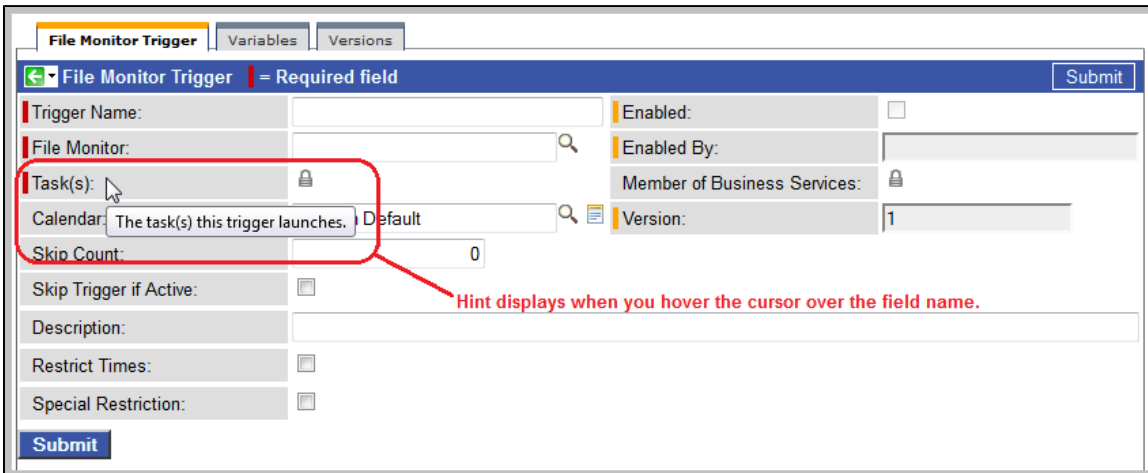
- Introduction
- Field Hints
- Color-Coding of Tabs
- Saving, Updating, Deleting, and Copying Records
- Expanding/Contracting a Text Field
- Browsing For and Selecting One or More Records
- Customizing the Navigation Pane

Introduction

A "form" is the screen used to create a record.

Field Hints

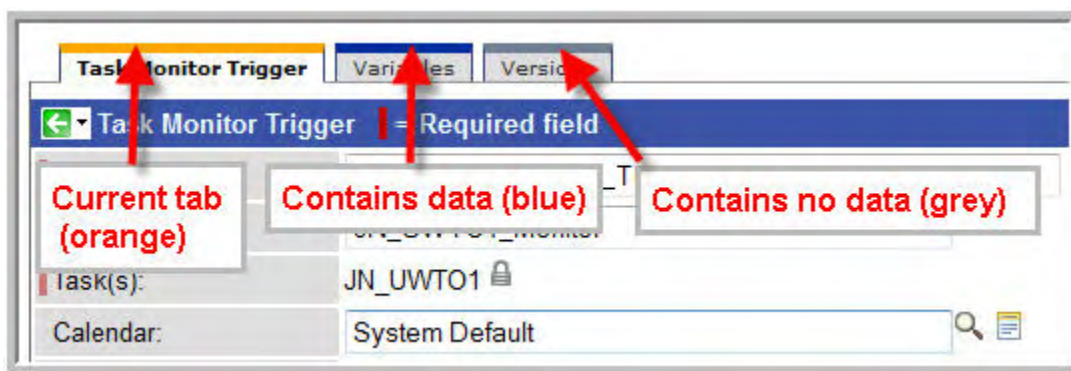
To display a field hint (a brief description of a field), hover the cursor over the field name.



Color-Coding of Tabs

Each record contains one or more tabs of information. The tabs are color-coded:

- The current tab displays an orange rule.
- Tabs that contain data display a blue rule.
- Tabs that do not contain data display a grey rule.



Saving, Updating, Deleting, and Copying Records

Save a New Record and Return to the Previous Menu

Step 1	Click New from the list screen, select the record type, if appropriate, and fill out the fields as required.
Step 2	Click the Submit button.

Save a New Record without Returning to the Menu

Step 1	Click New from the list screen, select the record type, if appropriate, and fill out the fields as required.
Step 2	Hover your cursor over the down arrow on the title bar or right-click the title bar and select Save .

Update a Record and Return to the Previous Menu

Step 1	Display the record and make the updates you want.
Step 2	Click the Update button.

**Note**

If you change the name of a task that is part of a workflow, Opwise automatically changes the name of that task within the workflow itself.

Update a Record without Returning to the Menu

Step 1	Display the record and make the updates you want.
Step 1	Hover your cursor over the down arrow on the title bar or right-click the title bar and select Save .

Delete a Record

Step 1	Display the record you want to delete.
Step 2	Click the Delete button and click Yes to confirm.
Step 3	You can also delete records from the list screen. See Deleting Records .



Copy a Record**Caution**

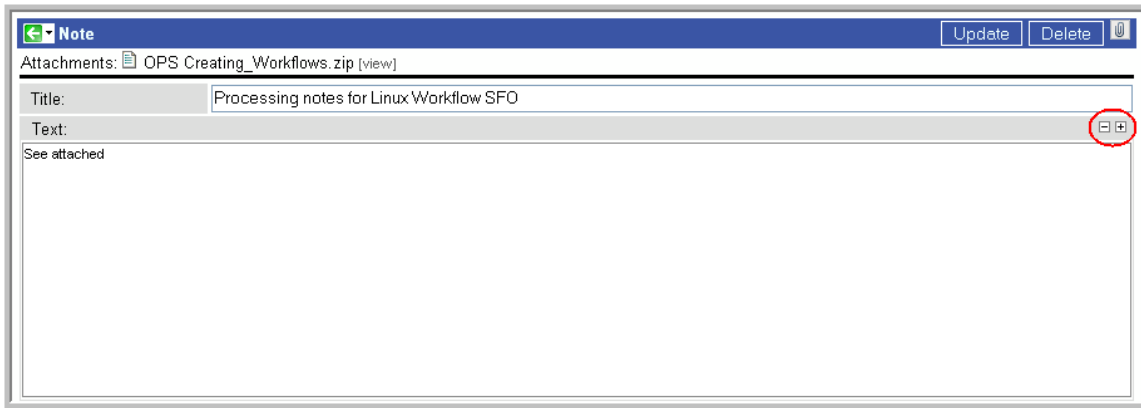
Do not use the **Update** button to copy a record. The **Update** button overwrites the existing record.

Also, do not use the following method of copying for a task, trigger, or calendar unless there are no associated records such as Actions, Notes, Variables, and so on. This method does not copy any records attached to the record. To copy a task, trigger, or calendar with all of its associated records, use the method described in [Copying Tasks](#), [Copying Triggers](#), and [Copying Calendars](#).

Step 1	Display the record definition you want to copy.
Step 2	Give the record a new name and make any other necessary changes.
Step 3	Hover over the down arrow or right-click the title bar and select Insert or Insert and Stay .

Expanding/Contracting a Text Field

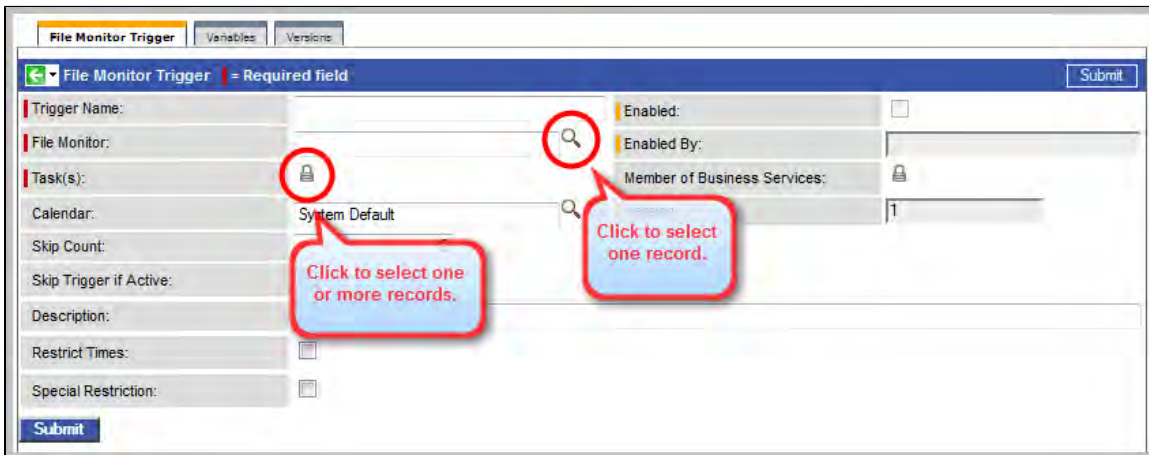
Free-text fields are defaulted to a particular display size. You can expand and contract the display of the field by clicking the plus  and minus  signs at the top right corner of the field.





Browsing For and Selecting One or More Records

Many fields contain data from another table. This type of field, called a Reference field, has either a magnifying glass or lock icon at the right end.

- Fields with a magnifying glass icon allow you to select one record.
- Fields with a lock icon allow you to select multiple records.

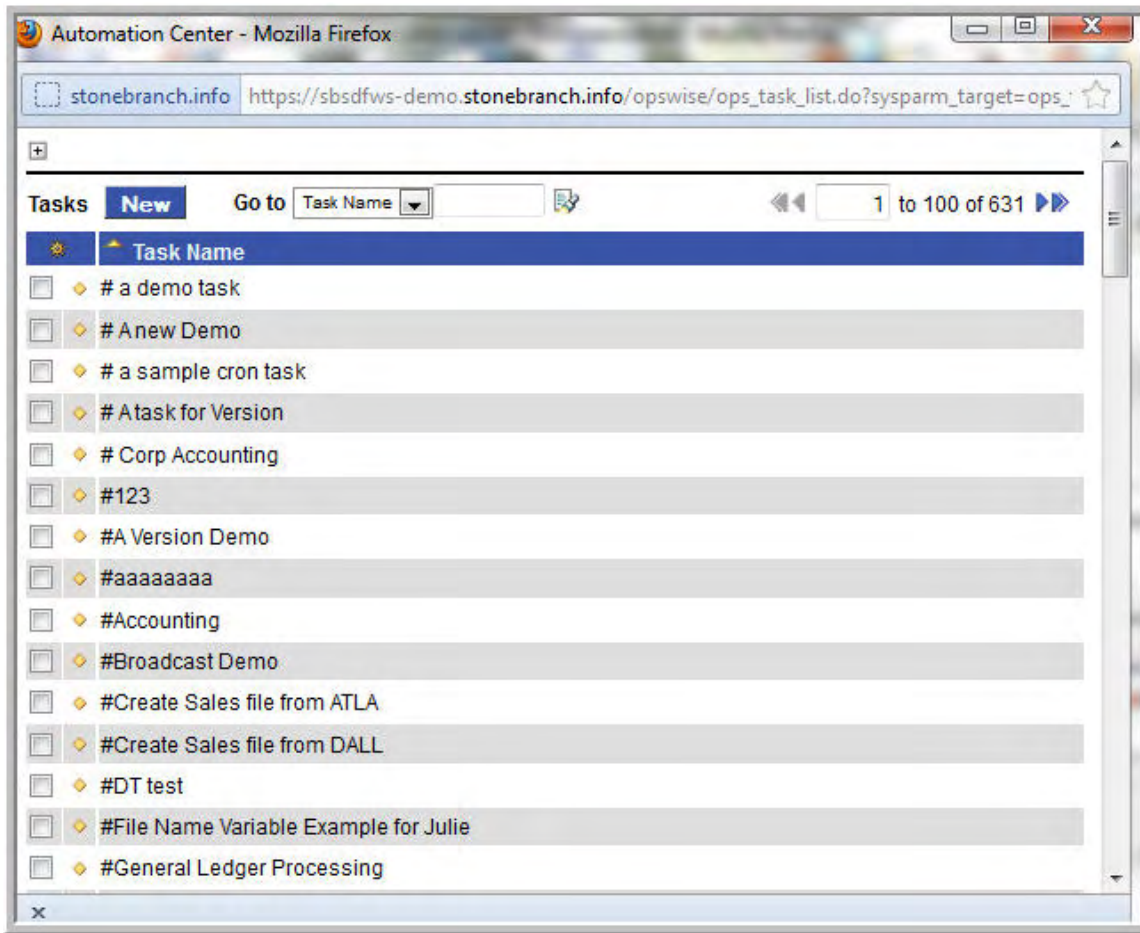


To select records for a multiple-record field, click the [lock icon](#). Two additional fields display:

- Browse field with a magnifying glass that lets you select multiple records.
- Field above the Browse field that will contain the record names you select. From here you can delete  and view  any of the selected records.

To select records from a browse field:

Step 1	Click the magnifying glass. A pop-up displays the contents of the associated table, as shown in the following example.
Step 2	To select a record, click its name.



Customizing the Navigation Pane

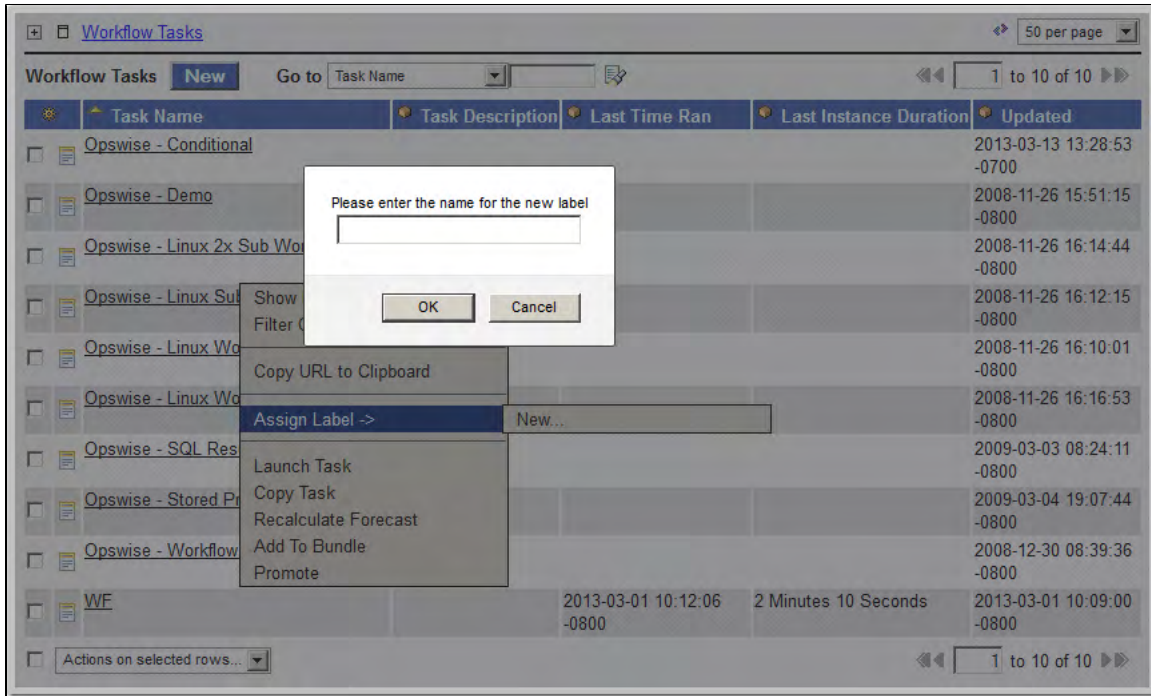
Using Labels, you can add menu options to the navigation pane and assign records to them. This allows you to organize records into business groups for easy access from your navigation pane. For example, you might create a Label for a department and assign all related records to that Label.

When you organize records into Labels, the records are still accessible from their original menus as well.

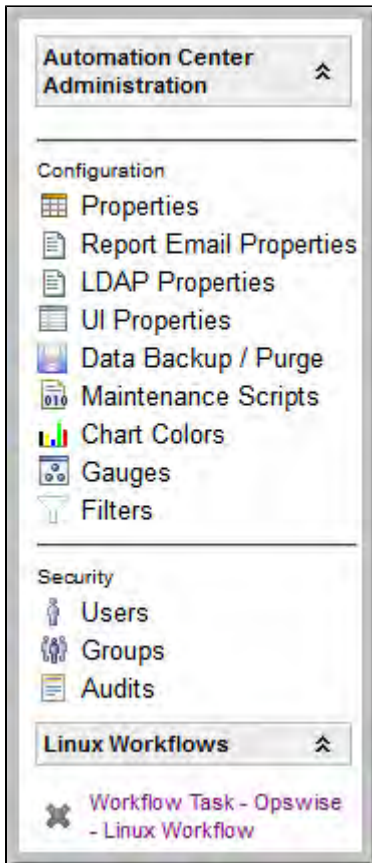
To create a new Label and assign a record to it

- | | |
|---------------|--|
| Step 1 | Right-click a record on the records list screen, or open the record and hover over the down arrow or right-click the title bar, to display a menu of commands. |
|---------------|--|

Step 2 Hover your cursor over **Assign Label ->** and click **New....** A pop-up dialog requests the name for a new label.



Step 3 Enter a name for the new Label and click **OK**. The new Label displays at the bottom of the navigation pane with the selected record added to it (click the carets next to the name of the new label to see the record).



To assign a record to an existing Label

Step 1	To assign another record to the Label, right-click the record; or, open the record and hover over the down arrow or right-click the title bar.
Step 2	Highlight Assign Label , select New and type in the name of the existing Label. The UI adds the record to the Label menu.

To remove a record from a Label

Either:

- Click the **X** to the left of the record name under the Label heading.
- Display the record, right-click the title bar. and select **Remove Label**.

To delete a Label

Step 1	Click the folder name. The UI displays the Label definition screen.
Step 2	Click the Delete button.

Using Wildcards

The Opswise Automation Center user interface supports two wildcards:

- Asterisk (*)
- Question mark (?)

You can use wildcards in record searches and when applying some rule or command against records. Fields that support wildcards are identified in the field description for that field.

- An asterisk (*) represents a wildcard of any number of characters. For example, a search for string "FEE*SF" returns all records whose name begins with "FEE" and ends with "SF", with any number and type of characters between the two strings.
- A question mark (?) represents a wildcard of one character in a specific position. For example, a search for string "FEE?SF" returns all records whose name begins with "FEE" and ends with "SF", with any single character between the two strings.
- Multiple question marks represent wildcards of multiple characters in a specific position. For example, a search for string "FEE??SF" returns all records whose name begins and ends with "FEE" and "SF", respectively, with any two characters between the two strings.

Naming Tips

Many functions within Opwise are executed against one or more records. For example, you can give a user permission to change only certain tasks, issue commands against a group of task instances, or filter a trigger list to display only certain triggers.

Two methods are available to help you organize your records to facilitate the use of these functions.

Method 1	Develop a naming scheme for records. For example, when naming tasks, you could prepend with SF all tasks related to San Francisco operations, or you could prepend with REPT all report-related tasks. With such a naming scheme, you can sort and filter lists by selecting records, for example, that begin with "REPT." You can assign permissions and execute commands against records using the same method.
Method 2	Use Business Services , which simply is a method of grouping records. Whenever you create a record, you can assign it to a Business Service. For example, you could have a Business Service called "SF" and a Business Service called "REPT." Using this method, you could then filter or sort a list based on the Business Service. As another example, you could assign permissions to a user, giving the user update permission to all records in the "REPT" Business Service. Business Services allow you to create groups based on business functions and organize all your Opwise records according to user-defined categories.

Business Services

- [Overview](#)
 - [Business Service Usage](#)
 - [Record Types for Business Services](#)
- [Creating Business Services](#)
- [Assigning a Record to One or More Business Services](#)

Overview

The Opwise Business Services feature allows you to organize your data into groups of related information.

You can create Business Services that represent your organization and assign [Opwise records](#) to one or more of those Business Services. You can then sort and filter screens based on the Business Services, as well as generate reports.

You also can take advantage of Business Services when you set up security by [assigning permissions](#) only to users and/or user groups that belong to specific Business Services.

Business Service Usage

For example, you may want to place all record types related to accounting in an Business Service named Accounting.

A Business Service of related record types can be identified via:

- Permissions
- Reports
- Dashboard view
- Filtering

Record Types for Business Services

You can assign one or more of the following record types to one or more Business Services:

- Agents
- Applications
- Calendars
- Credentials
- Scripts
- Tasks
- Task Instances
- Triggers

Creating Business Services

You may need administrative privileges to perform this procedure.

Step 1 From the navigation pane, select **Automation Center > Business Services**. The Business Services List screen displays.

Name	Description	Version
Operations	This is the Operation business service.	1
Tech Support	This is the Tech Support business service.	1

Step 2 Click **New**. The Business Service Definition screen displays.

The screenshot shows the 'Business Service' definition interface. At the top, there are two tabs: 'Business Service' and 'Versions'. Below the tabs is a blue header bar with a back arrow, the text 'Business Service', a red asterisk indicating a required field, and a 'Submit' button. The main form area contains three fields: 'Name:' with an empty text input, 'Description:' with a large text area, and 'Version:' with a text input containing the number '1'. A 'Submit' button is located at the bottom left of the form area.

Step 3 Enter a business service Name and Description and click **Submit**.

Assigning a Record to One or More Business Services


When defining an Opswise record, use the Member of Business Services field to select one or more Business Services for that record.

Home Page, Dashboard, and Gauges

Home Page, Dashboard, and Gauges

Home Page

The [Home Page](#) displays when you log in to Opswise Automation Center. It is associated with the login ID of a user; users can customize their own Home Page.

To access the Home Page, after you have logged in and navigated to other pages, click the Home  icon in the right-hand corner of any screen.

Dashboard

The [Dashboard](#) also is customizable. To access the Dashboard, click **Automation Center > Dashboard** from the navigation pane.

Gauges

A gauge is a "live" report, using information from an [Opswise Report table](#), whose information is updated automatically according to the refresh setting on each of those pages.

Both the Home Page and the Dashboard use [Gauges](#) to display data. You can add and remove gauges on your Home Page and the Dashboard, and you can create your own gauges for display on either or both those pages.


Using the Home Page

- Overview
- Moving Widgets
- Removing Widgets
- Refreshing
- Adding Widgets
 - Widget Descriptions

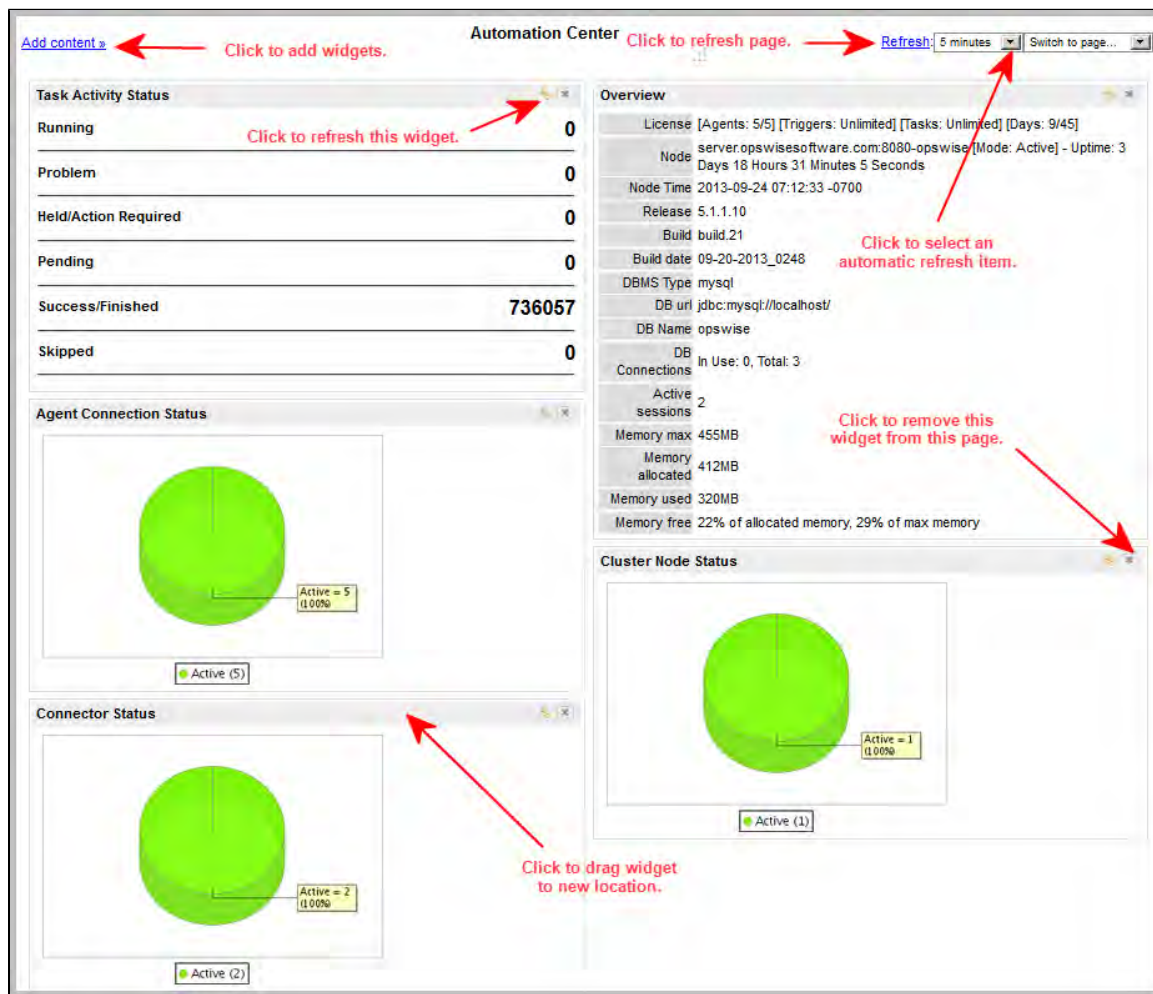
Overview

The Home Page displays when you log in to Opswise Automation Center.

Each window on the Home Page is a **widget** containing a different set of information, in text or graphic format: filter, gadget, gauge, label, scroller, system application, or world clocks. You can customize your Home Page to display any number of available widgets.

Once you navigate away from the Home Page, you can return to it by clicking the Home  icon that displays at the top right corner of every page.

A sample Home Page is shown below.



This sample Home Page displays the following widgets:

Name	Type of Widget	Description

Task Activity Status	gauge	Summary of task instances, sorted by status. You can click on any status for detailed information.
Overview	gadget	System configuration information.
Agent Connection Status	gauge	Status of all connected agents.
Cluster Node Status	gauge	Status of defined cluster nodes. These gauges are created automatically.
Connector Status	gauge	Defined Connectors. Connectors (Message Hubs and Transporters) form a communications link between the Controller and Opwise agents. These gauges are created automatically.

You can click on information and/or graphics in the widgets to display detailed information.

Moving Widgets

Each widget on the Home Page has a grey title bar. To move an widget, click the title bar and drag it to a new position.

Removing Widgets

To remove an widget from the Home Page, click the **X** in the top right corner of the widget's title bar.

Refreshing

You can specify the following refresh times on the Home Page: Off (no refresh); 1, 5, 15, or 30 minutes; 1 hour. The default is Off (no refresh).

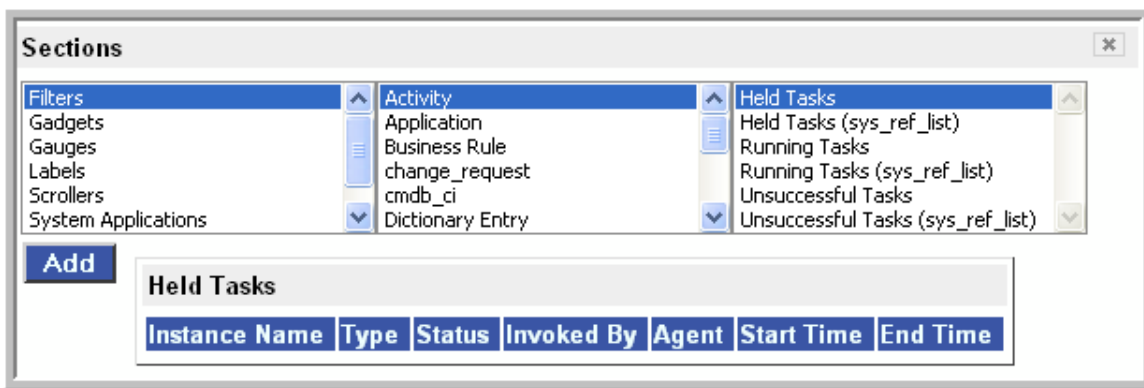
To manually refresh the Home Page, click **Refresh**.

To refresh a specific widget on the Home Page, click the widget's Refresh  icon.

Adding Widgets

Most widgets are available to add to your Home Page. However, some widgets listed in the **Sections** dialog, below, refer to program internals and are not applicable to your installation. These widgets are omitted from the [Widget Descriptions](#) table.

Step 1 At the top left corner of the Home Page, click the **Add Content** link. The **Sections** dialog displays.



The **Sections** dialog displays three columns of information:

- First column: List of widget types.
- Second column: List of sub-types for the selected widget type.
- Third column: List of specific widgets for the selected widget type and sub-type.

Step 2 Using the descriptions provided in the [Widget Descriptions](#) table, below, make a selection in each column to define the widget you want to add.

Step 3 Click **Add**. The content you specified is added as a new widget on your Home Page.

Widget Descriptions

The following table provides a description for all [available widgets](#) that are applicable to an Opwise installation.

First Column	Second Column	Third Column
<p>Filters – A filter is a list of records whose content is defined by the filter.</p>	<p>List of filter types. The content of this list is defined by users. When you create and save a filter on any Opwise list, such as Triggers or Tasks, the type of filter is added to this dialog.</p> <p>For example, if you create and save a filter on the Task list called Windows Tasks, the second column displays Tasks and the third column displays Windows Tasks. There also may be some items in this list that are created by the interface platform and are not of interest.</p>	<p>List of filters for the type selected in the second column.</p>
<p>Gadgets – A gadget is a hard-coded feature. Two types are currently supported: sticky notes and system information.</p>	<p>Sticky Note. Allows users to type informational messages into the home page.</p> <p>System Information. The Overview option provides information about the Opwise system, such as build date, active sessions, number of transactions, and so on.</p>	
<p>Gauges – A gauge is a "live" report using data from an Opwise Report table. See Opwise Automation Center Reports for instructions about how to create a new report and package it into a gauge, or how to package an existing report into a gauge.</p>	<p>List of gauge types that have been defined. The gauge type corresponds to the table that was selected when the report was defined, such as Activity or Audit Records.</p>	<p>List of reports that have been defined using the table (gauge type) shown in the second column. For example, a number of Activity reports have been defined, such as "Active Task Instances By Status." Those reports display in this list.</p>
<p>Labels – Labels allow you to customize the navigation pane with business-centric options. For example, you might want to add the option "Fee-Related Tasks" to the navigation pane, then add your fee-related tasks to that menu option. For more information, see Customizing the Navigation Pane.</p>	<p>All labels that have been added to the navigation pane.</p>	<p>Records that have been assigned to the label.</p>
<p>Scrollers</p>	<p>Not implemented.</p>	
<p>System Applications</p>	<p>Allows you to insert links to some sections of the navigation pane.</p>	
<p>World Clocks</p>	<p>Clocks showing current time in Los Angeles, New York, London, and Kiev.</p>	

Using the Dashboard

- Overview
- Customizing the Dashboard
- Refreshing Dashboard Data
 - Manually Refreshing Dashboard Data
 - Setting the Dashboard Refresh Rate

Overview

The dashboard lets you to set up a display of information that users commonly refer to throughout the day. This information is extracted from the database and displayed as [gauges](#).


To access the dashboard, click **Automation Center > Dashboard** from the navigation pane.

Customizing the Dashboard

The dashboard provides slots for up to nine gauges. Opswise is distributed with a set of default gauges and a default dashboard configuration.

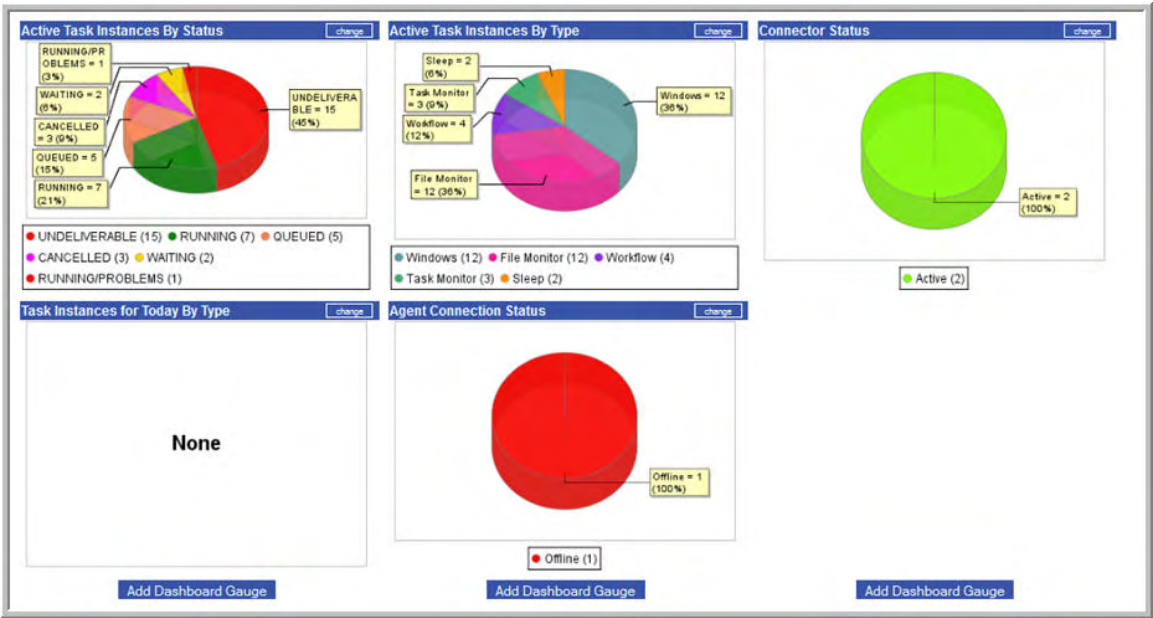
You can change any of the displayed gauges to another gauge or add additional gauges.

If you want to remove a gauge from the dashboard, you must delete it from the [Gauges list screen](#).

 **Warning**
 Deleting a gauge from the Gauges list screen will remove the gauge from the dashboard *and the home page* if it also is displayed there.

Step 1 If the gauge does not yet exist, follow the instructions in [Creating and Deleting Gauges](#) to create the new gauge.

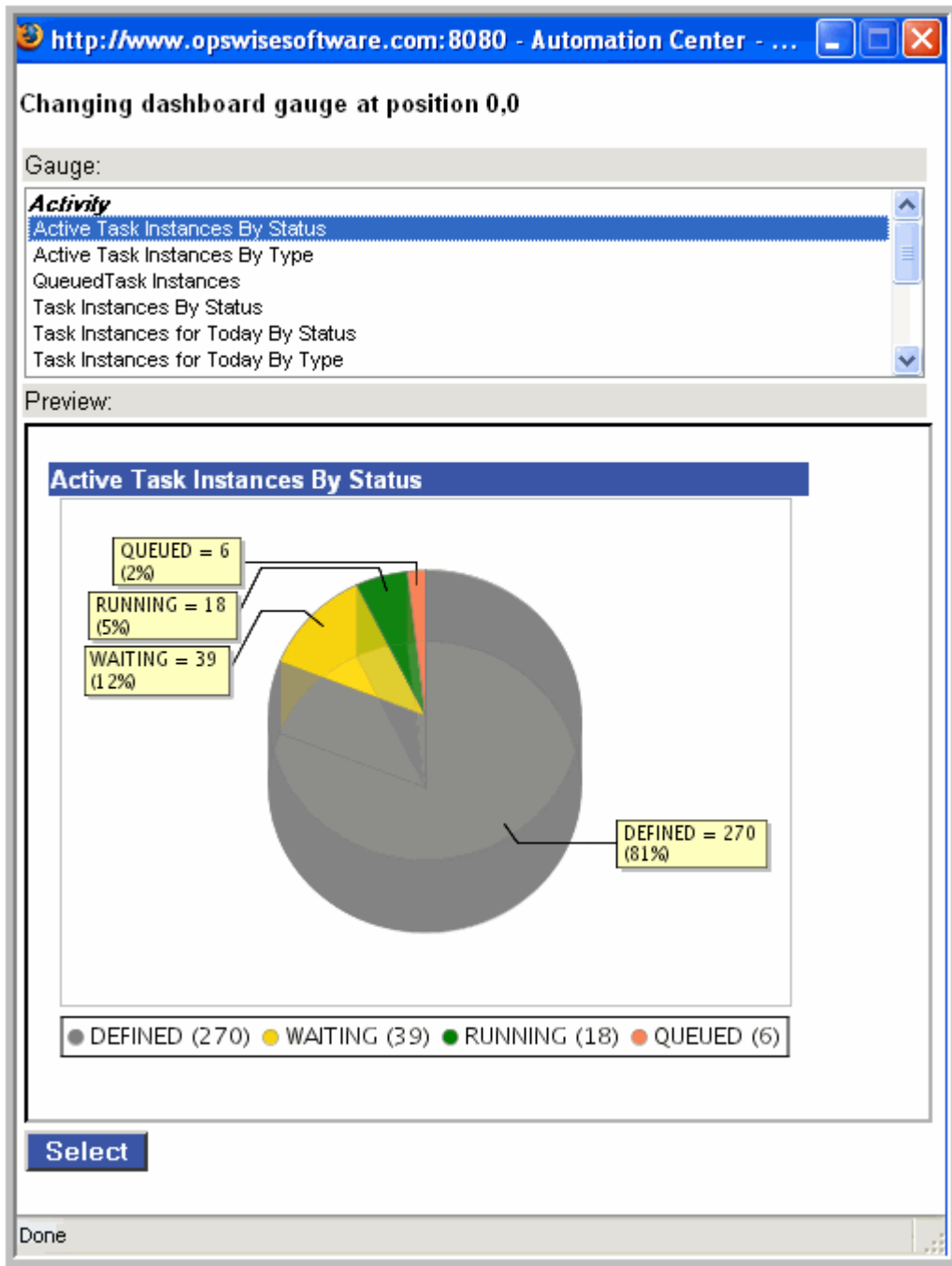
Step 2 Select **Automation Center > Dashboard** from the navigation pane. A default dashboard displays, as shown in the example below.



The dashboard screenshot shows the following data for each gauge:

- Active Task Instances By Status:**
 - UNDELIVERABLE = 15 (45%)
 - RUNNING = 7 (21%)
 - QUEUED = 5 (15%)
 - CANCELLED = 3 (9%)
 - WAITING = 2 (6%)
 - RUNNING/PROBLEMS = 1 (3%)
- Active Task Instances By Type:**
 - Windows = 12 (36%)
 - File Monitor = 12 (36%)
 - Workflow = 4 (12%)
 - Task Monitor = 3 (9%)
 - Sleep = 2 (6%)
- Connector Status:**
 - Active = 2 (100%)
- Task Instances for Today By Type:**
 - None
- Agent Connection Status:**
 - Offline = 1 (100%)

Step 3 To add a gauge to an empty slot, click **Add Dashboard Gauge**. Or, to replace a gauge with a different one, click the **change** button on the gauge you want to change. Opwise displays a list of available gauges.

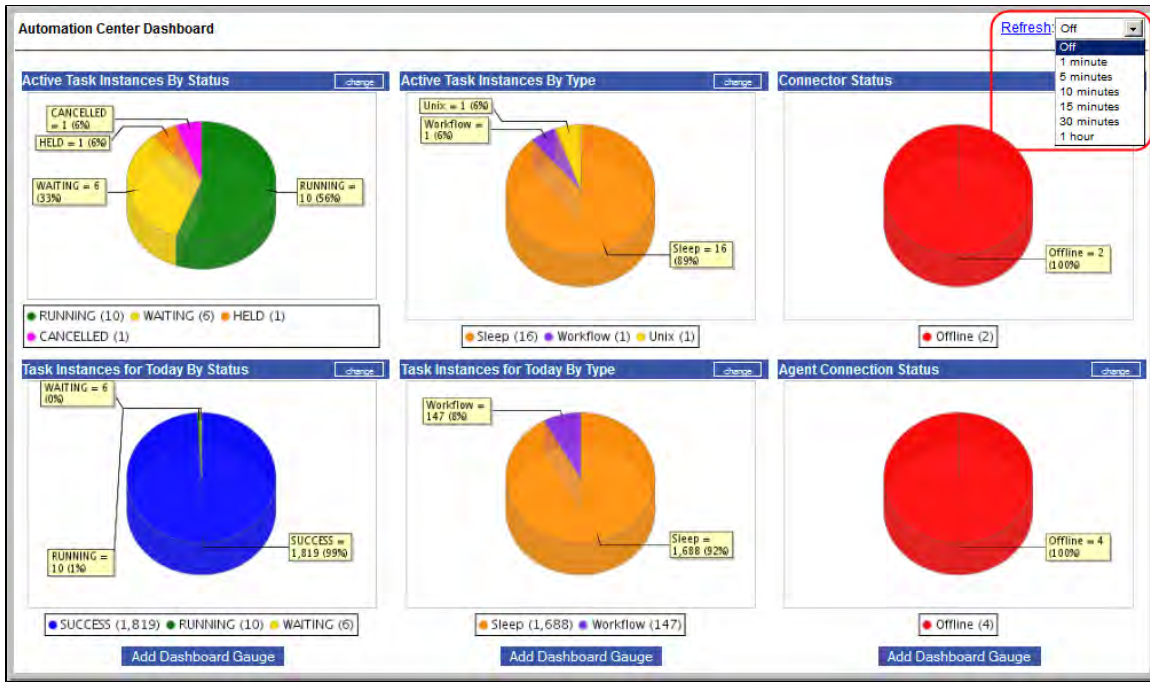


Step 4 Select the new gauge from the list. Opwise replaces the old gauge with the new one you selected.

Step 5 Repeat the above procedure for any other changes you want to make.

Refreshing Dashboard Data

You can manually refresh the data displayed on your dashboard and set an automatic refresh rate.



Manually Refreshing Dashboard Data

Click **Refresh**. Opwise repaints the screen with the current data.

Setting the Dashboard Refresh Rate

Step 1	Click the down arrow to display refresh rate options.
Step 2	Select a refresh rate: Off; 1, 5, 10, 15, or 30 minutes; 1 hour.

Creating and Deleting Gauges

- Overview
- Creating Gauges
 - Creating a Gauge via the Gauge Definition Screen
 - Gauge Definition Screen Field Definitions
 - Creating a Gauge via the Reports Screen
- Deleting Gauges

Overview

A gauge is a "live" report, using data from an [Opswise Report table](#), that can be displayed on your [home page](#) or [dashboard](#). Gauges are updated automatically according to the refresh setting on each of those pages.

When you create a gauge, it does not automatically display on your home page and/or the dashboard. You must choose to display the gauge from each of those pages.

However, if you delete a gauge that was displayed on your home page and/or the dashboard, it is automatically removed from those pages.

Creating Gauges

You can create a gauge either of two ways:

- Via the [Gauge Definition screen](#).
- Via the [Reports screen](#).

Creating a Gauge via the Gauge Definition Screen

To add a gauge via the Gauges definition screen:

Step 1 From the navigation pane, select **Automation Center Administration > Configuration > Gauges**. The Gauges list screen then displays, which lists of all currently defined gauges, whether they were created via the Gauge definition screen or via the [Reports](#) screen.

Name	Field	Query	Tab
Active Task Instances By Status	status_code	status_code<180^EQ	ops_e:
Active Task Instances By Type	type	status_code<180^EQ	ops_e:
Activity Count	status_code		ops_e:
Agent Connection Status	mode		ops_a:
Application Status			ops_a:
Cluster Node Status	node_mode		ops_cl:
Connector Status	mode		ops_co:
Task Instances for Today By Status	status_code	sys_created_onONToday@javascript.gs.daysAgoStart(0)@javascript.gs.daysAgoEnd(0)^EQ	ops_e:
Task Instances for Today By Type	type	sys_created_onONToday@javascript.gs.daysAgoStart(0)@javascript.gs.daysAgoEnd(0)^EQ	ops_e:
Unsuccessful Task Instances			ops_e:
User Summary Counts			sys_us:

Step 2 Click **New**. The Gauge definition screen displays.

Step 3 Using the field descriptions provided below as a guide, complete the fields as needed.

Step 4 Click **Submit** to save the gauge. Opswise creates a entry for the gauge on the Gauges List screen.

Gauge Definition Screen Field Definitions

Field Name	Description
Name	User-defined. Name used within Opwise to identify this gauge. It can contain a maximum of 40 alphanumeric characters. It is the user's responsibility to develop a workable naming scheme for gauges.
Type	Type of gauge.
Table	Opwise Report table from which the information in this gauge is provided.
View	
Aggregate	Type of aggregate shown in this gauge: Average, Count, Median, or Sum.
Field	
Group field	
Order	
Upper limit	
Lower limit	
Title	Title displayed on top of the gauge.
Query	
Url	
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Try it button	Displays the gauge as defined by the current field values on the screen.
Delete button	Deletes the current record.
Count Gauges tab	

Creating a Gauge via the Reports Screen

You can create a gauge from the Reports screen either by:

- Creating a gauge from an existing report.
- Creating a new report, and then creating a gauge for that report.

See [Creating a Gauge from a Report](#) for detailed information.

Deleting Gauges

To delete a gauge (and automatically remove it from your home page and/or dashboard, if it is being displayed on either of those pages):

Step 1 From the navigation pane, select **Automation Center Administration > Configuration > Gauges**. The Gauges List screen displays a list of all currently defined gauges.

Name	Field	Query	Tab
Active Task Instances By Status	status_code	status_code<180^EQ	ops_e:
Active Task Instances By Type	type	status_code<180^EQ	ops_e:
Activity Count	status_code		ops_e:
Agent Connection Status	mode		ops_a:
Application Status			ops_a:
Cluster Node Status	node_mode		ops_cl:
Connector Status	mode		ops_co:
Task Instances for Today By Status	status_code	sys_created_onONToday@javascript.gs.daysAgoStart(0)@javascript.gs.daysAgoEnd(0)^EQ	ops_e:
Task Instances for Today By Type	type	sys_created_onONToday@javascript.gs.daysAgoStart(0)@javascript.gs.daysAgoEnd(0)^EQ	ops_e:
Unsuccessful Task Instances			ops_e:
User Summary Counts			sys_us:

Step 2 Either:

1. Select the gauge you want to delete.
2. Click **Delete** on the **Actions on selected rows ...** menu. See [Deleting Records](#) for detailed instructions.

OR

1. Click the **Name** of the Gauge that you want to delete. The Gauge Definition screen for that Gauge displays.
2. Click the **Delete** button.

Resources



Overview

[*Resources Overview*](#)



Agents and Connectors

[*Introduction*](#)

[*Displaying Information About Agents and Connectors*](#)

[*Starting/Stopping Agents and Connectors*](#)

[*Sending Status Notifications on Opwise Component Status*](#)

[*Linux Unix Agent*](#)

[*Windows Agent*](#)

[*z/OS Agent*](#)

[*Indesca Agent*](#)

[*Agent Clusters*](#)



Other Resources

[*Cluster Nodes*](#)

[*Virtual Resources*](#)

[*Script Library*](#)

[*Email Template*](#)

[*Email Connection*](#)

[*Database Connection*](#)

[*SAP Connection*](#)

[*SNMP Manager*](#)

[*Applications*](#)



The information on these pages also is located in the [Opwise Automation Center 5.1.1 User Guide.pdf](#).

Resources Overview

Opswise resources are records that both define your Opswise Automation Center system and that you set up to help facilitate operations:

- **Agents and Connectors:** Opswise Agents and the Message Hub and Transporter components of the Opswise Outboard
 - Linux/Unix
 - Windows
 - z/OS
 - Indesca/Infitrans
- **Agent Clusters:** Groups of agents from which Opswise uses pre-defined parameters to select the most appropriate agent for a task.
- **Cluster Nodes:** Provide the business logic of Opswise.
- **Virtual Resources:** Allow you to create throttling schemes for tasks.
- **Script Library:** Allows you to execute scripts stored in the Opswise database.
- **Email Template:** Allows you to construct information that can be copied to create [Email tasks](#).
- **Email Connection:** Provides email server information required for Opswise to send an email.
- **Database Connection:** Provides database server information required for running [SQL tasks](#) and [Stored Procedure tasks](#).
- **SAP Connection:** Provides SAP server information required for running [SAP tasks](#).
- **SNMP Manager:** Allows you to generate SNMP notifications.
- **Applications:** Define the names of the applications being monitored.

Agents and Connectors

- Introduction
- Displaying Information About Agents and Connectors
 - Displaying Agent Information
 - Displaying Connector Information
- Starting/Stopping Agents and Connectors
- Suspending Agents, Agent Clusters, and Agent Cluster Memberships
 - Suspending an Agent
 - Suspending an Agent Cluster
 - Suspending an Agent Cluster Membership
- Resetting the Current Task Count
- Sending Notifications on Opwise Component Status
 - Email Notifications
 - SNMP Notifications

Introduction

Opwise Agent and Connector resources refer to the Agent, running and launching programs on one or more remote machines, along with the Opwise Connectors: the Message Hub and Transporter components of the Opwise Outboard. (See [Opwise Automation Center System Overview](#) for a diagram and description of the entire Opwise Automation Center system.)

The Connectors must be up and running in order for you to run tasks on an Agent.

Displaying Information About Agents and Connectors

When you start an Agent or a Connector for the first time, Opwise Controller automatically creates a database record for that Agent or Connector. You can view these records for details and status information.



Note

You also can view status information about Agents and Connectors from the [Command Line Interface \(CLI\)](#).

Displaying Agent Information

Step 1

From the navigation pane, select **Automation Center Resources > All Agents** (or select the specific type of agent). The Agents List screen displays:

Agent Name	Host Name	Type	Agent Id	Version	Last Heartbeat	Current Task Count	Suspended	Status
serverhost - AGNT0001	serverhost	Linux/Unix	AGNT0001	1.7.0	2013-04-17 07:23:57 -0700	0	false	✓
serverhost - AGNT0002	serverhost	Linux/Unix	AGNT0002	1.7.0	2013-04-17 07:23:57 -0700	0	false	✓
serverhost - AGNT0003	serverhost	Linux/Unix	AGNT0003	1.7.0	2013-04-17 07:23:57 -0700	0	false	✓
serverhost - AGNT0004	serverhost	Linux/Unix	AGNT0004	1.7.0	2013-04-17 07:23:57 -0700	0	false	✓

Step 2 To display more details about an agent on the list, click the Agent Name. The Agent Definition screen for that agent displays (this example is for a Linux Unix Agent).

Most fields are display-only; however, you can make the following changes:

1. Add a [Member of Business Services](#).
2. Assign [Credentials](#).
3. Change the [heartbeat interval](#). The heartbeat is a status message sent from the Agent to the Controller.
4. Change the default [Log Level](#).
5. Select whether or not to apply a [Task Execution Limit](#) (and [Limit Amount](#)) on the Agent.

You also can choose to:

- Temporarily [suspend](#) the agent's ability to run tasks.
- [Reset](#) the Current Task Count.

Agents List Screen Field Descriptions

The following table describes the default display columns on the Agents List screen.

Column	Description
Agent Name	Required. Defined by the user when installing the agent. This is the name used within Opwise Automation Center to identify this resource.
Host Name	Specified by the user during installation. The IP address or domain/name of the host machine where the component resides.
Type	Agent's platform: Linux/Unix, Windows, z/OS, or Indesca.
Agent ID	Unique ID for this agent, created during installation. The name of the communications server message queue. Opwise Automation Center uses the communications server for communications between the Controller scheduler and Opwise Automation Center agents.
Version	System-supplied. Version number of the agent program.
Last Heartbeat	System-supplied. The date and time the most recent heartbeat was received from the component.

Current Task Count	System supplied; current number of tasks currently being run by this Agent.
Suspended	Specification (true or false) for whether or not this Agent has been suspended from the ability to run tasks.
Status	System-supplied. The status of the agent. The green button with the checkmark means the agent is running. A red circle with an X means the agent is not running.

Agent Definition Screen Field Definitions

For detailed descriptions of the fields on the Agent Definition screen for each type of agent, click the appropriate link below:

- [Linux/Unix Agent](#)
- [Windows Agent](#)
- [z/OS Agent](#)
- [Indesca Agent](#)

Displaying Connector Information

From the navigation pane, select **Automation Center Resources > Connectors**. The Connectors List screen displays.

Connector Name	Host Name	Type	Queue	Version	Last Heartbeat	Status
Msghub - serverhost - HUB01	serverhost	Msghub	HUB01	5.1.0	2013-04-18 13:17:57 -0700	✓
Transport - serverhost - TP01	serverhost	Transport	TP01	5.1.0	2013-04-18 13:17:24 -0700	✓

Normally, you would only consult this display if you are experiencing connection problems with your Agent.

Connectors List Screen Field Descriptions

The following table below describes the fields on the Connectors List screen.

Field Name	Description
Connector Name	System-supplied name for this connector.
Host Name	Specified by the user during installation. The IP address or domain/name of the host machine where the component resides.
Type	Either msghub or transport .
Queue	System-supplied. Internal use only.
Version	System-supplied. Version number of the agent program.
Last Heartbeat	System-supplied. The date and time the most recent heartbeat was received from the component.

Status	System-supplied. The status of the component. The green button with the checkmark means the component is active. A red circle with an X means the component is not active.
--------	--

Starting/Stopping Agents and Connectors

For instructions on starting and stopping Agents and Connectors, see [Starting-Stopping Opwise Components](#) and select your platform.

Suspending Agents, Agent Clusters, and Agent Cluster Memberships

If an Agent or Agent Cluster reaches its Task Execution Limit, all new work queued against that Agent or Agent Cluster will transition into the [Execution Wait](#) status until the Current Task Count falls below the Limit Amount.

You also can manually suspend (and resume) Agents and Agent Clusters, as well as Agent memberships in Agent Clusters.



Note

The following [roles and permissions](#) are required to suspend/resume Agents, Agent Clusters, and Agent Cluster Memberships:

- Agent Suspend/Resume requires the **ops_admin role** and the appropriate [Agent permissions](#) for Agent Suspend/Resume commands.
- Agent Cluster Suspend/Resume and Agent Cluster Membership Suspend/Resume require the **ops_agent_cluster_admin** role.

Suspending an Agent

You can temporarily suspend an Agent's ability to run tasks from the Agent Lists screen or an Agent Definition screen. Any tasks queued against a suspended Agent will transition into Execution Wait status until the Agent has been resumed.

- To suspend an Agent from the [Agents List screen](#), either:
 - Right-click the **Agent Name** of the agent to be suspended and click **Suspend Agent** on the pop-up menu.
 - Click the box to the left of the **Agent Name** and, from the **Action on selected rows...** drop-down list at the bottom of the page, click **Suspend Agent**.
- To suspend an Agent from an [Agent Definition screen](#), click the **Suspend Agent** button. A **Resume Agent** button replaces the **Suspend Agent** button.

Resuming an Agent

(To end the suspension, and resume an Agent's ability to run tasks, either:

- Click **Resume Agent** on the pop-up menu or from the **Action on selected rows...** drop-down list.
- Click the **Resume Agent** button.

Suspending an Agent Cluster

You can temporarily suspend a cluster of Agents' ability to run tasks from the Agent Clusters List screen or an Agent Cluster Definition screen. Any tasks queued against a suspended agent cluster will transition into Execution Wait status until the agent cluster has been resumed.

- To suspend an Agent Cluster from the [Agent Clusters List screen](#), either:
 - Right-click the **Cluster Name** of the agent cluster to be suspended and click **Suspend Agent Cluster** on the pop-up menu.
 - Click the box to the left of the agent cluster. From the **Action on selected rows...** drop-down list at the bottom of the page, click **Suspend Agent**.
- To suspend an Agent Cluster from an [Agent Cluster Definition screen](#), click the **Suspend Cluster** button. A **Resume Cluster** button replaces the *Suspend Cluster * button.

Resuming an Agent Cluster

To end the suspension, and resume a cluster of Agents' ability to run tasks, either:

- Click **Resume Agent Cluster** on the pop-up menu or from the **Action on selected rows...** drop-down list.
- Click the **Resume Cluster** button.

Suspending an Agent Cluster Membership

You can temporarily suspend an Agent's membership in an agent cluster from an Agent Cluster Definition screen. Suspending an Agent's membership in an agent cluster is equivalent to removing the Agent from the agent cluster, except it is meant to be temporary. The Agent will not be available as a selection from the agent cluster when a task is queued against the agent cluster until the agent's membership has been resumed.



Note

If a task specifies both an Agent and an agent cluster in which that Agent is a member, and the specified Agent has been suspended from the agent cluster, the Agent still has the ability to run the task. Directly specifying an Agent overrides its suspension from an agent cluster.

To suspend an Agent's membership from the Agent Cluster Definition screen, click the [Agents in Cluster](#) tab and then either:

- Right-click an **Agent** on the list and then click **Suspend Cluster Membership** on the pop-up menu.
- Click the box to the left of an **Agent** and then, from the **Action on selected rows...** drop-down list at the bottom of the page, click **Suspend Cluster Membership**.

Resuming an Agent Cluster Membership

To end the suspension, and resume an Agent's membership in an agent cluster:

- Click **Resume Cluster Membership** on the pop-up menu or from the **Action on selected rows...** drop-down list.

Resetting the Current Task Count

The Current Task Count field on the [Agent Definition](#) screen and the [Agent Cluster Definition](#) screen identifies the current number of tasks currently being run by, respectively, that Agent or Agent Cluster.

If there is a limit to the number of tasks that an Agent or Agent Cluster can run concurrently (as specified by the **Task Execution Limit** and **Limit Amount** fields), you can reset the current task count to 0. This can help avoid a situation where the Controller believes the Agent to be running more tasks than it actually is running, and therefore might impose the task limit on the Agent unnecessarily.

To reset the Current Task Count field, hover your cursor over the down arrow on the [Agent Definition](#) screen or the [Agent Cluster Definition](#) screen title bar, or right-click the title bar, and then click, respectively, **Reset Agent Task Count** or **Reset Cluster Task Count**.



Note


The following [roles and permissions](#) are required to reset the current task count:

- Reset Agent Task Count requires the **ops_admin** role and the [Update Agent](#) permission.
- Reset Cluster Task Count requires the **ops_agent_cluster_admin** role.

Sending Notifications on Opwise Component Status

You can configure Opwise components (Agents, Connectors, and Cluster Nodes) to send a notification via email or SNMP when the component goes down (Offline) or comes back up (Active).

Step 1	From the navigation pane, select Automation Center Resources > [Opwise component] . The component list screen for the selected component displays.
Step 2	Click a <component> Name to display the component definition screen for that component.
Step 3	Click the Connector Notifications tab to display a list of all notifications configured for the resource.
Step 4	Click New . The Connector Notifications Wizard screen displays.
Step 5	Select the type of notification you want to configure: Email Notification or SNMP Notification .

Step 6	Complete the fields as needed (see the field descriptions, below). <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note Built-in variables are available to pass data about the Agent or Connector into the notification (see Agent Variables or Connector Variables, as appropriate).</p> </div>
Step 7	Click the Submit button to save the record.
Step 8	If appropriate, repeat these steps for any additional notifications you want to add.

Email Notifications

Shown below is the Email Notification screen that displays for both Agents and Connectors.

Email Notification Screen Field Descriptions

The following table describes the fields and buttons on the Email Notification screen.

Field Name	Description
Mode	<ul style="list-style-type: none"> Offline = Trigger the notification when the component goes offline. Active = Trigger the notification when the component comes up.
Email Template	Optional. The name of the Email template defined using the Email template screen. The Email template allows you to specify standard recipients and text for outgoing emails. Type in a name, or click the magnifying glass to browse to an existing Email template or create a new one. You must specify either an Email template or Email connection, or both. If you specify both, the Email server specified in the Email Connection record overrides the server in the template.

Email Connection	Required. Name of the Email connection defined using the Email connection screen. The email connection specifies information about the email server. You can also specify the Email connection in the Email template (see above). You must specify an Email template and/or an Email connection. If you specify an Email template and an Email connection, the server selected in the Email connection overrides the server selected in the Email template. Type in a name, click the magnifying glass to browse for an existing Email server definition, or create a new one.
Reply-To	Required. Specifies the email address of the sender. Use commas to separate multiple recipients. Variables supported.
To	Required. Specifies the email address of the recipient. Use commas to separate multiple recipients. Variables supported.
CC	Optional. Specifies the email address of the party being sent a copy of the email, if any. Use commas to separate multiple recipients. Variables supported.
BCC	Optional. Specifies the email address of the party being sent a blind (hidden) copy of the email, if any. Use commas to separate multiple recipients. Variables supported.
Subject	Optional. Specifies the subject line of the email. Variables supported.
Body	Optional. Contains the text of the email message. Variables supported. If both the email template and the email task contain text in the body, the text is appended. Specific built-in variables are available for passing information about the Agent or Connectors. You must use the appropriate variables for each component type; that is, use <code>ops_agent</code> variables for Agent notifications and <code>ops_connector</code> variables for Connector notifications.
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Delete button	Deletes the current record.

SNMP Notifications

Shown below is the SNMP Notification screen that displays for both Agents and Connectors.

SNMP Notification Screen Field Descriptions

The following table describes the fields and buttons on the SNMP Notification screen.

Field Name	Description
------------	-------------

Mode	<ul style="list-style-type: none">• Offline = Trigger the notification when the component goes offline.• Active = Trigger the notification when the component comes up.
SNMP Manager	The SNMP Manager that will receive the SNMP notification.
Notification Severity	Optional. Informational only. Indicates the severity of this notification. Options: Normal, Warning, Minor, Major Critical.
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Delete button	Deletes the current record.

Linux Unix Agent

- [Overview](#)
- [Linux/Unix Agent Definition](#)
- [Linux/Unix Agent Definition Field Descriptions](#)

Overview

The Linux/Unix Agent resource provides information about a Linux or Unix Agent running on a Linux or Unix platform. To run a Linux/Unix task, you need an Opwise Linux/Unix Agent installed and running on the target machine.

Linux/Unix Agent Definition

The Linux/Unix Agent definition provides the information necessary for the scheduler to locate and communicate with the machine where the Opwise Agent resides. Opwise creates this record automatically when the Agent connects with the Controller.

To view a Linux/Unix Agent definition:

Step 1 From the navigation pane, select **Automation Center Resources > Linux/Unix Agents**. The Linux/Unix Agents list screen displays a list of connected Linux/Unix Agents.



Note

You also can select **Automation Center Resources > All Agents** from the navigation pane to display a list of all agents: Linux/Unix, Windows, z/OS, and Indesca.

Step 2 Select a Linux/Unix agent from the list. The Linux/Unix Agent definition screen displays.

Linux/Unix Agent		Agents In Cluster		Connector Notifications		Task Instances	
Linux/Unix Agent - Required field							
Agent Name:	serverhost - AGNT0001	Credentials:					
Agent Id:	AGNT0001	Version:	1.7.0				
Member of Business Services:	🔒	Build Id:	build.50				
Host Name:	serverhost	Build Date:	20110717221446				
IP Address:	192.168.50.50	Operating System:	Linux				
PID:	9978	Operating System Release:	i686 2.6.9-67.0.20.ELsmp (#1 SMP Thu Ju				
Status:	Active	CPU:	GenuineIntel Intel(R) Core(TM)2 Duo CPU				
Last Heartbeat:	2013-04-18 12:27:39 -0700	CPU Load:	4				
Heartbeat Interval:	120	Started Date:					
Log Level:	Informational	Jobs Run:	2				
Task Execution Limit:	Unlimited	Suspended:	<input type="checkbox"/>				
Current Task Count:	4						

Step 3 Most fields are display-only; however, you can make the following changes:

1. Add a [Member of Business Services](#).
2. Assign [Credentials](#).
3. Change the [heartbeat interval](#). The heartbeat is a status message sent from the Agent to the Controller.
4. Change the [Log Level](#) (default is Informational).
5. Select whether or not to apply a [Task Execution Limit](#) (and [Limit Amount](#)) on the Agent.

You also can choose to:

- Temporarily [suspend](#) the agent's ability to run tasks.
- [Reset](#) the Current Task Count.

See the field descriptions, below, for details about all fields on this Agent definition screen.

Linux/Unix Agent Definition Field Descriptions

The following table describes the fields, buttons, and tabs on the Linux/Unix Agent definition screen.

Field Name	Description
Agent Name	Required. Defined by the user when installing the agent. This is the name used within Opswise Automation Center to identify this resource.
Credentials	Credentials under which this Agent runs tasks. These credentials are overridden by any credentials provided in the task definition for any tasks being run by this Agent.
Agent ID	Unique ID for this agent, created during installation. The name of the communications server message queue. Opswise Automation Center uses the communications server for communications between the Controller scheduler and Opswise Automation Center agents.
Version	System-supplied. Version number of the agent program.
Member of Business Service	Optional. User-defined at installation. Allows you to specify one or more Business Services that this resource definition belongs to.
Build ID	System-supplied, provided by the agent. The build ID of the Opswise Automation Center agent. Internal use only.
Host Name	Specified by the user during installation. The IP address or domain/name of the host machine where the component resides.
Build Date	System-supplied, provided by the agent. The date the agent program was last built.
IP Address	Provided by the user during installation. The TCP/IP address of the machine where the agent is running.
Operating System	System-supplied. The operating system where the agent is running.
PID	System-supplied, provided by the agent. Process ID of the agent.
Operating System Release	System-supplied. Release information for the operating system where the agent is running.
Status	System-supplied. The status of the agent.
CPU	System-supplied. Information about the CPU on the agent machine.
Last Heartbeat	System-supplied. The date and time the most recent heartbeat was received from the component.

CPU Load	System-supplied. The current CPU load on the agent machine, expressed as a percentage. For example, 1 means 1% currently utilized.
Heartbeat Interval	User-modifiable. The heartbeat interval in seconds. The heartbeat is a status message sent from the agent to the core processor.
Started Date	System-supplied. The date/timestamp when the agent was last started.
Log Level	User-modifiable. The level of logging the agent should perform. Options: <ul style="list-style-type: none"> • Severe Error • Errors • Warning • Informational • Debug • Trace
Jobs Run	Total number of jobs that have been run through the Controller to this Agent.
Task Execution Limit	Specification for whether a limited or unlimited number of task instances can be run concurrently on the Agent. (Default is unlimited .) For purposes of imposing this task execution limit, running task instances are those in any of these statuses: Cancel Pending, Queued, Received, Running, Submitted, and Started.
Limit Amount	If Task Execution Limit = Limited; number of tasks that can be running at the same time by the agent.
Current Task Count	Current number of tasks currently being run by this Agent. (See Resetting the Current Task Count for information on resetting the current task count.)
Suspended	Indication that the agent's ability to run tasks has been suspended.
Update button	Saves updates to the record.
Suspend Agent button	Suspend the agent's ability to run tasks.
Resume Agent button	Resume the suspended agent's ability to run tasks.
Delete button	Deletes the current record.
Agents in Cluster tab	Provides a list of agent clusters that this agent belongs to, if any. See Agent Clusters .
Connector Notifications tab	System-supplied. Displays a list of connector notifications that have been defined for this component.

**Task
Instances**
tab

System-supplied. Displays a list of all instances that have run or are ready to run on this agent since it last started.

Windows Agent

- [Overview](#)
- [Windows Agent Definition](#)
- [Windows Agent Definition Field Descriptions](#)

Overview

The Windows Agent resource provides information about a Windows Agent running on a Windows platform. To run a Windows task, you need a Windows Agent installed and running on the target machine.

Windows Agent Definition

The Windows Agent definition provides the information necessary for the scheduler to locate and communicate with the machine where the Opwise Agent resides. Opwise creates this record automatically when the Agent connects with the Controller.

To view a Windows Agent definition:

Step 1 From the navigation pane, select **Automation Center Resources > Windows Agents**. The Windows Agents list screen displays a list of connected Windows Agents.



Note

You also can select **Automation Center Resources > All Agents** from the navigation pane to display a list of all agents: Linux/Unix, Windows, z/OS, and Indesca.

Step 2 Select a Windows agent from the list. The Windows Agent definition screen displays.

Windows Agent		Agents In Cluster		Connector Notifications		Task Instances	
Windows Agent = Required field							
Agent Name:	qa-w2k3 - W2K3	Credentials:		Update	Suspend Agent	Delete	↑ ↓
Agent Id:	W2K3	Version:	5.1.0.15				
Member of Business Services:	🔒	Build Id:	141				
Host Name:	qa-w2k3	Build Date:	20130415111431				
IP Address:	192.168.50.50	Operating System:	Microsoft Windows				
PID:	1156	Operating System Release:	Server 2003 family, Standard Edition				
Status:	Active	CPU:	INTEL (2 CPU(s))				
Last Heartbeat:	2013-04-18 15:10:08 -0400	CPU Load:	0				
Heartbeat Interval:	120	Started Date:	2013-04-15 14:36:09 -0400				
Log Level:	Informational	Jobs Run:	0				
Task Execution Limit:	Limited	Limit Amount:					
Current Task Count:	0	Suspended:	<input type="checkbox"/>				
Update Suspend Agent Delete							

Step 3 Most fields are display-only; however, you can make the following changes:

1. Add a **Member of Business Services**.
2. Assign **Credentials**.
3. Change the **heartbeat interval**. The heartbeat is a status message sent from the Agent to the Controller.
4. Change the **Log Level** (default is Informational).
5. Select whether or not to apply a **Task Execution Limit** (and **Limit Amount**) on the Agent.

You also can choose to:

- Temporarily **suspend** the agent's ability to run tasks.
- **Reset** the Current Task Count.

See the field descriptions, below, for details about all fields on this Agent definition screen.

Windows Agent Definition Field Descriptions

The following table describes the fields, buttons, and tabs on the Windows Agent definition screen.

Field Name	Description
Agent Name	Required. Defined by the user when installing the agent. This is the name used within Opwise Automation Center to identify this resource.
Credentials	Credentials under which this Agent runs tasks. These credentials are overridden by any credentials provided in the task definition for any tasks being run by this Agent.
Agent ID	Unique ID for this agent, created during installation. The name of the communications server message queue. Opwise Automation Center uses the communications server for communications between the Controller scheduler and Opwise Automation Center agents.
Version	System-supplied. Version number of the agent program.
Member of Business Services	Optional. User-defined at installation. Allows you to specify one or more Business Services that this resource definition belongs to.
Build ID	System-supplied, provided by the agent. The build ID of the Opwise Automation Center agent. Internal use only.
Host Name	Specified by the user during installation. The IP address or domain/name of the host machine where the component resides.
Build Date	System-supplied, provided by the agent. The date the agent program was last built.
IP Address	Provided by the user during installation. The TCP/IP address of the machine where the agent is running.
Operating System	System-supplied. The operating system where the agent is running.
PID	System-supplied, provided by the agent. Process ID of the agent.
Operating System Release	System-supplied. Release information for the operating system where the agent is running.
Status	System-supplied. The status of the agent.
CPU	System-supplied. Information about the CPU on the agent machine.
Last Heartbeat	System-supplied. The date and time the most recent heartbeat was received from the component.

CPU Load	System-supplied. The current CPU load on the agent machine, expressed as a percentage. For example, 1 means 1% currently utilized.
Heartbeat Interval	User-modifiable. The heartbeat interval in seconds. The heartbeat is a status message sent from the agent to the core processor.
Started Date	System-supplied. The date/timestamp when the agent was last started.
Log Level	User-modifiable. The level of logging the agent should perform. Options: <ul style="list-style-type: none"> • Severe Error • Errors • Warning • Informational • Debug • Trace
Jobs Run	Total number of jobs that have been run through the Controller to this Agent.
Task Execution Limit	Specification for whether a limited or unlimited number of task instances can be run concurrently on the Agent. (Default is unlimited .) For purposes of imposing this task execution limit, running task instances are those in any of these statuses: Cancel Pending, Queued, Received, Running, Submitted, and Started.
Limit Amount	If Task Execution Limit = Limited; number of tasks that can be running at the same time by the agent.
Current Task Count	Current number of tasks currently being run by this Agent. (See Resetting the Current Task Count for information on resetting the current task count.)
Suspended	Indication that the agent's ability to run tasks has been suspended.
Update button	Saves updates to the record.
Suspend Agent button	Suspend the agent's ability to run tasks.
Resume Agent button	Resume the suspended agent's ability to run tasks.
Delete button	Deletes the current record.
Agents in Cluster tab	Provides a list of agent clusters that this agent belongs to, if any. See Agent Clusters .
Connector Notifications tab	System-supplied. Displays a list of connector notifications that have been defined for this component.

**Task
Instances**
tab

System-supplied. Displays a list of all instances that have run or are ready to run on this agent since it last started.

z/OS Agent

- Overview
- z/OS Agent Definition
- z/OS Agent Definition Field Descriptions

Overview

The z/OS Agent resource provides information about a z/OS Agent running on a z/OS platform. To run a z/OS task, you need an Opwise z/OS Agent installed and running on the target machine.

z/OS Agent Definition

The z/OS Agent definition provides the information necessary for the scheduler to locate and communicate with the machine where the Opwise Agent resides. Opwise creates this record automatically when the Agent connects with the Controller.

To view a z/OS Agent definition:

Step 1 From the navigation pane, select **Automation Center Resources > z/OS Agents**. The z/OS Agents list screen displays a list of connected z/OS Agents.



Note

You also can select **Automation Center Resources > All Agents** from the navigation pane to display a list of all agents: Linux/Unix, Windows, z/OS, and Indesca.

Step 2 Select a z/OS agent from the list. The z/OS Agent definition screen displays.

Step 3 Most fields are display-only; however, you can make the following changes:

1. Add a **Member of Business Services**.
2. Assign **Credentials**.
3. Change the **heartbeat interval**. The heartbeat is a status message sent from the Agent to the Controller.
4. Change the **Log Level** (default is Informational).
5. Select whether or not to apply a **Task Execution Limit** (and **Limit Amount**) on the Agent.

You also can choose to:

- Temporarily **suspend** the agent's ability to run tasks.
- **Reset** the Current Task Count.

See the field descriptions, below, for details about all fields on this Agent definition screen.

z/OS Agent Definition Field Descriptions

The following table describes the fields, buttons, and tabs on the z/OS Agent definition screen.

Field Name	Description
Agent Name	Required. Defined by the user when installing the agent. This is the name used within Opwise Automation Center to identify this resource.
Credentials	Credentials under which this Agent runs tasks. These credentials are overridden by any credentials provided in the task definition for any tasks being run by this Agent.
Agent ID	Unique ID for this agent, created during installation. The name of the communications server message queue. Opwise Automation Center uses the communications server for communications between the Controller scheduler and Opwise Automation Center agents.
Version	System-supplied. Version number of the agent program.
Member of Business Service	Optional. User-defined at installation. Allows you to specify one or more Business Services that this resource definition belongs to.
Build ID	System-supplied, provided by the agent. The build ID of the Opwise Automation Center agent. Internal use only.
Host Name	Specified by the user during installation. The IP address or domain/name of the host machine where the component resides.
Build Date	System-supplied, provided by the agent. The date the agent program was last built.
IP Address	Provided by the user during installation. The TCP/IP address of the machine where the agent is running.
Operating System	System-supplied. The operating system where the agent is running.
PID	System-supplied, provided by the agent. Process ID of the agent.
Operating System Release	System-supplied. Release information for the operating system where the agent is running.
Status	System-supplied. The status of the agent.
CPU	System-supplied. Information about the CPU on the agent machine.
Last Heartbeat	System-supplied. The date and time the most recent heartbeat was received from the component.

CPU Load	System-supplied. The current CPU load on the agent machine, expressed as a percentage. For example, 1 means 1% currently utilized.
Heartbeat Interval	User-modifiable. The heartbeat interval in seconds. The heartbeat is a status message sent from the agent to the core processor.
Started Date	System-supplied. The date/timestamp when the agent was last started.
Log Level	User-modifiable. The level of logging the agent should perform. Options: <ul style="list-style-type: none"> • Severe Error • Errors • Warning • Informational • Debug • Trace
Jobs Run	Total number of jobs that have been run through the Controller to this Agent.
Task Execution Limit	Specification for whether a limited or unlimited number of task instances can be run concurrently on the Agent. (Default is unlimited .) For purposes of imposing this task execution limit, running task instances are those in any of these statuses: Cancel Pending, Queued, Received, Running, Submitted, and Started.
Limit Amount	If Task Execution Limit = Limited; number of tasks that can be running at the same time by the agent.
Current Task Count	Current number of tasks currently being run by this Agent. (See Resetting the Current Task Count for information on resetting the current task count.)
Suspended	Indication that the agent's ability to run tasks has been suspended.
Update button	Saves updates to the record.
Suspend Agent button	Suspend the agent's ability to run tasks.
Resume Agent button	Resume the suspended agent's ability to run tasks.
Delete button	Deletes the current record.
Agents in Cluster tab	Provides a list of agent clusters that this agent belongs to, if any. See Agent Clusters .
Connector Notifications tab	System-supplied. Displays a list of connector notifications that have been defined for this component.

**Task
Instances**
tab

System-supplied. Displays a list of all instances that have run or are ready to run on this agent since it last started.

Indesca Agent

- [Overview](#)
- [Indesca Agent Definition](#)
- [Indesca Agent Definition Field Descriptions](#)

Overview

The Indesca Agent resource provides information about an Indesca Agent. To run an Indesca task, you need an Indesca Agent installed and running on the target machine.

Indesca Agent Definition

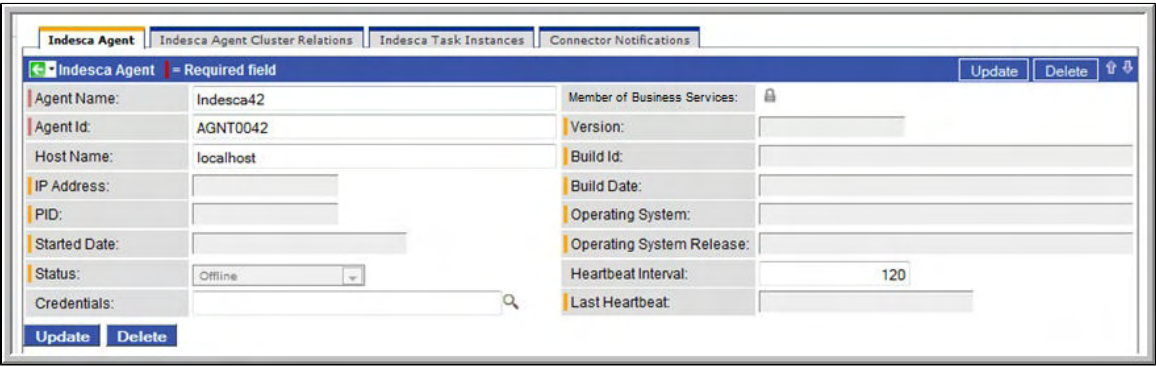
The Indesca Agent definition provides the information necessary for the scheduler to locate and communicate with the machine where the Opwise Agent resides. Opwise creates this record automatically when the Agent connects with the Controller.

To view an Indesca Agent definition:

Step 1 From the navigation pane, select **Automation Center Resources > Indesca/Infitrans Agents**. The Indesca Agents list screen displays a list of connected Indesca Agents.

Note
You also can select **Automation Center Resources > All Agents** from the navigation pane to display a list of all agents: Linux/Unix, Windows, z/OS, and Indesca.

Step 2 Select a Indesca agent from the list. The Indesca Agent definition screen displays.



Step 3 Most fields are display-only; however, you can make the following changes:

1. Add a [Member of Business Services](#).
2. Assign [Credentials](#).
3. Change the [heartbeat interval](#). The heartbeat is a status message sent from the Agent to the Controller.

See the field descriptions, below, for details about all fields on this Agent definition screen.

Indesca Agent Definition Field Descriptions

The following table describes the fields, buttons, and tabs on the Indesca Agent definition screen.

Field Name	Description
------------	-------------

Agent Name	Required. Defined by the user when installing the agent. This is the name used within Opwise Automation Center to identify this resource.
Member of Business Services	Optional. User-defined at installation. Allows you to specify one or more Business Services that this resource definition belongs to.
Agent ID	Unique ID for this agent, created during installation. The name of the communications server message queue. Opwise Automation Center uses the communications server for communications between the Controller scheduler and Opwise Automation Center agents.
Version	System-supplied. Version number of the agent program.
Host Name	Specified by the user during installation. The IP address or domain/name of the host machine where the component resides.
Build ID	System-supplied, provided by the agent. The build ID of the Opwise Automation Center agent. Internal use only.
IP Address	Provided by the user during installation. The TCP/IP address of the machine where the agent is running.
Build Date	System-supplied, provided by the agent. The date the agent program was last built.
PID	System-supplied, provided by the agent. Process ID of the agent.
Operating System	System-supplied. The operating system where the agent is running.
Started Date	System-supplied. The date/timestamp when the agent was last started.
Operating System Release	System-supplied. Release information for the operating system where the agent is running.
Status-Agent	System-supplied. The status of the agent.
Heartbeat Interval	User-modifiable. The heartbeat interval in seconds. The heartbeat is a status message sent from the agent to the core processor.
Credentials	Credentials under which this Agent runs tasks. These credentials are overridden by any credentials provided in the task definition for any tasks being run by this Agent.
Last Heartbeat	System-supplied. The date and time the most recent heartbeat was received from the component.
Update button	Saves updates to the record.
Delete button	Deletes the current record.

Agent Cluster Relations tab	Provides a list of agent clusters that this agent belongs to, if any. See Agent Clusters .
Task Instances tab	System-supplied. Displays a list of all instances that have run or are ready to run on this agent since it last started.
Connector Notifications tab	System-supplied. Displays a list of connector notifications that have been defined for this component.

Agent Clusters

- [Overview](#)
- [Creating a New Agent Cluster](#)
- [Agent Cluster Field Descriptions](#)
- [Assigning Agents to the Cluster](#)
- [Suspending Agent Clusters and Agent Cluster Memberships](#)

Overview

For Windows and Linux/Unix agents, Opswise allows you to create clusters (groups) of agents.


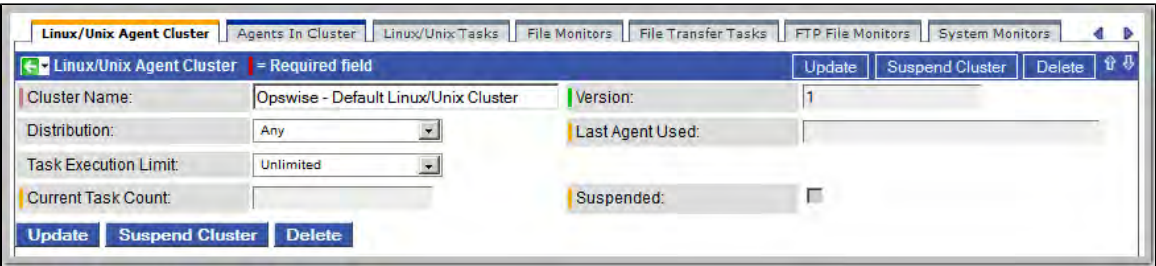
If you specify an agent cluster in a task, Opswise selects an agent from the cluster based on the selection method that you specified when you created the cluster. If you specify both an agent and an agent cluster in a task, Opswise first attempts to run the task on the agent; if the agent is unavailable, Opswise selects an agent from the agent cluster.



Note

The instructions and screens, below, for creating Windows Agent Clusters and Linux/Unix Agent Clusters, and assigning agents to those clusters, are the same.

Creating a New Agent Cluster

<p>Step 1</p>	<p>From the navigation pane, select (for example) Automation Center Resources > Linux/Unix Agent Clusters. The Linux/Unix Clusters List screen displays.</p> 
<p>Step 2</p>	<p>Click New. The Agent Cluster Definition screen displays.</p> 
<p>Step 3</p>	<p>Using the field descriptions provided below as a guide, complete the fields as needed.</p>
<p>Step 4</p>	<p>Right-click on the title bar and select Save to save the record and remain on the current display.</p>
<p>Step 5</p>	<p>To add Agents to this cluster, see Assigning Agents to the Cluster.</p>
<p>Step 6</p>	<p>If appropriate, repeat these steps for any additional agent clusters that you want to add.</p>

Agent Cluster Field Descriptions

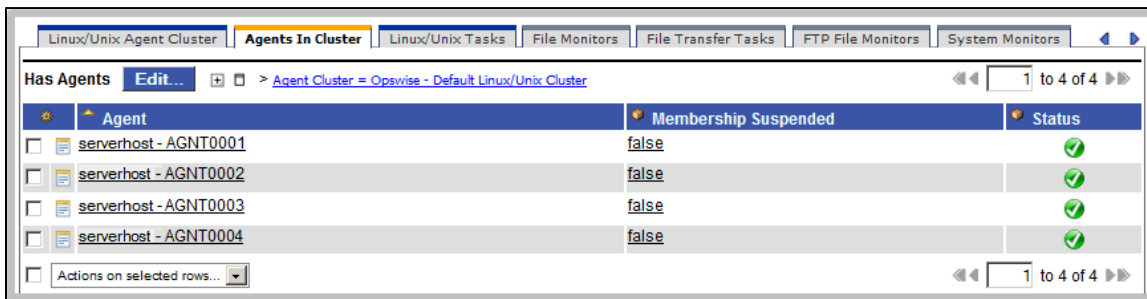
The following table describes the fields, buttons, and tabs on the Agent Cluster screens.

Field Name	Description
Cluster Name	Required. Name used within Opwise to identify this cluster. Up to 40 alphanumeric. It is the user's responsibility to develop a workable naming scheme for clusters.
Version	System-supplied. The version number of the current record, which is incremented by Opwise every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Distribution.	The method used to select an agent. <ul style="list-style-type: none"> • Any - Select any agent in the cluster. • Round Robin - Select the next agent in a round robin series. • Lowest CPU Utilization - Selects the agent whose CPU utilization is currently the lowest.
Last Agent Used	System-supplied. Displays the agent selected the last time a task was sent to this cluster.
Task Execution Limit	Specification for whether a limited or unlimited number of task instances can be run concurrently by the Agents in this agent cluster. (Default is unlimited .) For purposes of imposing this task execution limit, running task instances are those in any of these statuses: Cancel Pending, Queued, Received, Running, Submitted, and Started.
Limit Amount	If Task Execution Limit = Limited; number of tasks that can be running at the same time by the agents in this agent cluster.
Current Task Count	Current number of tasks currently being run by the Agents in this Agent Cluster. (See Resetting the Current Task Count for information on resetting the current task count.)
Suspended	Indication that the ability for this cluster of agents to run tasks has been suspended.
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Suspend Cluster button	Suspend the ability for this cluster of agents to run tasks.
Resume Cluster button	Resume the ability for this suspended cluster of agents to run tasks.
Delete button	Deletes the current record.
Agents in Cluster tab	List of agents assigned to this cluster.
<Task Type> tabs	Each tab - one for each task type that can specify an agent cluster - provides a list of tasks currently being dispatched to this agent cluster.
Versions tab	Stores copies of all previous versions of the current record. See Record Versioning .

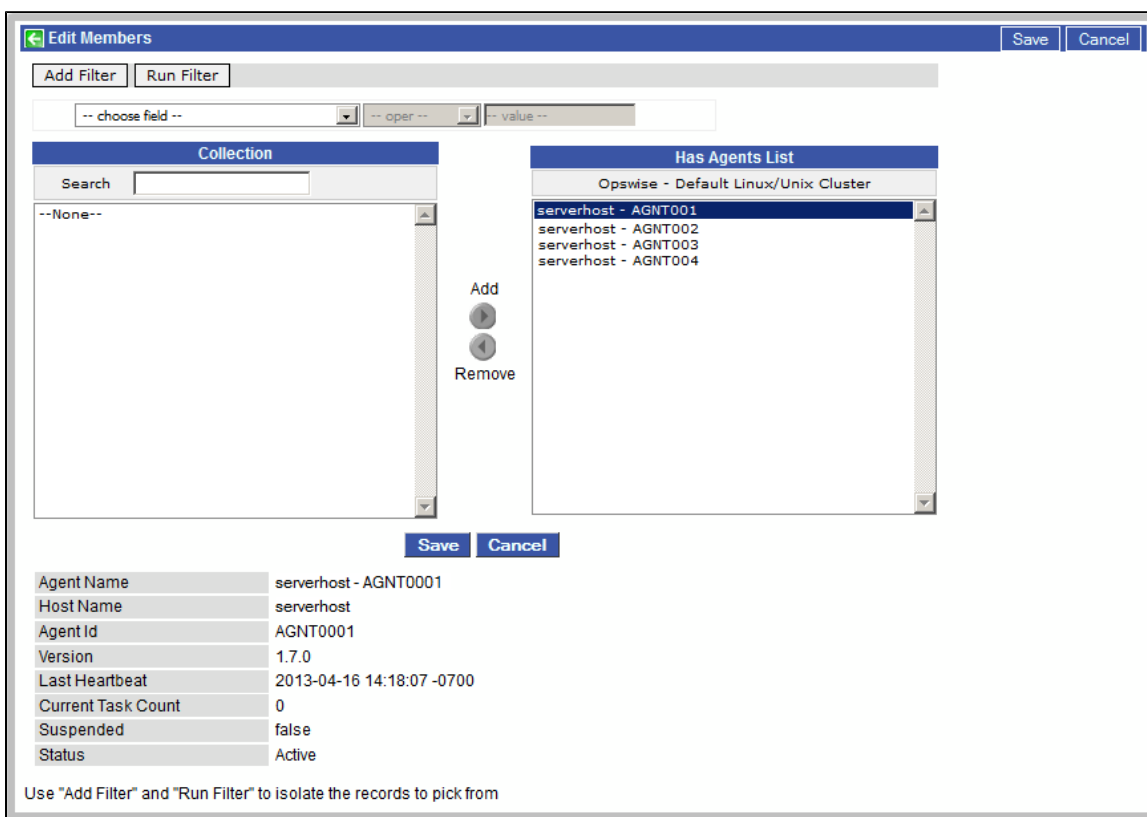
Assigning Agents to the Cluster

Step 1 On the Linux/Unix Clusters List screen (for example), click the Cluster Name of the cluster to which you want to assign one or more existing agents.

Step 2 Click on the **Agents in Cluster** tab to display a list of agents currently assigned to the cluster.



Step 3 Click the **Edit** button. The Edit Members screen displays:



Step 4 The agents listed under Collection are existing agents of the current type that do not already belong to this cluster. The agents listed under Has Agents List are agents that belong to this cluster.

Step 5 To filter the agents listed under Collection:

1. Select filter conditions in the --choose field--, --oper--, and --value-- fields. (See [Create a Filter](#) for information about how to construct a filter.)
2. If you want to add more filter conditions, click **Add Filter**.
3. When you have defined the filter you want, click **Run Filter**. The Collection list now displays only those agents that match the filter.
4. To remove filter conditions, click the X (Delete) icon that displays to the right of each set of filter conditions, and then click **Run Filter**.

Step 6	To add to or remove agents from the Has Agents List: <ul style="list-style-type: none">• To add an agent to the list, double-click on the agent name in the Collection list.• To remove an agent from the list, double-click on the name in the Has Agents List.
Step 7	As you click on an agent, Opswise displays details about the agent at the bottom of the form.
Step 8	When you are finished, click Save .

Suspending Agent Clusters and Agent Cluster Memberships

You can temporarily suspend the ability for a cluster of agents to run tasks, and you can temporarily suspend the agent cluster membership of any agent in an agent cluster.

For information on how to implement these suspensions, see [Suspending Agents, Agent Clusters, and Agent Cluster Memberships](#).

Cluster Nodes

- Introduction
- Displaying Information About Cluster Nodes
 - Cluster Nodes Definition Field Descriptions
- Starting/Stopping Cluster Nodes
- Sending Notifications on Opswise Component Status
 - Email Notifications
 - SNMP Notifications

Introduction

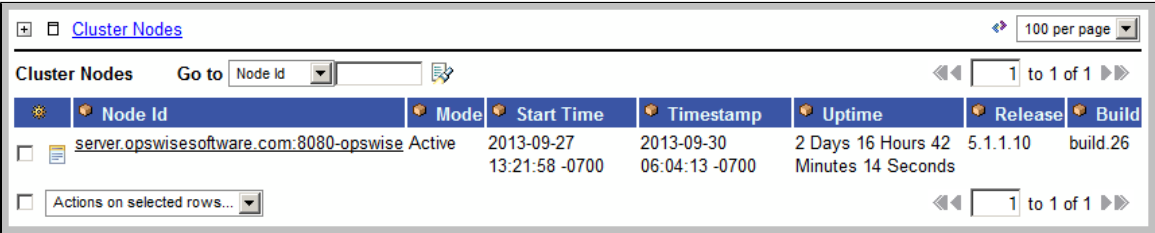
Cluster Nodes are the Controller components of Opswise Automation Center.

(See [Opswise Automation Center System Overview](#) for a diagram and description of the entire Opswise Automation Center system.)

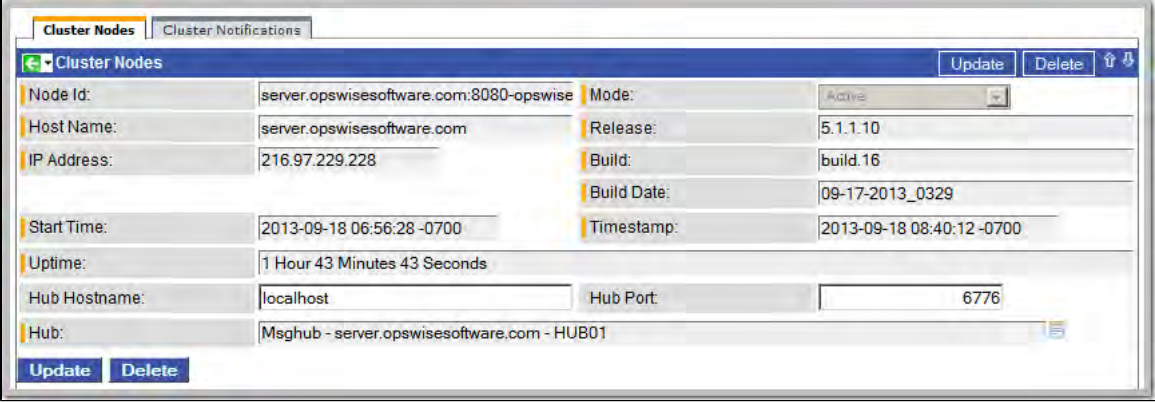
Displaying Information About Cluster Nodes

When you start a Cluster Node for the first time, Opswise automatically creates a database record for that Cluster Node. You can view these records for details and status information.

Step 1 From the navigation pane, select **Automation Center Resources > Cluster Nodes**. The Cluster Nodes List screen displays:



Step 2 To display details about a specific Cluster Node on the list, click the **Node ID** for that Cluster Node. The Cluster Node Definition screen displays:




Only the **Hub Hostname** and **Hub Port** fields can be modified. See the field descriptions, below.

Cluster Nodes Definition Field Descriptions

The following table describes the fields on the Cluster Nodes List screen.

Field Name	Description
Node ID	URL of the cluster node.


Mode	<p>Current mode of the cluster node:</p> <ul style="list-style-type: none"> • Active: Cluster node processes events and messages and interfaces with the database. It is the active node for automated operations.\ • Passive/Available: Cluster node is running and connected to its Message Hub. It performs the following tasks: <ul style="list-style-type: none"> • Accepts HTTP requests for data. It can access the database, generate reports, monitor and display data. • Does not process any events or messages. • Takes over as Active node if it determines that the Active node is not running. • Passive/Unavailable: Cluster node is running but is not connected to its Message Hub. It performs the following tasks: <ul style="list-style-type: none"> • Accepts HTTP requests for data. It can access the database, generate reports, monitor and display data. • Does not process any events or messages. • Takes over as Active node if it determines that the Active node is not running and no other node is in Passive/Available mode. • Offline: Cluster node is not running. <p>(See Passive Cluster Node Limitations for further information on Passive cluster node capabilities.)</p>
Host Name	Specified by the user during installation. The IP address or domain/name of the host machine where the component resides.
Release	System-supplied. Release number for this node. Support purposes only.
IP Address	System-supplied. IP address of this node.
Build	System-supplied. Build ID for this node. Support purposes only.
Build Date	System-supplied. Build date for this node. Support purposes only.
Start Time	System-supplied. Date and time this node was last started.
Timestamp	System-supplied. Date and time of this node's last heartbeat.
Uptime	System-supplied. Amount of time this node has been running.
Hub Hostname	Hostname of the Message Hub connected to this Cluster Node.
Hub Port	<p>Port number of the Message Hub connected to the Cluster Node (default is 6776).</p> <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;">  Note Before changing this port number, check Ports Configuration to see the list of default ports. </div>
Hub	System-supplied. Connector Name of the Message Hub connected to this Cluster Node.
Update button	Saves updates to the record.
Delete button	Deletes the current record.
Cluster Notifications tab	System-supplied. Displays a list of cluster notifications that have been defined for this Cluster Node.

Starting/Stopping Cluster Nodes

For instructions on starting and stopping Cluster Nodes, see [Starting-Stopping Opwise Components](#) and select your platform.

Sending Notifications on Opwise Component Status

You can configure an Opwise component (Cluster Nodes, Agents, and Connectors) to send a notification via email or SNMP when the component goes Offline or becomes Active.

Step 1	From the navigation pane, select Automation Center Resources > [Opswise component] . The component list screen for the selected component displays.
Step 2	Click a <component> Name to display the component definition screen for that component.
Step 3	Click the Notifications tab to display a list of all notifications configured for the component.
Step 4	Click New . The Notifications Wizard screen displays.
Step 5	Select the type of notification you want to configure: Email Notification or SNMP Notification .
Step 6	Complete the fields as needed (see the field descriptions, below). <div style="background-color: #ffffcc; padding: 10px; margin: 10px 0;"> <p> Note Built-in variables are available to pass data about the Cluster Node into the notification (see Controller Variables).</p> </div>
Step 7	Click the Submit button to save the record.
Step 8	If appropriate, repeat these steps for any additional notifications you want to add.

Email Notifications

Shown below is the Email Notification screen that displays for Cluster Nodes.

Email Notification Screen Field Descriptions

The following table describes the fields and buttons on the Email Notification screen.

Field Name	Description
Mode	<ul style="list-style-type: none"> Offline = Trigger the notification when the cluster node goes offline. Active = Trigger the notification when the cluster node comes up. Passive/Available = Trigger the notification when the cluster node becomes passive/available. Passive/Unavailable = Trigger the notification when the cluster node becomes passive/unavailable.

Email Template	Optional. The name of the Email template defined using the Email template screen. The Email template allows you to specify standard recipients and text for outgoing emails. Type in a name, or click the magnifying glass to browse to an existing Email template or create a new one. You must specify either an Email template or Email connection, or both. If you specify both, the Email server specified in the Email Connection record overrides the server in the template.
Email Connection	Required. Name of the Email connection defined using the Email connection screen. The email connection specifies information about the email server. You can also specify the Email connection in the Email template (see above). You must specify an Email template and/or an Email connection. If you specify an Email template and an Email connection, the server selected in the Email connection overrides the server selected in the Email template. Type in a name, click the magnifying glass to browse for an existing Email server definition, or create a new one.
Reply-To	Required. Specifies the email address of the sender. Use commas to separate multiple recipients. Variables supported.
To	Required. Specifies the email address of the recipient. Use commas to separate multiple recipients. Variables supported.
CC	Optional. Specifies the email address of the party being sent a copy of the email, if any. Use commas to separate multiple recipients. Variables supported.
BCC	Optional. Specifies the email address of the party being sent a blind (hidden) copy of the email, if any. Use commas to separate multiple recipients. Variables supported.
Subject	Optional. Specifies the subject line of the email. Variables supported.
Body	Optional. Contains the text of the email message. Variables supported. If both the email template and the email task contain text in the body, the text is appended. Specific built-in variables are available for passing information about the Cluster Node. You must use the appropriate variables for this component type; <code>ops_cluster</code> variables.
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Delete button	Deletes the current record.

SNMP Notifications

Shown below is the SNMP Notification screen that displays for Cluster Nodes.

SNMP Notification Screen Field Descriptions

The following table describes the fields and buttons on the SNMP Notification screen.

Field Name	Description
Mode	<ul style="list-style-type: none">• Offline = Trigger the notification when the cluster node goes offline.• Active = Trigger the notification when the cluster node comes up.• Passive/Available = Trigger the notification when the cluster node becomes passive/available.• Passive/Unavailable = Trigger the notification when the cluster node becomes passive/unavailable.
Node ID	URL of the cluster node.
SNMP Manager	The SNMP Manager that will receive the SNMP notification.
Notification Severity	Optional. Informational only. Indicates the severity of this notification. Options: Normal, Warning, Minor, Major Critical.
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Delete button	Deletes the current record.

Virtual Resources

- Overview
 - Using a Virtual Resource
- Creating a Virtual Resource
 - Field Descriptions
- Assigning Tasks to a Virtual Resource
- Resetting a Renewable Virtual Resource

Overview

A virtual resource allows you to set up a throttling scheme that will manage the number of specific tasks that can run at one time.

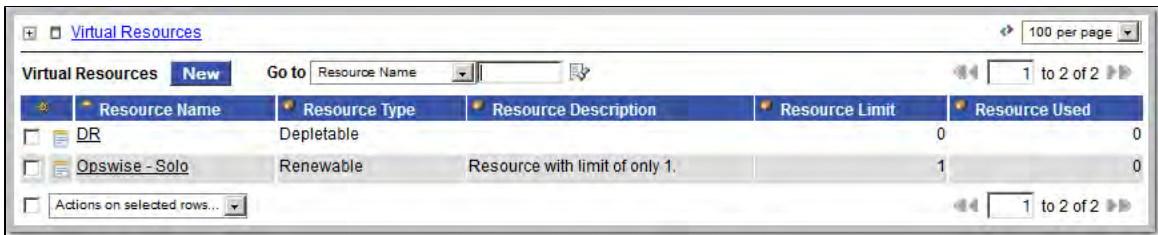
Using a Virtual Resource

Outlined below is the basic procedure and processing flow for using a virtual resource:

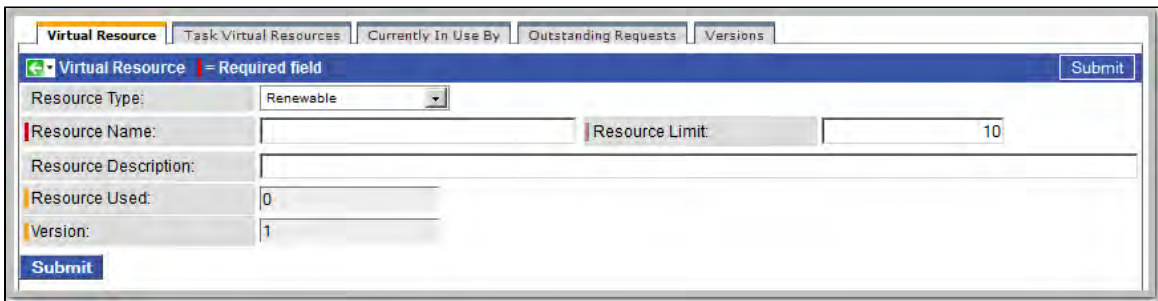
Step 1	<p>Create a virtual resource.</p> <p>There are three types of virtual resources:</p> <ol style="list-style-type: none"> 1. Renewable: Resources that renew; that is, when a task has finished using them, they can be returned and made available to other tasks sharing the same resources. 2. Boundary: Resources that are like "windows." Only those tasks defined to fit through that window (or Resource Limit) will run. For example, if you define a Boundary Resource with Resource Limit of 5, and Task A requires a window (amount) of 5, Task B requires a window (amount) of 5, and Task C requires a window (amount) of 10, both A and B will run. However, C will go into a Resource Wait state. If the Boundary Resource is updated to a Resource Limit of 10, C will run. 3. Depletable: Resources that do not renew. Once consumed by a task, they are gone.
Step 2	Assign a resource limit to the virtual resource as appropriate for the resource type.
Step 3	Assign tasks to the virtual resource.
Step 4	Specify the number of resource units that each task will consume. For example, a task that requires a small amount of processing power might consume one unit; a task that requires a high amount of resources might consume three units. The number of units you specify for each task is relative to the maximum number that you assign to the resource.
Step 5	Save the virtual resource record.
Step 6	<p>When a task with a virtual resource requirement launches, Opswise checks the virtual resource record to see if enough units are available to run the task, based on what other tasks assigned to that virtual resource are currently running.</p> <ul style="list-style-type: none"> • If enough units are available, the task runs and the number of available units is decremented by the amount specified in the task. For example, if the resource has a maximum of ten and the task uses two, the remaining amount available on that virtual resource for use by other tasks is eight. • If there are not enough units available, the task is put into Resource Wait status and is listed in the Outstanding Requests tab in the virtual resource. When the required amount of resource becomes available, the task is launched. • If multiple tasks are in Resource Wait status, the virtual resource priority is used to determine which task will be first to acquire the resource when it becomes available.
Step 7	Tabs on the Virtual Resource record keep track of tasks that are currently "running" on this virtual resource and tasks that are waiting to "run" on this virtual resource.

Creating a Virtual Resource

Step 1 From the navigation pane, select **Automation Center Resources > Virtual Resources**. The Virtual Resources List screen displays:



Step 2 Click **New**. The Virtual Resource Definition screen displays.



Step 3 Using the field descriptions provided below, fill in the fields.

Step 4 Click the **Submit** button to save the record and return to the menu or right-click on the title bar and select **Save** to save the record and remain on the current display.

Note
 This sample Virtual Resource Definition screen shows a Resource Limit of 1. Because each task has a minimum value of 1, this virtual resource would be limited to running only one task at a time.

Field Descriptions

The following table provides descriptions of the fields on the Virtual Resource Definition screen.

Field Name	Description
Resource Type	Type of resource: Renewable, Boundary, or Depletable (see Step 1 in Using a Virtual Resource , above).
Resource Name	Required. Name used within Opswise to identify this resource. Up to 40 alphanumeric. It is the user's responsibility to develop a workable naming scheme for resources.
Resource Limit	Set the number of resources available for the specific resource type.
Resource Description	Description of this virtual resource.
Resource Used	For Resource Type of Renewable only; system-supplied. Shows how many units are currently in use, as of the time you opened the record.
Version	System-supplied. The version number of the current record, which is incremented by Opswise every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.

Delete button	Deletes the current record.
Task Virtual Resources tab	Lists tasks that are assigned to this virtual resource.
Currently In Use By tab	Lists the task instances that have acquired this virtual resource and the number of units acquired, at the time you opened this virtual resource record.
Outstanding Requests tab	Lists the task instances that are currently waiting to acquire this virtual resource and the number of units required for each waiting task instance, at the time you opened this record.
Versions tab	Stores copies of all previous versions of the current record. See Record Versioning .

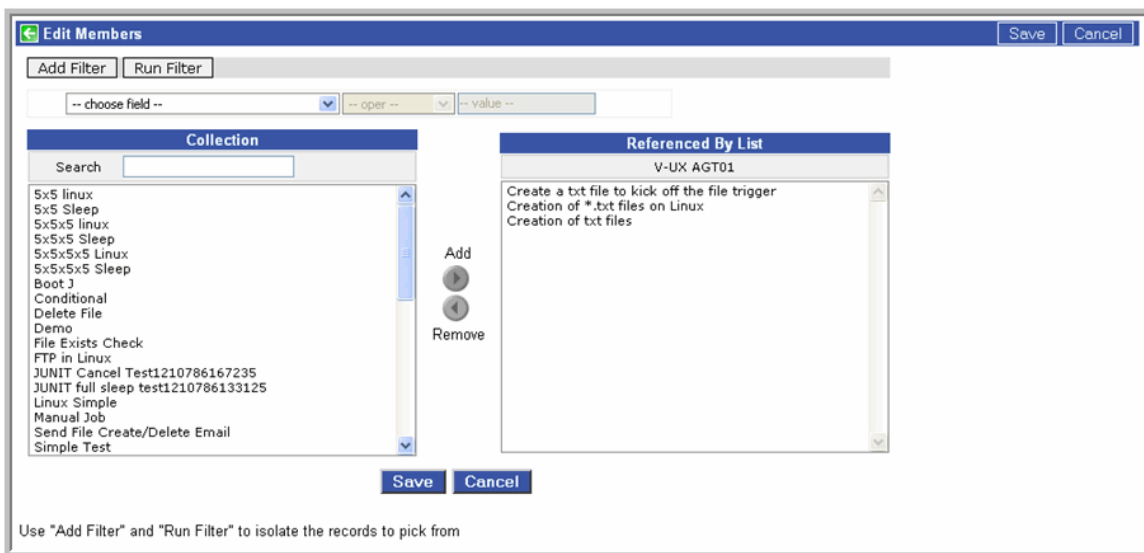
Assigning Tasks to a Virtual Resource



Note
You can also assign a task to a virtual resource from the task screen.

Step 1 On the Virtual Resources Definition screen, click the Task Virtual Resources tab.

Step 2 On the Task Virtual Resources List screen, click the Edit button. The Edit Members screen displays:



Step 3 The tasks listed under Collection are existing tasks that are not assigned to this virtual resource. The tasks listed under Referenced By List are tasks that refer to this virtual resource.

Step 4 To add to or remove tasks from the Referenced By List:

- To add a task to the list, double-click the task in the Collection list.
- To remove a task from the list, double-click the task in the Referenced By List.



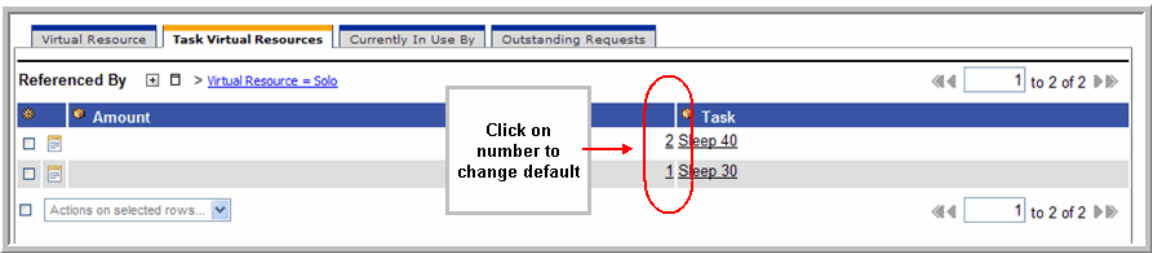
Note
When you click a task, Opwise displays details about the task at the bottom of the screen.

Step 5 To filter the tasks listed under Collection:

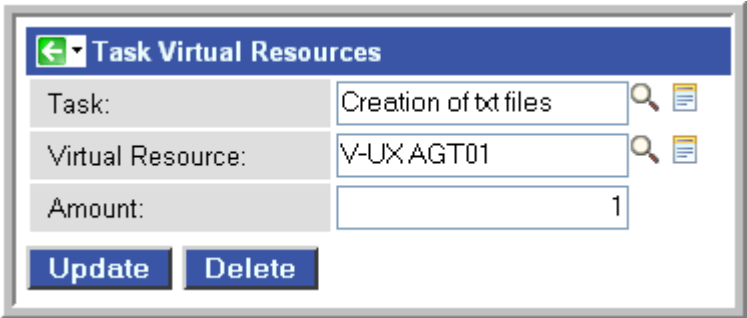
1. Select filter conditions in the --choose field--, --oper--, and --value-- fields. (See [Create a Filter](#) for information about how to construct a filter.)
2. If you want to add more filter conditions, click **Add Filter**.
3. When you have defined the filter you want, click **Run Filter**. The Collection list now displays only those tasks that match the filter.
4. To remove filter conditions, click the X (Delete) icon that displays to the right of each set of filter conditions, and then click **Run Filter**.

Step 6 When you are finished, click **Save**.

Note that the default Amount assigned to each task is 1, as shown.



Step 7 To change the amount, click on the 1. The task virtual resource record opens.



Step 8 Change the Amount field to the new number of units and click **Update**.

Resetting a Renewable Virtual Resource

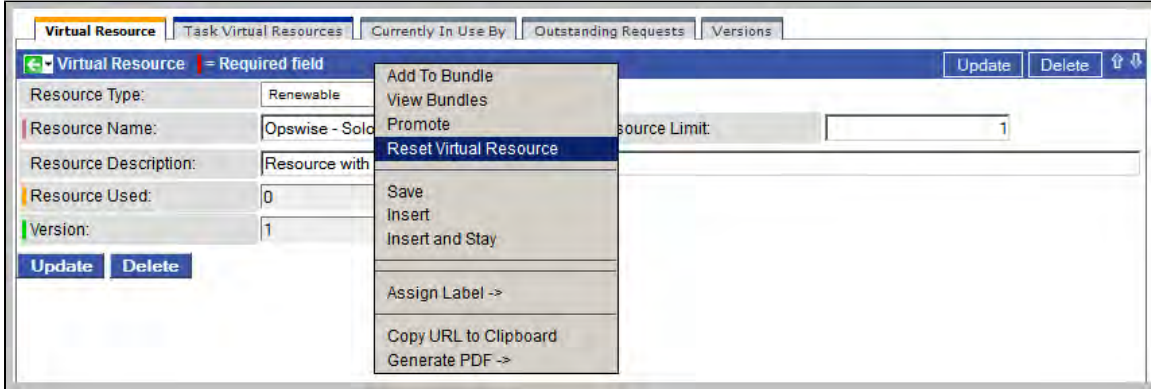
You can reset the [Resource Used](#) amount of a **Renewable** virtual resource to accurately reflect the actual number of resources [currently in use](#).

Resetting a **Renewable** virtual resource requires the `ops_admin` role.

(For **Boundary** and **Depletable** virtual resources, the [Resource Used](#) amount is always reset to 0, as it does not apply to these types of virtual resources.)

Step 1 From the Virtual Resources list screen, select the Virtual Resource that you want to reset.

Step 2 Click on the down arrow next to **Virtual Resource**, or right-click anywhere in the **Virtual Resource** title bar, to display a context-sensitive menu of actions that you can perform for this virtual resource.



Step 3 Click **Reset Virtual Resource** to reset the **Resource Used** amount to accurately reflect the acquired units displayed from the **Currently In Use By** tab.

Script Library

- [Overview](#)
- [Import Procedure](#)
- [Script Library Field Descriptions](#)
- [Executing a Script Using a Task](#)

Overview

The Script Library allows you to store scripts and SAP definition files in the Opswise database for execution on a remote target. When you import a script or SAP definition file into Opswise and execute it via an Opswise task, it is transmitted to the remote machine for execution.

You can use the Script Library with the following task types: Windows, Linux/Unix, and SAP.

You cannot import compiled executables into the Script Library. The content of scripts must be text that can be processed by some shell, script host, or command interpreter.

You can embed [Opswise variables](#) in the script content. Embedded variables are resolved at trigger/run time before the script is sent to the agent.

Opswise variables can be passed as parameters, but the script still has to be written to parse the variables. However, you cannot pass variables as parameters that contain data longer than the parameter field (for example, SQL results).

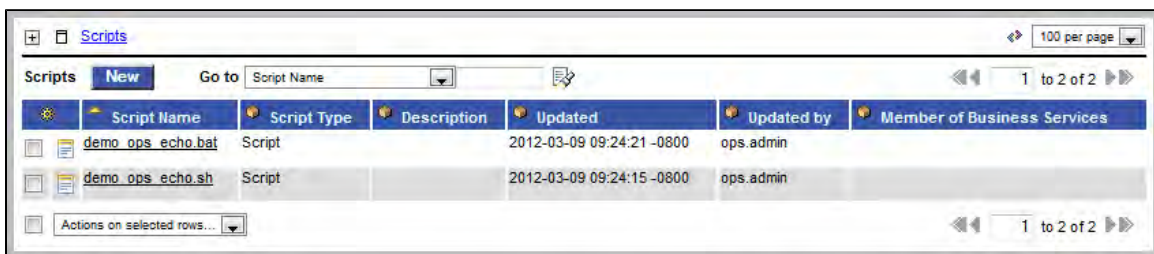
For example, the following script shows how an Opswise variable could be used.

```
#!/bin/bash
echo Task Name: ${ops_task_name}
echo Task Instance: ${ops_task_id}
```

View the [Script Library](#) video to learn more about this feature.

Import Procedure

Step 1 From the navigation pane, select **Automation Center Resources > Script Library**. Opswise displays a list of all existing Scripts, as shown in the following sample list.



The screenshot shows a web interface for the 'Scripts' library. It includes a 'New' button, a 'Go to' search field, and a table with columns for 'Script Name', 'Script Type', 'Description', 'Updated', 'Updated by', and 'Member of Business Services'. Two scripts are listed: 'demo_ops_echo.bat' and 'demo_ops_echo.sh', both updated by 'ops.admin' on 2012-03-09. The interface also features pagination controls and an 'Actions on selected rows...' dropdown.

Script Name	Script Type	Description	Updated	Updated by	Member of Business Services
demo_ops_echo.bat	Script		2012-03-09 09:24:21 -0800	ops.admin	
demo_ops_echo.sh	Script		2012-03-09 09:24:15 -0800	ops.admin	

Step 2 To add a new script, click **New**. Opwise displays a blank Script screen.

Step 3 Using the field descriptions provided below, fill in the fields.

Step 4 To import the script, click **Upload Script File**.

Step 5 At the prompt, browse for and select the script you are importing.


Step 6 In the encoding field, choose the character set of the script. It must be one of the following: ISO-8859-1, US-ASCII, UTF-8, UTF-16, UTF-16BE, UTF-16LE. To display a field tip about any of the options, click the down arrow to display the whole list, then hover over the option.

Step 7 Click **Upload**.

Step 8 Clicking the **Upload** button imports the script you selected and creates a new script record.

Script Library Field Descriptions

Field Name	Description
Script Name	<p>Required. Name of the script. This name can be the same as the name of the script file. You can specify a file extension. The default file extension for Windows is <code>.bat</code>.</p> <p>If the name has the extension <code>.ps1</code>, Windows will run the script as a powershell script. You may have to create the appropriate file association and security for this to work.</p>

Version	Task definition only; system-supplied. The version number of the current record, which is incremented by Opwise every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Description	User-supplied description of this record.
Script Type	Type of script. Options: Script (for use in Windows or Linux/Unix tasks) or SAP Definition (for use in an SAP task).
Resolve Opwise Variables	Controls whether or not the script will be parsed in pursuit of Opwise variables. It allows the Controller to avoid the overhead of parsing a script that does not contain variables. <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;">  Note Variables <i>could</i> be embedded with this field disabled; likewise, you could have a script with no variables but have this field enabled. However, enabling this field for a script that does not contain Opwise variables will impose an unnecessary burden (however small) on the Controller. </div>
Content	Content of the script or batch file.
Updated	Date and time this record was last updated.
Updated by	User who last updated this record.
Member of Business Services	User-defined. Allows you to select one or more Business Services that this record definition belongs to. Click the lock icon to unlock the field and select Business Services .
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Delete button	Deletes the current record.
Upload Script File button	Uploads a script from the local file system.
Used By <task type> Tasks tabs	Each tab lists the tasks of that type that are using this script.
Notes tab	Displays all notes associated with this task.
Versions tab	Stores copies of all previous versions of the current record. See Record Versioning .

Executing a Script Using a Task

Step 1 Create one of the following task types: [Windows](#), [Linux/Unix](#), or [SAP](#).

Step 2	Complete the fields according to the field descriptions.
Step 3	For Windows and Linux/Unix tasks, in the Command or Script field, select Script . For SAP, in the Library or File System field, select Script Library .
Step 4	In the Script field, browse the Script Library and select the script you want to execute.

Email Template

- Overview
- Creating a New Email Template
- Email Template Field Descriptions

Overview

The Email template allows you to construct commonly-used information that can be copied to create [Email tasks](#).

If an Email task specifies a template, Opwise uses the information in the template to construct and execute the Email task. Any information specified in the task overrides what is specified in the template.

Creating a New Email Template

Step 1 From the navigation pane, select **Automation Center Resources > Email Templates**. The Email Templates List screen displays.

Step 2 Click **New**. The Email Template Definition screen displays.

The screenshot shows the 'Email Template Definition' screen. At the top, there are tabs for 'Email Template' and 'Versions'. Below the tabs is a blue header bar with a back arrow, the text 'Email Template', a red exclamation mark icon, and the text '= Required field'. A 'Submit' button is located in the top right corner of the header. The main form area contains several fields: 'Template Name' (text input), 'Email Connection' (text input with a search icon), 'Reply-To' (text input), 'To:' (text area with a plus icon), 'Cc:' (text area with a plus icon), 'Bcc:' (text area with a plus icon), 'Subject:' (text input), and 'Body:' (text area with a plus icon). At the bottom, there is a 'Version:' field with the value '1' and another 'Submit' button.

Step 3 Using the field descriptions provided below as a guide, complete the fields as needed.

Step 4 Click the **Submit** button to save the record and return to the menu, or, right-click on the title bar and select **Save** to save the record and remain on the current display.

Step 5 If appropriate, repeat these steps for any additional templates you want to add.

Email Template Field Descriptions

The following table describes the fields, buttons, and tabs on the Email Template Definition screen.

Field Name	Description
Template Name	Required. Name used within Opwise to identify this resource. Up to 40 alphanumeric characters. It is the user's responsibility to develop a workable naming scheme for resources.
Email Connection	Required. Name of the Email connection defined using the Email connection screen. The email connection specifies information about the email server. You can also specify the Email connection in the Email template (see above). You must specify an Email template and/or an Email connection. If you specify an Email template and an Email connection, the server selected in the Email connection overrides the server selected in the Email template. Type in a name, click the magnifying glass to browse for an existing Email server definition, or create a new one.
Reply-To	Required. Specifies the email address of the sender. Use commas to separate multiple recipients. Variables supported.
To	Required. Specifies the email address of the recipient. Use commas to separate multiple recipients. Variables supported.
CC	Optional. Specifies the email address of the party being sent a copy of the email, if any. Use commas to separate multiple recipients. Variables supported.
BCC	Optional. Specifies the email address of the party being sent a blind (hidden) copy of the email, if any. Use commas to separate multiple recipients. Variables supported.
Subject	Optional. Specifies the subject line of the email. Variables supported.
Body	Optional. Contains the text of the email message. Variables supported. If both the email template and the email task contain text in the body, the text is appended.
Version	System-supplied. The version number of the current record, which is incremented by Opwise every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Delete button	Deletes the current record.
Versions tab	Stores copies of all previous versions of the current record. See Record Versioning .

Email Connection

- [Overview](#)
- [Creating a New Email Connection](#)
- [Email Connection Field Descriptions](#)

Overview

An Email connection provides all of the email server information necessary for Opwise to send an email.

Email connections are used these ways within Opwise:

- An [Email Task](#) uses the Email connection to generate emails independent of tasks.
- An [Email Notification](#) uses the Email connection to generate notifications related to tasks.
- [Agents and Connectors](#) and [Cluster Nodes](#) use the Email connection to generate email notifications.
- [System Operations](#) use Email connections to generate system notifications.



Note

The Email connections described here are not used for emailing reports. See [Setting Up the Email Server](#).


Creating a New Email Connection

Step 1	From the navigation pane, select Automation Center Resources > Email Connections . The Email Connection List screen displays.
Step 2	Click New . The Email Connection Definition screen displays:
Step 3	Using the field descriptions provided below as a guide, complete the fields as needed.
Step 4	Right-click and select Save to save the record and remain on the current display.
Step 5	To test the connection, click Test Connection .
Step 6	Click the Submit button to save the record.
Step 7	If appropriate, repeat these steps for any additional Email connections you want to add.

Email Connection Field Descriptions

The following table describes the fields, buttons, and tabs on the Email Connection Definition screen.

Field Name	Description
------------	-------------

Connection Name	Required. Name (maximum 40 alphanumeric characters) used within Opwise to identify this resource. It is the user's responsibility to develop a workable naming scheme for resources.
Outgoing Mail Server (SMTP)	Required. The name or IP address of the outgoing email server.
SMTP Port	Port number on the machine where the email server resides. Any port number between 1 and 65535.
SMTP Requires SSL	Specifies whether your SMTP server requires SSL.
Email Address	Required. The email address of the sender.
Authentication Req'd	If enabled, the user name and password (below) are required.
User Name	Name Opwise will use to connect to the server.
Password	Password Opwise will use to connect to the server.
Use for System Notifications	Indicates whether or not this Email Connection is to be used for system notifications. <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note Only one Email Connection can be used for system notifications. If this field is checked on an Email Connection screen, it will appear unchecked on all other Email Connection screens. If you then check this field on another Email Connection screen, it automatically will be unchecked from the screen on which it had been checked.</p> </div>
Version	System-supplied. The version number of the current record, which is incremented by Opwise every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Test Connection button	After saving the record to the database, click Test Connection to run a connection test.
Delete button	Deletes the current record.
Email Tasks tab	Provides a list of tasks that use this email server.
Versions tab	Stores copies of all previous versions of the current record. See Record Versioning .

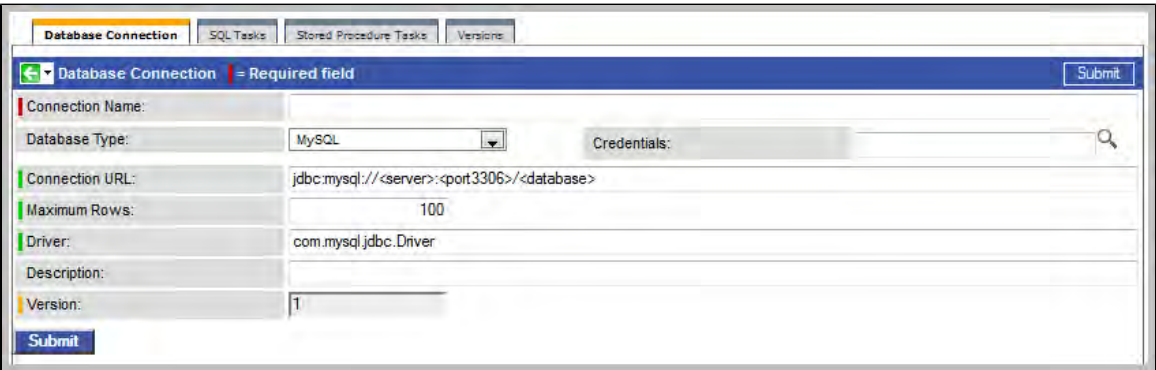
Database Connection

- [Overview](#)
- [Creating a New Database Connection](#)
- [Database Connection Field Descriptions](#)

Overview

The Database Connection provides all the database server information necessary for Opwise to execute an [SQL task](#) or a [Stored Procedure Task](#).

Creating a New Database Connection

Step 1	From the navigation pane, select Automation Center Resources > Database Connections . The Database Connections List screen displays.
Step 2	Click New . The Database Connection Definition screen displays.
	
Step 3	Using the field descriptions provided below as a guide, complete the fields as needed.
Step 4	Right-click and select Save to save the record and remain on the current display.
Step 5	To test the connection, click Test Connection .
Step 6	Click the Submit button to save the record.
Step 7	If appropriate, repeat these steps for any additional Database Connections you want to add.

Database Connection Field Descriptions

The table below describes the fields, buttons, and tabs on the Database connection screen.

Field Name	Description
Connection Name	Required. Name (maximum 40 alphanumeric characters) used within Opwise to identify this resource. It is the user's responsibility to develop a workable naming scheme for resources.

Database Type	<p>Required. The type of database. Options:</p> <ul style="list-style-type: none"> • MySQL • MS SQL Server • Oracle • DB2 • Sybase SQL Anywhere • Other
Credentials	<p>Credentials under which an Agent runs this task. These credentials override any credentials provided in the Agent resource definition for any Agent running this task.</p>
Connection URL	<p>The URL of the database.</p> <p>Note: If you are using a MySQL database and want the ability to issue multiple SQL commands from a single task, you need to enable this by appending the following string to the end of the connection string:</p> <pre>?allowMultiQueries=true</pre> <p>For example:</p> <pre>jdbc:mysql://localhost:3306/opwise?allowMultiQueries=true</pre>
Maximum Rows	Optional. If necessary, specify a limit to the number of rows you want returned by the SQL statement.
Driver	Name of the JDBC driver.
Description	Optional. Description of the database.
Version	System-supplied. The version number of the current record, which is incremented by Opwise every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Test Connection button	After saving the record to the database, click Test Connection to run a connection test.
Delete button	Deletes the current record.
SQL and Stored Procedure Tasks tabs	A separate tab is provided for each task type that uses a database (SQL and Stored Procedures). Each tab displays a list of tasks that are using this database connection.
Versions tab	Stores copies of all previous versions of the current record. See Record Versioning .

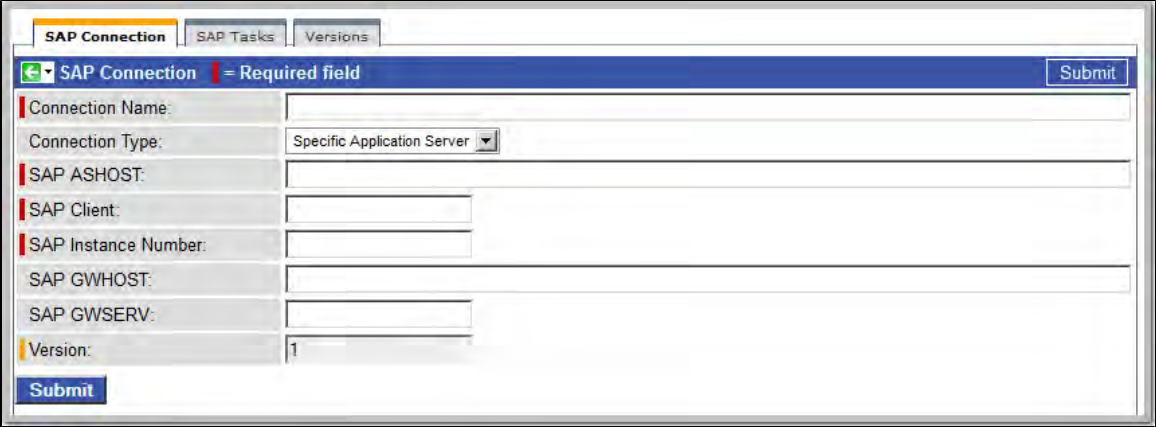
SAP Connection

- [Overview](#)
- [Creating a New SAP Connection](#)
- [SAP Connection Field Descriptions](#)

Overview

The SAP Connection provides all the SAP server information necessary for Opwise to execute an [SAP Task](#) on an SAP system. These instructions assume the user is familiar with SAP.

Creating a New SAP Connection

Step 1	From the navigation pane, select Automation Center Resources > SAP Connections . The SAP Connections List screen displays.
Step 2	Click New . The SAP Connection Definition screen displays.
	
Step 3	Using the field descriptions provided below as a guide, complete the fields as needed.
Step 4	Right-click and select Save to save the record and remain on the current display.
Step 5	Click the Submit button to save the record.
Step 6	If appropriate, repeat these steps for any additional SAP connections you want to add.

SAP Connection Field Descriptions

The table below describes the fields, buttons, and tabs on the SAP Connection screen.

Field Name	Description
Connection Name	Required. Name (maximum 40 alphanumeric characters) used within Opwise to identify this resource. It is the user's responsibility to develop a workable naming scheme for resources.

Connection Type	Type of SAP connection. Options: <ul style="list-style-type: none"> • Specific Application Server Connection to a specific SAP application server (type A RFC connection). • Load Balancing Connection to an SAP system where the application server is determined by load balancing (type B RFC connection).
SAP ASHOST	Required. Host name of an SAP application server. If the path to the server goes through SAP routers, prefix the host name with the SAP router string.
SAP Client	Required. SAP Client number.
SAP Instance Number	Required. SAP instance number.
SAP GWHOST	Host name of the SAP gateway.
SAP GWSERV	Service name of the SAP gateway.
System ID	System ID of the SAP system to which you want to connect.
Message Server	Host name of the message server.
Group	Application servers group name.
Version	System-supplied. The version number of the current record, which is incremented by Opwise every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Delete button	Deletes the current record.
SAP Tasks tab	Lists the SAP tasks that use this SAP connection.
Versions tab	Stores copies of all previous versions of the current record. See Record Versioning .

SNMP Manager

- [Overview](#)
- [Creating a New SNMP Manager](#)
- [SNMP Manager Field Descriptions](#)

Overview

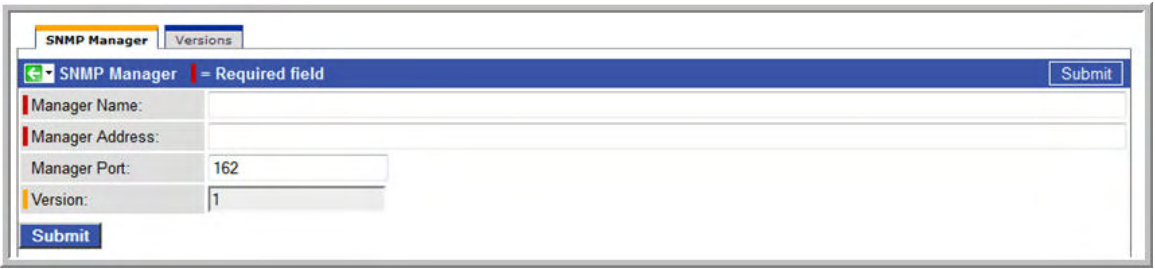
An SNMP Manager is the network manager to which Opwise sends [SNMP notifications](#).

SNMP Managers can receive SNMP notifications when:

- An Agent or Connector (Transporter or Message Hub) goes down or comes back up. See [Sending Notifications on Opwise Component Status](#).
- An SNMP Notification is associated with a task. See [Creating SNMP Notifications](#).

Creating a New SNMP Manager

The SNMP Manager provides all the information necessary for Opwise to send an SNMP message using [SNMP Notifications](#).

Step 1	From the navigation pane, select Automation Center Resources > SNMP Managers . The SNMP Managers List screen displays.
Step 2	Click New . The SNMP Manager Definition screen displays: 
Step 3	Using the field descriptions provided below as a guide, complete the fields as needed.
Step 4	Click Submit to save the record.
Step 5	If appropriate, repeat these steps for any additional SNMP managers you want to add.

SNMP Manager Field Descriptions

The following table describes the fields and buttons on the SNMP Manager screen.

Field Name	Description
Manager Name	Required. Name used within Opwise to identify this resource. Up to 40 alphanumeric. It is the user's responsibility to develop a workable naming scheme for resources.
Manager Address	Required. Name or IP address of the SNMP manager.
Manager Port	Port number used by the SNMP manager. Any port number between 1 and 65535.
Version	System-supplied. The version number of the current record, which is incremented by Opwise every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .

Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Delete button	Deletes the current record.
Versions tab	Stores copies of all previous versions of the current record. See Record Versioning .

Tasks and Workflows



Tasks



Workflows

Creating Tasks

Types of Tasks

Manually Running and Controlling Tasks

Retrieving Output from a Completed Task

Copying Tasks



Monitoring Tasks and Workflows

Monitoring Activity from the Activity Screen

Monitoring Activity from the Task Instances Screen

Monitoring Workflows

Viewing Task Instances for a Specific Task

Displaying Task Instance Status

Monitoring Activity History

Creating and Maintainin



Task Actions

Creating Email Notificat

Creating SNMP Notifica

Setting Abort Actions

Setting Variable Actions

Setting System Operatic

Setting Mutually Exclus.

Creating Step Condition

Creating Notes

Variables and Functions



The information on these pages also is located in the [Opswise Automation Center 5.1.1 User Guide.pdf](#).

Creating Tasks

- [Tasks](#)
 - [Task Types](#)
 - [Built-In Variables](#)
- [Tasks List](#)
- [Creating a Task](#)

Tasks

An Opwise task executes a process on a machine, either local or remote. The process might be resident on the machine (agent-based process), or the task itself (such as a File Monitor task) might embed the process.

You can launch tasks within [workflows](#), by way of [triggers](#), or manually.

Task Types

Task Type	Usage
Workflow	Create a sequence of connected tasks, which could include other workflows.
Linux/Unix	Run a platform-specific application on a Linux/Unix machine.
Windows	Run a platform-specific application on a Windows machine.
z/OS	Run a platform-specific application on a z/OS machine.
Indesca	Run a platform-specific application on a machine where Indesca is running.
SAP	Send commands to an SAP system and gather status information and output back from SAP.
File Transfer	Execute file transfers on remote machines using FTP, SFTP, and INFITRAN protocols.
Manual	Create a pause in the workflow during which the user must take some action.
Sleep	Execute a sleep command for a specified period of time or until a specific time.
SQL	Execute one or a series of SQL statements against the database specified in the task.
Stored Procedure	Execute a stored procedure against the database specified in the task.
Email	Create and send emails.
Task Monitor	Monitor another task or tasks for one or more specific statuses.
File Monitor	Monitor a specific remote machine for the creation, deletion, change, existence, or non-existence of one or more files at a specific location.
FTP File Monitor	Monitor for a file on a remote machine where an FTP server is running.
System Monitor	Monitor a specific remote machine and check for free disk space.
Application Control	Execute a start, stop, or query command against an application in the Opwise network.

Built-In Variables

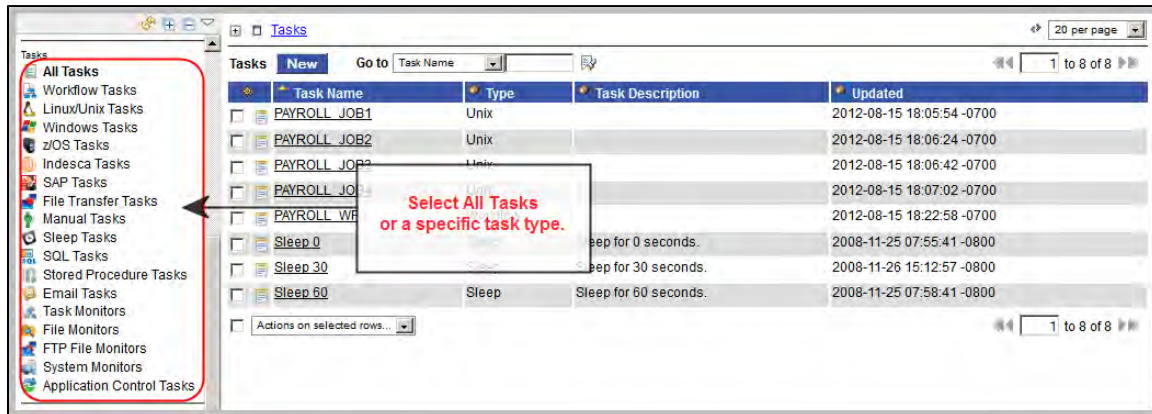
Several [built-in variables](#) are available for use in all task types; other built-in variables exist for specific task types.

Tasks List

The Tasks List screen displays a list of all currently defined tasks for all task types.

To access the Tasks List screen, select **Automation Center > Tasks > All Tasks** from the navigation pane. (To see a list of tasks for a single

task type, select that task type from the navigation pane.)



The following table provides column descriptions for the default display of the Task List screen.

For information about customizing this list, including filtering, sorting, searching, and other list features, see [Using Lists](#).

Column	Description
Task Name	User-defined. Name assigned to this task.
Type	Type of task. Options: <ul style="list-style-type: none"> • Workflow Tasks • Linux/Unix Tasks • Windows Tasks • z/OS Tasks • Indesca Tasks • SAP Tasks • File Transfer Tasks • Manual Tasks • Sleep Tasks • SQL Tasks • Stored Procedure Tasks • Email Tasks • Task Monitors • File Monitors • FTP File Monitors • System Monitors • Application Control Tasks
Task Description	User-defined. Copied from the Task Description field in the task.
Updated	System-supplied. The date and time this record was last updated.

Creating a Task

You can create a task either of two ways.

Step 1	From the navigation pane, select Automation Center > Tasks > <task type> .
Step 2	When the Tasks List screen for that task type displays, click the New button. The Task Definition screen for that task type displays.

OR

Step 1	From the navigation pane, select Automation Center > Tasks > All Tasks .
---------------	---

Step 2 When the All Tasks List screen displays, click **New**. The Task Wizard screen displays.



Step 3 Click the task type for task you want to create. The Task Definition screen for that task type displays.

See each [task type](#) for detailed information on creating that task.

Linux Unix Task

- [Before You Begin](#)
- [Built-In Variables](#)
- [Creating a New Linux/Unix Task](#)
- [Linux/Unix Task Field Descriptions](#)
- [Specifying When a Task Runs](#)
- [Monitoring Task Execution](#)

Before You Begin

The Linux/Unix task allows you to run a platform-specific application on a Linux/Unix machine. To run a Linux/Unix task, you must first complete the following tasks:

- Install an Opwise Linux/Unix agent on a Linux/Unix machine.
- Launch the agent. When the agent connects with the Controller, it automatically creates an [agent resource definition](#) in the database.
- Optionally, customize the agent heartbeat and log levels, as described in [Linux/Unix Agent Definition Field Descriptions](#).

Built-In Variables

The built-in variables outlined below can be used in a Linux/Unix task to pass data where appropriate:

- [Task and Task Instance Variables](#)
- [Script Variables](#)
- [Agent Variables](#)

Creating a New Linux/Unix Task

Step 1 From the navigation pane, select **Automation Center > Tasks > Linux/Unix Tasks**. The Linux/Unix Tasks List screen displays.

Step 2 Click **New**. The Linux/Unix Task Definition screen displays.

Step 3 Using the field descriptions provided below as a guide, complete the fields as needed.

Step 4 Click the **Submit** button to save the record and return to the Linux/Unix Tasks List screen, or right-click on the title bar and select **Save** to save the record and remain on the Linux/Unix Task Definition screen.

Step 5 If appropriate, repeat these steps for any additional tasks you want to add.

The task run statistics shown at the bottom of the screen appear after the first time this task has been launched.

Linux/Unix Task Field Descriptions

The table below describes the fields, buttons, and tabs on the task definition and task instance screens. Color coding is provided that differentiates the following three types of fields:

- Fields that display on the task definition and task instance screens are shown in black.
- Fields that display only on the task definition screen are shown in green.
- Fields that display only on the task instance screen are shown in maroon.

Field Name	Description
Task/Instance Name	Required. Name used within Opwise to identify this task. Up to 40 alphanumeric; variables supported. It is the user's responsibility to develop a workable naming scheme for tasks.
Invoked by	Task instance only; system-supplied. How the task instance was launched. One of the following: <ul style="list-style-type: none"> • Trigger: (Trigger Name) - The instance was launched by the named trigger. • Workflow: (Workflow Name) - The instance was launched by the named workflow. • Manually Launched - The instance was launched by a user. To determine the name of the user: <ol style="list-style-type: none"> 1. From the Activity or Task Instances screen, click the task instance name to open the record. 2. The Execution User field identifies the user who launched the task instance.
Task	Task instance only; system-supplied. Hover over the paper icon to display more information about the task instance. Click the paper icon to display the task definition record.
Execution User	Task instance only; system-supplied. If the task was launched manually, the ID of the user who launched it.
Instance Reference Id	Task instance only; system-supplied. Opwise increments this number each time the task is run.
Credentials	Credentials under which an Agent runs this task. These credentials override any credentials provided in the Agent resource definition for any Agent running this task.
Version	Task definition only; system-supplied. The version number of the current record, which is incremented by Opwise every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning.
Credentials Variable	Optional. If enabled, the Credentials field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: <code>\${variable name}</code> . The variable must be a supported type as described in Variables and Functions .
Run as sudo	Optional. Run the command as Sudo (superuser do).
Agent	Optional. The name of the agent resource definition that identifies the machine where the operation will run. If you do not specify an agent, you must specify an agent cluster (see below).
Agent Cluster	Optional. You can specify an agent cluster in addition to or in place of a specific agent. An agent cluster is a group of agents, one of which Opwise will choose to run this task. If you specify an agent and an agent cluster, Opwise Automation Center first tries to run the task on the specific agent. If the agent is not available, Opwise reverts to the agent cluster. See Agent Clusters for more information.
Agent Variable	Optional. If enabled, the Agent field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: <code>\${variable name}</code> . The variable must be a supported type as described in Variables and Functions .
Agent Cluster Variable	Optional. If enabled, the Agent Cluster field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: <code>\${variable name}</code> . The variable must be a supported type as described in Variables and Functions .

Hold on Start	If enabled, when the task is launched it appears in the Activity display with a status of Held. The task runs when the user releases it.
Hold Reason	Information about why the task will be put on hold when it starts.
Cluster Broadcast	Task definition only; optional. You can specify a Cluster Broadcast in place of a specific agent and/or Agent Cluster. When you specify an agent cluster in the Cluster Broadcast field, Opwise Automation Center runs the task on all the agents in the cluster. Each instance of the task running on its own agent becomes a separate task instance record in the database and displays separately in the Activity monitor. See Agent Clusters for more information about defining agent clusters.
Task Description	User-supplied description of this record.
Status	Task instance only; system-supplied. See Task Instance Statuses .
Exit Code	Task instance only; system-supplied. The exit code captured by the agent when executing the task (for example, a command or script).
Status Description	Task instance only; system-supplied. Provides additional information, if any, about the status of the task.
Queued Time	Task instance only; system-supplied. The time the task was queued for processing.
Process ID	Task instance only; system-supplied. The ID of the process that was launched.
Start Time	Task instance only; system-supplied. The date and time the task started.
CPU Time	Task instance only; system-supplied. The amount of CPU time the task took to run.
End Time	Task instance only; system-supplied. The date and time the task instance completed.
Duration	Task instance only; system-supplied. The amount of time the task took to run.
Member of Groups	User-defined. Allows you to select one or more Business Services that this record definition belongs to. Click the lock icon to unlock the field and select Business Services .
User Estimated Duration	Task definition only; optional. The estimated amount of time it should normally take to run this task. Opwise uses this information to calculate the User Estimated End Time on a task instance record.
Command or Script	Specifies whether a single command or a script is being executed. Options: <ul style="list-style-type: none"> • Command (default) • Script

Script	Required (if Script is selected in Command or Script field). Name of the script that has been uploaded into the Script Library and will be executed by this task.
Command	Required (if Command is selected in Command or Script field). Command being executed on the remote machine. Variables supported.
Parameters	Optional. Any arguments needed by the program to execute properly. Variables supported.
Runtime Directory	Optional. The directory from which the application should be executed. Variables supported.
Exit Code Processing	Required. Specifies how Opwise Automation Center should determine whether the executed command failed or completed successfully. Options: <ul style="list-style-type: none"> • Success Exitcode Range - The command is considered completed successfully if its exit code falls within the range specified in the Exit Codes field (see below). • Failure Exitcode Range - The command is considered failed if its exit code falls within the range specified in the Exit Codes field (see below). • Success Output Contains - The command is considered completed successfully if its output contains the text specified in the Scan Output For field (see below). • Failure Output Contains - The command is considered failed if its output contains the text specified in the Scan Output For field (see below). • Step Conditions - The command is considered completed successfully/failed if any of its specified condition codes falls within the range specified under the Step Conditions tab (see Creating Step Conditions).
Output Type	Required if Exit Code Processing = Success Output Contains or Failure Output Contains. This field specifies the type of output. Options: <ul style="list-style-type: none"> • Standard Output (STDOUT) • Standard Error (STDERR) • File
Exit Codes	Required if Exit Code Processing = Success Exitcode Range or Failure Exit Code Range. This field specifies the range. Format: Numeric. Use commas to list a series of exitcodes; use hyphens to specify a range. Example: 1,5, 22-30.
Scan Output For	Required if Exit Code Processing = Success Output Contains or Failure Output Contains. This field specifies the text for which Opwise should scan the output file. Opwise will process this field as a regular expression.
Output File (Exit Code Processing)	Required if Output Type = File. This field specifies the path and file name of the output file that should be scanned for the text in the Scan Output For field.
(Environment Variables) Name and Value	Optional. Allows you to enter environment variables needed by the program to run. For each variable, enter a Name and Value, and then click Add . You can add a maximum of 4,000 characters for the combined Names and Values of all variables. The variable is listed in the space underneath. To delete a variable, click the X button.
Environment Variables List	Displays - on the Linux/Unix Tasks List screen - any environment variables added to this task.

Automatic Output Retrieval	<p>Optional. Allows you to specify whether you want Opwise to automatically retrieve any output from the job and attach it to the task instance record. Options:</p> <ul style="list-style-type: none"> • None - Do not attach any output to the task instance record. • Standard Output - Attach all standard output. • Standard Error - Attach standard error output. • File - Attach the file specified in the Output File field.
Output File (Automatic Output Retrieval)	<p>Required if Automatic Output Retrieval=File. This field specifies the path and filename containing the output that you want automatically retrieved and attached to the task instance.</p>
Start Line	<p>Optional. Allows you to instruct Opwise to retrieve data beginning at the line indicated. If a Start Line value is not specified on the screen, the default is 1.</p>
Number of Lines	<p>Optional. Allows you to limit the retrieved data to the number of lines specified. If a Number of Lines value is not specified, the default is the value of the Retrieve Output Default Maximum Lines Opwise system property.</p>
Scan Text	<p>Optional. Instructs Opwise to scan the data for the text specified and retrieve only that. Opwise will process this field as a regular expression.</p>
Late Start	<p>If enabled, and if the task instance starts after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late start (see Late Start Type). To determine whether a task instance started late, open the task instance and locate the Started Late field; the field is checked if the instance started after the specified time. This field only appears on the task instance if the user specified a Late Start in the task definition.</p>
Late Start Type	<p>Required if Late Start is enabled. Options are:</p> <ul style="list-style-type: none"> • Time - Flag the task if it starts after the specified time. • Duration - Flag the task if it starts a certain amount of time after the programmed start time. The task must have a specific start time.
Late Start Time	<p>Time after which the task start time is considered late. Use hh:mm, 24-hour time</p>
Late Start Duration	<p>Duration (amount of relative time) after which the task is considered to have started late. For a task within a workflow, the duration is the period between the time the workflow starts and the time the task itself starts. For example, a task might have a Late Start Duration of 60 minutes. If the workflow starts at 9:00 a.m. but the task itself does not start until 10:30, the task has started late.</p> <p>For a task that is not within a workflow, Late Start Duration has meaning only if the task has been held upon starting. For example, if a task has a Late Start Duration of 60 minutes and the Hold on Start field is enabled, if the task is not released from hold within the amount of time specified in the Late Start Duration field, the task has started late.</p>
Started Late	<p>Task instance only; system-supplied. This field is flagged if the task started later than the time specified in the Late Start fields.</p>
Late Finish	<p>If enabled, and if the task instance finishes after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late finish (see Late Finish Type). To determine whether a task instance finished late, open the task instance and locate the Finished Late field; the field is checked if the instance finished after the specified time or lasted longer than expected. This field only appears on the task instance if the user specified a Late Finish in the task definition.</p>

Late Finish Type	<p>Required if Late Finish is enabled. Options are:</p> <ul style="list-style-type: none"> • Time - Flag the task if it finishes after the specified time (see Late Finish Time). • Duration - Flag the task if it finishes a certain amount of time after the programmed finish time (see Late Finish Duration). The task must have a specific finish time.
Late Finish Time	If Late Finish Type is Time, use this to specify the time after which the task finish time is considered late. Use hh:mm, 24-hour time.
Late Finish Duration	If Late Finish Type is Duration, use this to specify the longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Finished Late	Task instance only; system-supplied. This field is flagged if the task finished later than the time or duration specified in the Late Finish fields.
Early Finish	If enabled, and if the task instance finishes before the time or period specified, the task instance is flagged as early. You can specify a time or duration to determine an early finish (see Early Finish Type). To determine whether a task instance finished early, open the task instance and locate the Finished Early field; the field is checked if the instance finished before the specified time or did not last as long as expected. This field only appears on the task instance if the user added Early Finish specifications to the task definition.
Early Finish Type	<p>Required if Early Finish is enabled. Options are:</p> <ul style="list-style-type: none"> • Time - Flag the task if it finishes before the specified time (see Early Finish Time). • Duration - Flag the task if it finishes a certain amount of time before the programmed finish time (see Early Finish Duration). The task must have a specific finish time.
Early Finish Time	If Early Finish Type is Time, use this to specify the time before which the task finish time is considered early. That is, enter a time at which the task should still be running. Use hh:mm, 24-hour time.
Early Finish Duration	If Early Finish Type is Duration, use this to specify the shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Finished Early	Task instance only; system-supplied. This field is flagged if the task finished earlier than the time specified in the Early Finish fields.
Maximum Retries	User-defined. The maximum number of times Opwise should retry this task after it has started and gone to a failed state.
Task Priority	Task instance only; the priority of this task instance, as set by the user via the Set Priority command. Options are: HIGH, MEDIUM, LOW.
Retry Indefinitely	User-defined. Enabled or disabled. Indicates whether Opwise should continue trying indefinitely to run this task. If you enable this field, it overrides any value placed in the Maximum Retries field (above).
Retry Interval	User-defined. The number of seconds between each retry.

Current Retry Count	Task instance only; system-supplied. Displays, only for a running task instance, the current number of times that Opwise has retried the task after it first went to failure status.
First Time Ran	Task definition only; system-supplied. Displays after the first time the task runs. The date and time this task first ran.
Lowest Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. The shortest amount of time this task has taken to run.
Last Time Ran	Task definition only; system-supplied. Displays after the first time the task runs. The most recent date and time the task ran.
Average Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. Shows the average amount of time this task takes to run.
Number of Instances	Task definition only; system-supplied. Displays after the first time the task runs. Shows the number of times this task has run.
Highest Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. The longest amount of time this task has taken to run.
Last Instance Duration	Task definition only; system-supplied. Displays after the first time the task runs. The amount of time the task took to run the last time it ran.
Virtual Resource Priority	<p>Priority for acquiring a resource when two or more tasks are waiting for the resource. This priority applies to all resources required by the task.</p> <p>Options: 1 (high) - 20 (low).</p> <p>Default is 10.</p>
Hold Resources on Failure	If enabled, the task instance will continue to hold Renewable resources if the task instance fails. Renewable resources will be returned only if the task instance status is either Complete, Finished, or Skipped.
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Launch Task button	Manually launches the task.
View Instances button	Displays a list of task instances for which there has been a status change or a modification to the task instance record within the last 30 days (an Updated on Last 30 Days filter has been pre-selected for the list).
Delete button	Deletes the current record.
View Parent button	Task instance only; Displays this task's parent task (workflow), if any.

Show Details button	Task instance only; displays detailed information about this task instance.
Retrieve Output button	Task instance only; see Retrieving Output .
Skip button	Task instance only; for tasks loaded into the schedule that haven't run yet. Allows you to tell Opwise to skip this task. See Skipping a Task .
Hold button	Task instance only; see Putting a Task on Hold .
Force Finish button	Task instance only; see Force Finishing a Task .
Re-run button	Task instance only; see Re-running a Task .
Output tab	Task instance only. Displays output generated from the process, if any, based on specifications provided by the user in the Automatic Output Retrieval fields in the task definition.
Variables tab	Displays all variables associated with this record.
Actions tab	<p>Allows you to specify actions that Opwise will take automatically based on events that occur during the execution of this task. Events are task instance status, exit codes, late start, late finish, and early finish. Actions are:</p> <ul style="list-style-type: none"> • Abort Action - Abort the task if certain events occur. For details, see Setting Abort Actions. • Email Notification - Send an email if certain events occur. For details, see Creating Email Notifications. • Set Variable - Used in workflows to set a variable based on the occurrence of certain events. For details, see Setting Variables within a Workflow. • SNMP Notification - Send an email if certain events occur. For details, see Creating SNMP Notifications. • System Operation - Run an Opwise system operation based on specified conditions. For details, see Setting System Operations.
Task Virtual Resources tab	Lists Virtual Resources to which this task is assigned.
Mutually Exclusive Tasks tab	Displays all tasks that have been set to be mutually exclusive of this task.
Triggers tab	Displays a list of all triggers that have been defined to launch this task. Also allows you to add new triggers. If you add a new trigger from this location, Opwise automatically constructs a default trigger name as follows: <current task name>#TRIGGER#. You can change the default name if desired. For instructions on creating triggers, see Creating Triggers .
Notes tab	Displays all notes associated with this task.
Versions tab	Stores copies of all previous versions of the current record. See Record Versioning .

Specifying When a Task Runs

You can run the task as part of a [workflow](#), [specify triggers](#) that run the task automatically based on times or events, or [run the task manually](#).

Monitoring Task Execution

You can monitor all system activity from the [Activity screen](#) and can view activity history from the [Activity History screen](#).

Windows Task

- [Before You Begin](#)
- [Built-In Variables](#)
- [Creating a New Windows Task](#)
- [Windows Task Field Descriptions](#)
- [Specifying When a Task Runs](#)
- [Monitoring Task Execution](#)

Before You Begin

The Windows task allows you to run a platform-specific application on a Windows machine. To run a Windows task, you must first complete the following tasks:

- Install an Opswise Windows agent on a Windows machine
- Launch the agent. When the agent connects with the Controller, it automatically creates an [agent resource definition](#) in the database.
- Optionally, customize the agent heartbeat and log levels, as described in [Windows Agent Definition Field Descriptions](#).

Built-In Variables

The built-in variables outlined below can be used in a Windows task to pass data where appropriate:

- [Task and Task Instance Variables](#)
- [Script Variables](#)
- [Agent Variables](#)

Creating a New Windows Task

Step 1 From the navigation pane, select **Automation Center > Tasks > Windows Tasks**. The Windows Tasks List screen displays.

Step 2 Click **New**. The Windows Task Definition screen displays.

Step 3 Using the field descriptions provided below as a guide, complete the fields as needed.

Step 4 Click the **Submit** button to save the record and return to the menu, or right-click on the title bar and select **Save** to save the record and remain on the current display.


Step 5 If appropriate, repeat these steps for any additional tasks you want to add.

Task run statistics appear after the first time this task has been launched.


Windows Task Field Descriptions

The table below describes the fields, buttons, and tabs on the task definition and task instance screens. Color coding is provided that differentiates the following three types of fields:

- Fields that display on the task definition and task instance screens are shown in black.
- Fields that display only on the task definition screen are shown in green.
- Fields that display only on the task instance screen are shown in maroon.

Field Name	Description
Task/Instance Name	Required. Name used within Opwise to identify this task. Up to 40 alphanumeric; variables supported. It is the user's responsibility to develop a workable naming scheme for tasks.
Invoked by	<p>Task instance only; system-supplied. How the task instance was launched. One of the following:</p> <ul style="list-style-type: none"> • Trigger: (Trigger Name) - The instance was launched by the named trigger. • Workflow: (Workflow Name) - The instance was launched by the named workflow. • Manually Launched - The instance was launched by a user. To determine the name of the user: <ol style="list-style-type: none"> 1. From the Activity or Task Instances screen, click the task instance name to open the record. 2. The Execution User field identifies the user who launched the task instance.
Task	<p>Task instance only; system-supplied. Hover over the paper icon to display more information about the task instance. Click the paper icon to display the task definition record.</p>
Execution User	<p>Task instance only; system-supplied. If the task was launched manually, the ID of the user who launched it.</p>
Instance Reference Id	<p>Task instance only; system-supplied. Opwise increments this number each time the task is run.</p>
Credentials	<p>Credentials under which an Agent runs this task. These credentials override any credentials provided in the Agent resource definition for any Agent running this task.</p>
Version	<p>Task definition only; system-supplied. The version number of the current record, which is incremented by Opwise every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning.</p>
Credentials Variable	<p>Optional. If enabled, the Credentials field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: <code>\${variable name}</code>. The variable must be a supported type as described in Variables and Functions.</p>
Run with Highest Privileges	<p>This option must be enabled in order to execute the task using an elevated privileges token, rather than one subject to User Account Control (UAC) restrictions. An elevated token allows a process to execute with all the privileges available to its specified credentials. For example, a task executed with an administrative account will behave as though it received permission via a UAC dialog to perform a privileged operation.</p> <p>This option will not give a user account privileges that have are not already granted to it. For example, taking ownership of a file is a privileged operation by default. A task will still fail even with this option selected if it is run with a regular user account that has not been granted the ability to change file ownership.</p> <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note This option will only affect tasks executed on Windows systems that support User Account Control (UAC). It will have no affect on tasks run on Windows releases prior to Vista (for example, Windows XP, Server 2003).</p> <p>When this option is selected, tasks will execute only if the agent is 5.1.0.10 or higher. Otherwise, the task will get a Start Failure.</p> </div>
Agent	<p>Optional. The name of the agent resource definition that identifies the machine where the operation will run. If you do not specify an agent, you must specify an agent cluster (see below).</p>

Agent Cluster	Optional. You can specify an agent cluster in addition to or in place of a specific agent. An agent cluster is a group of agents, one of which Opwise will choose to run this task. If you specify an agent and an agent cluster, Opwise Automation Center first tries to run the task on the specific agent. If the agent is not available, Opwise reverts to the agent cluster. See Agent Clusters for more information.
Agent Variable	Optional. If enabled, the Agent field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: \${variable name}. The variable must be a supported type as described in Variables and Functions .
Agent Cluster Variable	Optional. If enabled, the Agent Cluster field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: \${variable name}. The variable must be a supported type as described in Variables and Functions .
Hold on Start	If enabled, when the task is launched it appears in the Activity display with a status of Held. The task runs when the user releases it.
Hold Reason	Information about why the task will be put on hold when it starts.
Cluster Broadcast	Task definition only; optional. You can specify a Cluster Broadcast in place of a specific agent and/or Agent Cluster. When you specify an agent cluster in the Cluster Broadcast field, Opwise Automation Center runs the task on all the agents in the cluster. Each instance of the task running on its own agent becomes a separate task instance record in the database and displays separately in the Activity monitor. See Agent Clusters for more information about defining agent clusters.
Task Description	User-supplied description of this record.
Status	Task instance only; system-supplied. See Task Instance Statuses .
Exit Code	Task instance only; system-supplied. The exit code captured by the agent when executing the task (for example, a command or script).
Status Description	Task instance only; system-supplied. Provides additional information, if any, about the status of the task.
Queued Time	Task instance only; system-supplied. The time the task was queued for processing.
Process ID	Task instance only; system-supplied. The ID of the process that was launched.
Start Time	Task instance only; system-supplied. The date and time the task started.
CPU Time	Task instance only; system-supplied. The amount of CPU time the task took to run.
End Time	Task instance only; system-supplied. The date and time the task instance completed.
Duration	Task instance only; system-supplied. The amount of time the task took to run.

Member of Business Services	User-defined. Allows you to select one or more Business Services that this record definition belongs to. Click the lock icon to unlock the field and select Business Services .
User Estimated Duration	Task definition only; optional. The estimated amount of time it should normally take to run this task. Opwise uses this information to calculate the User Estimated End Time on a task instance record.
Command or Script	Specifies whether a single command or a script is being executed. Options: <ul style="list-style-type: none"> • Command (default) • Script
Command	Required (if Command is selected in Command or Script field). Command being executed on the remote machine. Variables supported.
Script	Required (if Script is selected in Command or Script field). Name of the script that has been uploaded into the Script Library and will be executed by this task.
Parameters	Optional. Any arguments needed by the program to execute properly. Variables supported.
Runtime Directory	Optional. The directory from which the application should be executed. Variables supported.
Interact with Desktop	This option must be enabled for a task that runs an application with a GUI requiring some manual actions from a user (for example, clicking buttons or entering values). <div data-bbox="423 1058 1446 1266" style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note This option is effective only for tasks executed on Windows XP or Server 2003. Windows Vista introduced the desktop isolation feature, which prevents tasks from accessing the interactive desktop session on Vista, Windows 7, and Server 2008. The Windows agent will execute the task, but the Interact with Desktop option has no effect. Therefore, an interactive application's GUI will not be visible on those platforms.</p> </div>
Exit Code Processing	Required. Specifies how Opwise Automation Center should determine whether the executed command failed or completed successfully. Options: <ul style="list-style-type: none"> • Success Exitcode Range - The command is considered completed successfully if its exit code falls within the range specified in the Exit Codes field (see below). • Failure Exitcode Range - The command is considered failed if its exit code falls within the range specified in the Exit Codes field (see below). • Success Output Contains - The command is considered completed successfully if its output contains the text specified in the Scan Output For field (see below). • Failure Output Contains - The command is considered failed if its output contains the text specified in the Scan Output For field (see below). • Step Conditions - The command is considered completed successfully/failed if any of its specified condition codes falls within the range specified under the Step Conditions tab (see Creating Step Conditions).
Output Type	Required if Exit Code Processing = Success Output Contains or Failure Output Contains. This field specifies the type of output. Options: <ul style="list-style-type: none"> • Standard Output (STDOUT) • Standard Error (STDERR) • File

Exit Codes	Required if Exit Code Processing = Success Exitcode Range or Failure Exit Code Range. This field specifies the range. Format: Numeric. Use commas to list a series of exitcodes; use hyphens to specify a range. Example: 1,5, 22-30.
Scan Output For	Required if Exit Code Processing = Success Output Contains or Failure Output Contains. This field specifies the text for which Opwise should scan the output file. Opwise will process this field as a regular expression.
Output File (Exit Code Processing)	Required if Output Type = File. This field specifies the path and file name of the output file that should be scanned for the text in the Scan Output For field.
(Environment Variables) Name and Value	Optional. Allows you to enter environment variables needed by the program to run. For each variable, enter a Name and Value, and then click Add . You can add a maximum of 4,000 characters for the combined Names and Values of all variables. The variable is listed in the space underneath. To delete a variable, click the X button.
Environment Variables List	Displays - on the Windows Tasks List screen - any environment variables added to this task.
Automatic Output Retrieval	Optional. Allows you to specify whether you want Opwise to automatically retrieve any output from the job and attach it to the task instance record. Options: <ul style="list-style-type: none"> • None - Do not attach any output to the task instance record. • Standard Output - Attach all standard output. • Standard Error - Attach standard error output. • File - Attach the file specified in the Output File field.
Output File (Automatic Output Retrieval)	Required if Automatic Output Retrieval=File. This field specifies the path and filename containing the output that you want automatically retrieved and attached to the task instance.
Start Line	Optional. Allows you to instruct Opwise to retrieve data beginning at the line indicated. If a Start Line value is not specified on the screen, the default is 1.
Number of Lines	Optional. Allows you to limit the retrieved data to the number of lines specified. If a Number of Lines value is not specified, the default is the value of the Retrieve Output Default Maximum Lines Opwise system property.
Scan Text	Optional. Instructs Opwise to scan the data for the text specified and retrieve only that. Opwise will process this field as a regular expression.
Late Start	If enabled, and if the task instance starts after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late start (see Late Start Type). To determine whether a task instance started late, open the task instance and locate the Started Late field; the field is checked if the instance started after the specified time. This field only appears on the task instance if the user specified a Late Start in the task definition.
Late Start Type	Required if Late Start is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it starts after the specified time. • Duration - Flag the task if it starts a certain amount of time after the programmed start time. The task must have a specific start time.
Late Start Time	Time after which the task start time is considered late. Use hh:mm, 24-hour time

Late Start Duration	<p>Duration (amount of relative time) after which the task is considered to have started late. For a task within a workflow, the duration is the period between the time the workflow starts and the time the task itself starts. For example, a task might have a Late Start Duration of 60 minutes. If the workflow starts at 9:00 a.m. but the task itself does not start until 10:30, the task has started late.</p> <p>For a task that is not within a workflow, Late Start Duration has meaning only if the task has been held upon starting. For example, if a task has a Late Start Duration of 60 minutes and the Hold on Start field is enabled, if the task is not released from hold within the amount of time specified in the Late Start Duration field, the task has started late.</p>
Started Late	<p>Task instance only; system-supplied. This field is flagged if the task started later than the time specified in the Late Start fields.</p>
Late Finish	<p>If enabled, and if the task instance finishes after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late finish (see Late Finish Type). To determine whether a task instance finished late, open the task instance and locate the Finished Late field; the field is checked if the instance finished after the specified time or lasted longer than expected. This field only appears on the task instance if the user specified a Late Finish in the task definition.</p>
Late Finish Type	<p>Required if Late Finish is enabled. Options are:</p> <ul style="list-style-type: none"> • Time - Flag the task if it finishes after the specified time (see Late Finish Time). • Duration - Flag the task if it finishes a certain amount of time after the programmed finish time (see Late Finish Duration). The task must have a specific finish time.
Late Finish Time	<p>If Late Finish Type is Time, use this to specify the time after which the task finish time is considered late. Use hh:mm, 24-hour time.</p>
Late Finish Duration	<p>If Late Finish Type is Duration, use this to specify the longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.</p>
Finished Late	<p>Task instance only; system-supplied. This field is flagged if the task finished later than the time or duration specified in the Late Finish fields.</p>
Early Finish	<p>If enabled, and if the task instance finishes before the time or period specified, the task instance is flagged as early. You can specify a time or duration to determine an early finish (see Early Finish Type). To determine whether a task instance finished early, open the task instance and locate the Finished Early field; the field is checked if the instance finished before the specified time or did not last as long as expected. This field only appears on the task instance if the user added Early Finish specifications to the task definition.</p>
Early Finish Type	<p>Required if Early Finish is enabled. Options are:</p> <ul style="list-style-type: none"> • Time - Flag the task if it finishes before the specified time (see Early Finish Time). • Duration - Flag the task if it finishes a certain amount of time before the programmed finish time (see Early Finish Duration). The task must have a specific finish time.
Early Finish Time	<p>If Early Finish Type is Time, use this to specify the time before which the task finish time is considered early. That is, enter a time at which the task should still be running. Use hh:mm, 24-hour time.</p>
Early Finish Duration	<p>If Early Finish Type is Duration, use this to specify the shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.</p>

Finished Early	Task instance only; system-supplied. This field is flagged if the task finished earlier than the time specified in the Early Finish fields.
Maximum Retries	User-defined. The maximum number of times Opwise should retry this task after it has started and gone to a failed state.
Retry Indefinitely	User-defined. Enabled or disabled. Indicates whether Opwise should continue trying indefinitely to run this task. If you enable this field, it overrides any value placed in the Maximum Retries field (above).
Retry Interval	User-defined. The number of seconds between each retry.
Current Retry Count	Task instance only; system-supplied. Displays, only for a running task instance, the current number of times that Opwise has retried the task after it first went to failure status.
First Time Ran	Task definition only; system-supplied. Displays after the first time the task runs. The date and time this task first ran.
Lowest Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. The shortest amount of time this task has taken to run.
Last Time Ran	Task definition only; system-supplied. Displays after the first time the task runs. The most recent date and time the task ran.
Average Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. Shows the average amount of time this task takes to run.
Number of Instances	Task definition only; system-supplied. Displays after the first time the task runs. Shows the number of times this task has run.
Highest Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. The longest amount of time this task has taken to run.
Last Instance Duration	Task definition only; system-supplied. Displays after the first time the task runs. The amount of time the task took to run the last time it ran.
Virtual Resource Priority	Priority for acquiring a resource when two or more tasks are waiting for the resource. This priority applies to all resources required by the task. Options: 1 (high) - 20 (low). Default is 10.
Hold Resources on Failure	If enabled, the task instance will continue to hold Renewable resources if the task instance fails. Renewable resources will be returned only if the task instance status is either Complete, Finished, or Skipped.
Submit button	Submits the new record to the database.

Update button	Saves updates to the record.
Launch Task button	Manually launches the task.
View Instances button	Displays a list of task instances for which there has been a status change or a modification to the task instance record within the last 30 days (an Updated on Last 30 Days filter has been pre-selected for the list).
Delete button	Deletes the current record.
View Parent button	Task instance only; Displays this task's parent task (workflow), if any.
Show Details button	Task instance only; displays detailed information about this task instance.
Retrieve Output button	Task instance only; see Retrieving Output .
Skip button	Task instance only; for tasks loaded into the schedule that haven't run yet. Allows you to tell Opwise to skip this task. See Skipping a Task .
Hold button	Task instance only; see Putting a Task on Hold .
Force Finish button	Task instance only; see Force Finishing a Task .
Re-run button	Task instance only; see Re-running a Task .
Output tab	Task instance only. Displays output generated from the process, if any, based on specifications provided by the user in the Automatic Output Retrieval fields in the task definition.
Variables tab	Displays all variables associated with this record.
Actions tab	<p>Allows you to specify actions that Opwise will take automatically based on events that occur during the execution of this task. Events are task instance status, exit codes, late start, late finish, and early finish. Actions are:</p> <ul style="list-style-type: none"> • Abort Action - Abort the task if certain events occur. For details, see Setting Abort Actions. • Email Notification - Send an email if certain events occur. For details, see Creating Email Notifications. • Set Variable - Used in workflows to set a variable based on the occurrence of certain events. For details, see Setting Variables within a Workflow. • SNMP Notification - Send an email if certain events occur. For details, see Creating SNMP Notifications. • System Operation - Run an Opwise system operation based on specified conditions. For details, see Setting System Operations.
Task Virtual Resources tab	Lists Virtual Resources to which this task is assigned.
Mutually Exclusive Tasks tab	Displays all tasks that have been set to be mutually exclusive of this task.

Triggers tab	Displays a list of all triggers that have been defined to launch this task. Also allows you to add new triggers. If you add a new trigger from this location, Opswise automatically constructs a default trigger name as follows: <current task name>#TRIGGER#. You can change the default name if desired. For instructions on creating triggers, see Creating Triggers .
Notes tab	Displays all notes associated with this task.
Versions tab	Stores copies of all previous versions of the current record. See Record Versioning .

Specifying When a Task Runs

You can run the task as part of a [workflow](#), [specify triggers](#) that run the task automatically based on times or events, or [run the task manually](#).

Monitoring Task Execution

You can monitor all system activity from the [Activity screen](#) and can view activity history from the [Activity History screen](#).

zOS Task

- [Before You Begin](#)
- [Built-In Variables](#)
- [Creating a New z/OS Task](#)
- [z/OS Task Field Descriptions](#)
- [Specifying When a Task Runs](#)
- [Monitoring Task Execution](#)
- [Special Processing on z/OS Tasks](#)
 - [Using Variables in JCL and In-Stream Data Sets](#)
 - [Skipping Steps during Initial Run](#)
 - [Overriding Key JCL Parameters from Opwise](#)
 - [Disabling Automatic Data Set Deletion](#)
 - [Rerunning a z/OS Task](#)
 - [Interactively Ignoring a Stepcode to Force a Task to Complete](#)
 - [CA7/CA11 Toleration](#)
 - [Viewing Ops Rerun Reports](#)
 - [Viewing Audit Trails on a Restart](#)

Before You Begin

The z/OS task allow you to run a platform-specific application on a z/OS machine. To run a z/OS task, you must first complete the following tasks:

- [Install an Opwise z/OS agent on a z/OS machine.](#)
- [Launch the agent. When the agent connects with the Controller, it automatically creates a \[z/OS agent resource definition\]\(#\) in the database.](#)
- [Optionally, customize the agent heartbeat and log levels, as described in \[z/OS Agent Definition Field Descriptions\]\(#\).](#)

Built-In Variables

The built-in variables outlined below can be used in a z/OS task to pass data where appropriate:

- [Task and Task Instance Variables](#)
- [Agent Variables](#)

Creating a New z/OS Task

Step 1 From the navigation pane, select **Automation Center > Tasks > z/OS Tasks**. The z/OS Tasks List screen displays.

Step 2 Click **New**. The z/OS Task Definition screen displays.

Step 3 Using the field descriptions provided below as a guide, complete the fields as needed.

Step 4 Click the **Submit** button to save the record and return to the menu, or right-click on the title bar and select **Save** to save the record and remain on the current display.

Step 5 If appropriate, repeat these steps for any additional tasks you want to add.

Task run statistics appear after the first time this task has been launched.

z/OS Task Field Descriptions

The table below describes the fields, buttons, and tabs on the task definition and task instance screens. Color coding is provided that differentiates the following three types of fields:

- Fields that display on the task definition and task instance screens are shown in black.
- Fields that display only on the task definition screen are shown in green.
- Fields that display only on the task instance screen are shown in maroon.

Field Name	Description
Task/Instance Name	Required. Name used within Opwise to identify this task. Up to 40 alphanumeric; variables supported. It is the user's responsibility to develop a workable naming scheme for tasks.

Invoked by	<p>Task instance only; system-supplied. How the task instance was launched. One of the following:</p> <ul style="list-style-type: none"> • Trigger: (Trigger Name) - The instance was launched by the named trigger. • Workflow: (Workflow Name) - The instance was launched by the named workflow. • Manually Launched - The instance was launched by a user. To determine the name of the user: <ol style="list-style-type: none"> 1. From the Activity or Task Instances screen, click the task instance name to open the record. 2. The Execution User field identifies the user who launched the task instance.
Task	<p>Task instance only; system-supplied. Hover over the paper icon to display more information about the task instance. Click the paper icon to display the task definition record.</p>
Execution User	<p>Task instance only; system-supplied. If the task was launched manually, the ID of the user who launched it.</p>
Instance Reference Id	<p>Task instance only; system-supplied. Opwise increments this number each time the task is run.</p>
Credentials	<p>Credentials under which an Agent runs this task. These credentials override any credentials provided in the Agent resource definition for any Agent running this task.</p>
Version	<p>Task definition only; system-supplied. The version number of the current record, which is incremented by Opwise every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning.</p>
Credentials Variable	<p>Optional. If enabled, the Credentials field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: \${variable name}. The variable must be a supported type as described in Variables and Functions.</p>
Agent	<p>Required. The name of the agent resource definition that identifies the machine where the operation will run.</p>
New Jobname	<p>Optional. Jobname that will replace the one in the JCL member. This allows you to override the value in your JCL from Opwise without having to modify the JCL.</p>
Agent Variable	<p>Optional. If enabled, the Agent field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: \${variable name}. The variable must be a supported type as described in Variables and Functions.</p>
New Jobclass	<p>New Jobclass to replace the one in the JCL member. This allows you to override the value in your JCL from Opwise without having to modify the JCL.</p>
Hold on Start	<p>If enabled, when the task is launched it appears in the Activity display with a status of Held. The task runs when the user releases it.</p>
New Msgclass	<p>Optional. New MSGCLASS to replace the one in the JCL member. This allows you to override the value in your JCL from Opwise without having to modify the JCL.</p>

PROCLIB	<p>Optional. When you use this parameter to specify the new value "PROC001," the submitted JCL will be modified to show the following statement:</p> <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <pre style="font-family: monospace;">/*JESPARM PROCLIB=PROC001</pre> </div> <p>This allows you to alter the default PROCLIB used by JES2 so that Opwise supplies your catalogued procedures and you do not have to modify the JCL.</p>
Schedule ID	<p>Optional. The CA7 Schedule ID; for CA7 toleration only. Click here for details.</p>
Hold Reason	<p>Information about why the task will be put on hold when it starts.</p>
Task Description	<p>User-supplied description of this record.</p>
Member of Business Services	<p>User-defined. Allows you to select one or more Business Services that this record definition belongs to. Click the lock icon to unlock the field and select Business Services.</p>
Status	<p>Task instance only; system-supplied. See Task Instance Statuses.</p>
Exit Code	<p>Task instance only; system-supplied. The exit code captured by the agent when executing the task (for example, a command or script).</p>
Status Description	<p>Task instance only; system-supplied. Provides additional information, if any, about the status of the task.</p>
Queued Time	<p>Task instance only; system-supplied. The time the task was queued for processing.</p>
CPU Time	<p>Task instance only; system-supplied. The amount of CPU time the task took to run.</p>
Start Time	<p>Task instance only; system-supplied. The date and time the task started.</p>
Duration	<p>Task instance only; system-supplied. The amount of time the task took to run.</p>
End Time	<p>Task instance only; system-supplied. The date and time the task instance completed.</p>
User Estimated Duration	<p>Task definition only; optional. The estimated amount of time it should normally take to run this task. Opwise uses this information to calculate the User Estimated End Time on a task instance record.</p>

JCL Location	<p>Required. The file and member name containing the JCL script.</p> <p>When you are using the JCL_LIBRARY feature, you can substitute the name of the library with a string starting with "&", that names the library specified in the uags.conf file with the JCL_library definitions. For example, the name of a job might look like the following:</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <pre>&PRODLIB(PAYJOB01)</pre> </div>
Exit Code Processing	<p>Required. Specifies how Opwise Automation Center should determine whether the executed command failed or completed successfully. Options:</p> <ul style="list-style-type: none"> • Success Exitcode Range - The command is considered completed successfully if its exit code falls within the range specified in the Exit Codes field (see below). • Failure Exitcode Range - The command is considered failed if its exit code falls within the range specified in the Exit Codes field (see below). • Success Output Contains - The command is considered completed successfully if its output contains the text specified in the Scan Output For field (see below). • Failure Output Contains - The command is considered failed if its output contains the text specified in the Scan Output For field (see below). • Step Conditions - The command is considered completed successfully/failed if any of its specified condition codes falls within the range specified under the Step Conditions tab (see Creating Step Conditions).
Exit Codes	<p>Required if Exit Code Processing = Success Exitcode Range or Failure Exit Code Range. This field specifies the range. Format: Numeric. Use commas to list a series of exitcodes; use hyphens to specify a range. Example: 1,5, 22-30.</p>
Scan Output For	<p>Required if Exit Code Processing = Success Output Contains or Failure Output Contains. This field specifies the text for which Opwise should scan the output file. Opwise will process this field as a regular expression.</p>
Output Type	<p>Required if Exit Code Processing = Success Output Contains or Failure Output Contains. This field specifies the type of output. Options:</p> <ul style="list-style-type: none"> • Standard Output (STDOUT) • Standard Error (STDERR) • File
Output File (Exit Code Processing)	<p>Required if Output Type = File. This field specifies the path and file name of the output file that should be scanned for the text in the Scan Output For field.</p>

Parameters -z/OS	<p>Optional. Allows you to enter parameters that will be inserted into the JCL. The parameter consists of a keyword and a value, commonly referred to as a key-value pair. You can enter as many key-value pairs as needed. The parameters you enter each create a separate JCL construct called the SET command. Each one appears as a new line inserted dynamically into the JCL submitted to Opwise for the current execution. The JCL is not permanently modified.</p> <p>For example, you might specify the key-value pair of RUNTYPE=PROD. This results in the following JCL SET statement being inserted in the job after the job card:</p> <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <pre>// SET RUNTYPE=PROD</pre> </div> <p>To enter a key-value pair, type in the name of the key and the value, and click Add. Opwise displays the parameter in the space underneath. You can add as many parameters as needed. To delete an entry, click the X button.</p> <p>The parameters fields also support two additional special functions:</p> <ul style="list-style-type: none"> • They allow you to specify any steps you want skipped during the job run. See Skipping Steps during Initial Run for detailed instructions. • They allow you to add data to DD* input streams. See Using Variables in JCL and In-Stream Data Sets for detailed instructions.
Parameters List	Displays - on the z/OS Tasks List screen - the contents of the Parameters field.
Automatic Output Retrieval - z/OS	<p>Optional. Allows you to specify whether you want Opwise to automatically retrieve output from the job and attach it to the task instance record. Options:</p> <ul style="list-style-type: none"> • None - Do not attach any output to the task instance record. • File - Attach the file specified in the Output File field. • Joblog - Attach output from the z/OS joblog.
Output File (Automatic Output Retrieval)	Required if Automatic Output Retrieval=File. This field specifies the path and filename containing the output that you want automatically retrieved and attached to the task instance.
Start Line	Optional. Allows you to instruct Opwise to retrieve data beginning at the line indicated. If a Start Line value is not specified on the screen, the default is 1.
Number of Lines	Optional. Allows you to limit the retrieved data to the number of lines specified. If a Number of Lines value is not specified, the default is the value of the Retrieve Output Default Maximum Lines Opwise system property.
Scan Text	Optional. Instructs Opwise to scan the data for the text specified and retrieve only that. Opwise will process this field as a regular expression.
Task Priority	Task instance only; the priority of this task instance, as set by the user via the Set Priority command. Options are: HIGH, MEDIUM, LOW.
Late Start	If enabled, and if the task instance starts after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late start (see Late Start Type). To determine whether a task instance started late, open the task instance and locate the Started Late field; the field is checked if the instance started after the specified time. This field only appears on the task instance if the user specified a Late Start in the task definition.
Late Start Type	<p>Required if Late Start is enabled. Options are:</p> <ul style="list-style-type: none"> • Time - Flag the task if it starts after the specified time. • Duration - Flag the task if it starts a certain amount of time after the programmed start time. The task must have a specific start time.

Late Start Time	Time after which the task start time is considered late. Use hh:mm, 24-hour time
Late Start Duration	<p>Duration (amount of relative time) after which the task is considered to have started late. For a task within a workflow, the duration is the period between the time the workflow starts and the time the task itself starts. For example, a task might have a Late Start Duration of 60 minutes. If the workflow starts at 9:00 a.m. but the task itself does not start until 10:30, the task has started late.</p> <p>For a task that is not within a workflow, Late Start Duration has meaning only if the task has been held upon starting. For example, if a task has a Late Start Duration of 60 minutes and the Hold on Start field is enabled, if the task is not released from hold within the amount of time specified in the Late Start Duration field, the task has started late.</p>
Started Late	Task instance only; system-supplied. This field is flagged if the task started later than the time specified in the Late Start fields.
Late Finish	If enabled, and if the task instance finishes after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late finish (see Late Finish Type). To determine whether a task instance finished late, open the task instance and locate the Finished Late field; the field is checked if the instance finished after the specified time or lasted longer than expected. This field only appears on the task instance if the user specified a Late Finish in the task definition.
Late Finish Type	<p>Required if Late Finish is enabled. Options are:</p> <ul style="list-style-type: none"> • Time - Flag the task if it finishes after the specified time (see Late Finish Time). • Duration - Flag the task if it finishes a certain amount of time after the programmed finish time (see Late Finish Duration). The task must have a specific finish time.
Late Finish Time	If Late Finish Type is Time, use this to specify the time after which the task finish time is considered late. Use hh:mm, 24-hour time.
Late Finish Duration	If Late Finish Type is Duration, use this to specify the longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Finished Late	Task instance only; system-supplied. This field is flagged if the task finished later than the time or duration specified in the Late Finish fields.
Early Finish	If enabled, and if the task instance finishes before the time or period specified, the task instance is flagged as early. You can specify a time or duration to determine an early finish (see Early Finish Type). To determine whether a task instance finished early, open the task instance and locate the Finished Early field; the field is checked if the instance finished before the specified time or did not last as long as expected. This field only appears on the task instance if the user added Early Finish specifications to the task definition.
Early Finish Type	<p>Required if Early Finish is enabled. Options are:</p> <ul style="list-style-type: none"> • Time - Flag the task if it finishes before the specified time (see Early Finish Time). • Duration - Flag the task if it finishes a certain amount of time before the programmed finish time (see Early Finish Duration). The task must have a specific finish time.
Early Finish Time	If Early Finish Type is Time, use this to specify the time before which the task finish time is considered early. That is, enter a time at which the task should still be running. Use hh:mm, 24-hour time.
Early Finish Duration	If Early Finish Type is Duration, use this to specify the shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.

Finished Early	Task instance only; system-supplied. This field is flagged if the task finished earlier than the time specified in the Early Finish fields.
Maximum Retries	User-defined. The maximum number of times Opwise should retry this task after it has started and gone to a failed state.
Retry Indefinitely	User-defined. Enabled or disabled. Indicates whether Opwise should continue trying indefinitely to run this task. If you enable this field, it overrides any value placed in the Maximum Retries field (above).
Retry Interval	User-defined. The number of seconds between each retry.
Current Retry Count	Task instance only; system-supplied. Displays, only for a running task instance, the current number of times that Opwise has retried the task after it first went to failure status.
User Estimated End Time	Task instance only; system-supplied. If the user entered information into the User Estimated Duration field in the task definition, Opwise uses this information to calculate an end time for the task instance, based on the date/time the task instance started.
Shortest Estimated End Time	Task instance only; system-supplied.
Average Estimated End Time	Task instance only; system-supplied.
Longest Estimated End Time	Task instance only; system-supplied.
Virtual Resource Priority	Priority for acquiring a resource when two or more tasks are waiting for the resource. This priority applies to all resources required by the task. Options: 1 (high) - 20 (low). Default is 10.
Hold Resources on Failure	If enabled, the task instance will continue to hold Renewable resources if the task instance fails. Renewable resources will be returned only if the task instance status is either Complete, Finished, or Skipped.
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Launch Task button	Manually launches the task.
View Instances button	Displays a list of task instances for which there has been a status change or a modification to the task instance record within the last 30 days (an Updated on Last 30 Days filter has been pre-selected for the list).
Delete button	Deletes the current record.

View Parent button	Task instance only; Displays this task's parent task (workflow), if any.
Show Details button	Task instance only; displays detailed information about this task instance.
Retrieve Output button	Task instance only; see Retrieving Output .
Skip button	Task instance only; for tasks loaded into the schedule that haven't run yet. Allows you to tell Opwise to skip this task. See Skipping a Task .
Hold button	Task instance only; see Putting a Task on Hold .
Force Finish button	Task instance only; see Force Finishing a Task .
Re-run button	Task instance only; see Re-running a Task .
Output tab	Task instance only. Displays output generated from the process, if any, based on specifications provided by the user in the Automatic Output Retrieval fields in the task definition.
z/OS Restartable JobSteps tab	Task instance only; see Rerunning a z/OS Task and Confirming any JCL Changes .
z/OS Restart Confirmation tab	Task instance only; see Rerunning a zOS Task and Confirming any JCL Changes .
Step Conditions tab	Displays a list of all step conditions defined for this task.
Variables tab	Displays all variables associated with this record.
Actions tab	Allows you to specify actions that Opwise will take automatically based on events that occur during the execution of this task. Events are task instance status, exit codes, late start, late finish, and early finish. Actions are: <ul style="list-style-type: none"> • Abort Action - Abort the task if certain events occur. For details, see Setting Abort Actions. • Email Notification - Send an email if certain events occur. For details, see Creating Email Notifications. • Set Variable - Used in workflows to set a variable based on the occurrence of certain events. For details, see Setting Variables within a Workflow. • SNMP Notification - Send an email if certain events occur. For details, see Creating SNMP Notifications. • System Operation - Run an Opwise system operation based on specified conditions. For details, see Setting System Operations.
Task Virtual Resources tab	Lists Virtual Resources to which this task is assigned.
Mutually Exclusive Tasks tab	Displays all tasks that have been set to be mutually exclusive of this task.

Triggers tab	Displays a list of all triggers that have been defined to launch this task. Also allows you to add new triggers. If you add a new trigger from this location, Opwise automatically constructs a default trigger name as follows: <current task name>#TRIGGER#. You can change the default name if desired. For instructions on creating triggers, see Creating Triggers .
Notes tab	Displays all notes associated with this task.
Versions tab	Stores copies of all previous versions of the current record. See Record Versioning .

Specifying When a Task Runs

You can run the task as part of a [workflow](#), [specify triggers](#) that run the task automatically based on times or events, or [run the task manually](#).

Monitoring Task Execution

You can monitor all system activity from the [Activity screen](#) and can view activity history from the [History screen](#).

Special Processing on z/OS Tasks

The following special processing features are available for running z/OS tasks:

- [Using Variables in JCL and In-Stream Data Sets](#)
- [Skipping Steps during Initial Run](#)
- [Overriding Key JCL Parameters from Opwise](#)

The following failure processing features are available for handling job failures:

- [Rerunning a z/OS Task](#)
- [Interactively Ignoring a Stepcode to Force a Task to Complete](#)

Opwise supports compatibility with other schedulers:

- [CA7/CA11 Toleration](#)

Opwise provides the following reports that track error processing:

- [Viewing Ops Rerun Reports](#)
- [Viewing Audit Trails on a Restart](#)

Each of these features is described in detail below.

Using Variables in JCL and In-Stream Data Sets

There are two categories of variables that can be defined as part of the z/OS task definition:

- JCL Symbolic Parameters
- Opwise Parameters

Parameter values can use Opwise built-in or user defined variables.

JCL Symbolic Parameters

Use the z/OS Task definition Add Parameter input fields to specify JCL symbolic parameters to be used in the JCL. Any parameter name that does not start with *@ is considered a JCL symbolic parameter. JCL symbolic parameters result in Opwise adding a JCL SET statement to the JCL before the first step EXEC statement.

As an example, a z/OS Task definition parameter name of PHLQ and value of APP.PROD will result in the following JCL SET statement being added to the JCL:

```
// SET PHLQ=APP.PROD
```

The PHLQ symbolic parameter in the example above can then be used in the remaining JCL as described by the IBM JCL Reference.

Opswise Parameters

Use the z/OS Task definition Add Parameters input fields to specify parameters that can be used in any JCL statement and in in-stream data sets. In-stream data sets are typically defined with a DD * JCL statement. Opswise will substitute the parameter values in the JCL statements and in the in-stream data before the JCL is submitted to JES.

Opswise parameters are defined with a parameter name that starts with the character sequence *@. Opswise parameters are referenced in the JCL and in-stream data by prefixing the parameter name with the @ character.

The following steps add an Opswise parameter with the name DATE1 and a value of 20110601:

Step 1	Open the z/OS task definition.
Step 2	<p>Use the Add Parameters input fields to add the first parameter in the following format:</p> <ul style="list-style-type: none"> Parameter name is the name of a variable preceded with @. For example: @DATE1. Parameter value is the value you want to set to the variable. For example: 20110601. <p>For example, you could specify the following in the Add Parameters field:</p> <div style="border: 1px solid gray; padding: 10px; margin: 10px 0;"> <p>Member of Business Services: <input type="checkbox"/></p> <p>User Estimated Duration: <input type="text" value="00"/> : <input type="text" value="00"/> : <input type="text" value="00"/> hh:mm:ss</p> <p>JCL Location: <input type="text" value="LAM1A.JCL.CNTL(UCM#202)"/></p> <p>Exit Code Processing: <input type="text" value="Success Exitcode Range"/></p> <p>Exit Codes: <input type="text" value="0"/></p> <p>Add parameters by inputing the name and value and clicking "Add":</p> <p>Name: <input type="text" value="*@DATE1"/> Value: <input type="text" value="20110601"/> <input type="button" value="Add"/></p> <p>Automatic Output Retrieval: <input type="text" value="Joblog"/></p> <p>Start Line: <input type="text" value="1"/></p> <p>Number of Lines: <input type="text" value="1000"/></p> <p>Scan Text: <input type="text"/></p> </div>

Step 3 Click the **Add** button. The new parameter is added to the parameter list; the screen looks like this:

Step 4 When the JCL is submitted for execution, the parameter DATE1 (shown in the following example) will be substituted with the value 20110601 in the JCL or in any in-stream data. The example also shows the DATE1 parameter being used in an in-stream data and in a JCL IF statement:

```
//INPUT DD *
@DATE1
/*
```

```
//AIF IF @DATE1 > 20110101 THEN
```

If an Opwise parameter must be concatenated with a non-space character, end the parameter name with a period (.). The example below uses the DATE1 parameter concatenated with a non-space character in an in-stream data set:

```
//INPUT DD *
DATE@DATE1.ACT9898
/*
```

Skipping Steps during Initial Run

You can specify in a z/OS task that one or more steps from the JCL should be skipped when Opwise launches the job. You achieve this by adding SKIPSTNN variables (or parameters) to your z/OS task record.

To configure your z/OS task to skip specific JCL steps:

Step 1 Open the task record.

Step 2	<p>Use the add parameters feature to add the first parameter in the following format:</p> <div data-bbox="261 210 1445 296" style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <pre>SKIPSTAA=STEPNAME</pre> </div> <ul style="list-style-type: none"> • SKIPST is a required string. • AA is any unique combination of alphanumerics, used only to make this SKIPST command unique. (You can add as many SKIPST commands as needed.) • STEPNAME is the JCL step name.
Step 3	<p>Click the Add button. The new parameter is added to your parameter list.</p>
Step 4	<p>Repeat the above step for each step you want to skip. Change the AA portion of the SKIPST parameter for each parameter you add. That is, each parameter name must be unique.</p> <p>In the following example, two skipstep parameters have been added to the record, instructing Opwise to skip JCL STEP03 and STEP05.</p> <div data-bbox="224 632 1224 919" style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p>Add parameters by inputting the name and value and clicking "Add":</p> <p>Name: <input type="text"/> Value: <input type="text"/> <input type="button" value="Add"/></p> <ul style="list-style-type: none"> • SKIPST01=STEP03 <input type="button" value="X"/> • SKIPST02=STEP05 <input type="button" value="X"/> </div>
Step 5	<p>To delete a parameter, click the X.</p>

Overriding Key JCL Parameters from Opwise

When you launch a z/OS task from Opwise, you can specify a different Jobname, Jobclass, Msgclass, Schedule ID or add a JOBPARM card. This enables you to run your JCL jobs from Opwise without having to go in and modify your JCL. You can do so by entering the new value into the appropriate field on the z/OS Task definition screen. See the screen shot under [Creating a New z/OS Task](#) for the location of these fields.

Disabling Automatic Data Set Deletion

[Universal Automation Center Agent \(UAG\)](#) will automatically detect and delete data sets that would cause a **NOT CATLGD 2** condition. The data set deletion takes place before the job is started. Starting with UAG 5.1.0.16, automatic data set deletion can be disabled for a z/OS task by defining the OPSDSEDEL parameter with a value of NO in the z/OS task definition.

To configure your z/OS task with automatic data set deletion disabled:

Step 1	Open the task record.
Step 2	<p>Use the add parameters feature to add the following parameter:</p> <div data-bbox="261 1514 789 1600" style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <pre>OPSDSEDEL=NO</pre> </div>

The OPSDSEDEL parameter accepts a value of YES (the default) or NO. A value of YES specifies that automatic data set deletion is enabled for the z/OS task. A value of NO specifies that automatic data set deletion is disabled for the z/OS task. The [Ops Rerun Report](#) will indicate if the feature has been disabled.

Rerunning a z/OS Task

Basic Rules for a Restart

When you rerun a z/OS task, Opwise allows you to rerun jobsteps using the methods described below. The Restartable Steps tab provides a list of steps and highlights in green from which steps you can start the rerun.

When you rerun a z/OS task, Opwise automatically performs the following:

- Deletes data sets that were created in dependent steps.
- Maintains Generation data group.



Note

You must restart a z/OS task from the Opwise Automation Center user interface in order for these clean-up procedures to be performed. Do not restart the task from the z/OS prompt.

Confirming Any JCL Changes

If you make any JCL changes, Opwise will prompt you for a confirmation, as described in the procedures below.

To Restart a z/OS Task from a Specific Step

- Step 1** On the Activity screen, click the Instance Name of the task you want to restart. The z/OS Task Instance screen displays.
- Step 2** Click the **Restartable JobSteps** tab. This displays a list of each step of the z/OS job. If applicable, the Failed column indicates "true," highlighted in red, to show where the job failed.
- Step 3** Assuming the failure was caused by error(s) in the JCL, examine the JCL and make your corrections. Opwise uses a background process to determine whether changes have been made to the JCL. Any changes trigger the confirmation process.
- Step 4** When you have corrected the JCL, consult the **Restartable** column to determine which steps are available as restart points. A value of True means you can restart from this JCL step; a value of False means you cannot restart from this step. If you choose to rerun specific steps, you can only run steps flagged as True in the Restartable column.

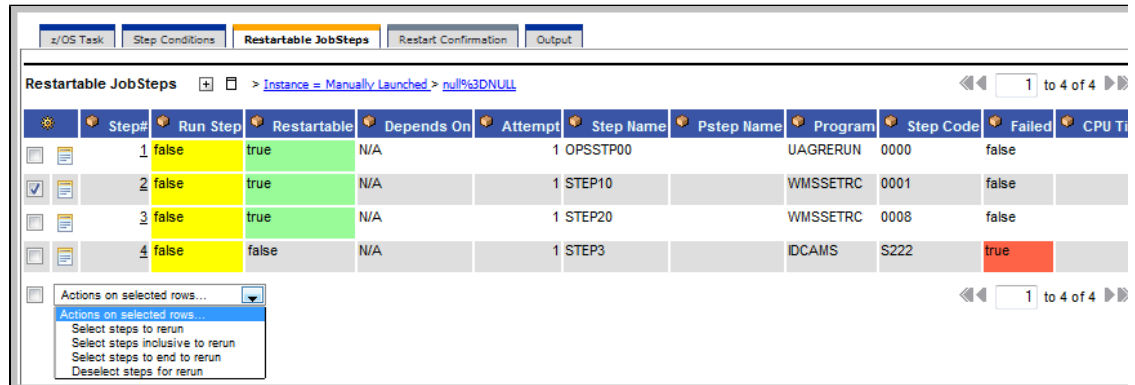
In the following example, the job failed at step 4 and the latest step you can restart from is step 3.

Step#	Run Step	Restartable	Depends On	Attempt	Step Name	Pstep Name	Program	Step Code	Failed	CPU Time
1	false	true	N/A	1	OPSSTP00		UAGRERUN	0000	false	0
2	false	true	N/A	1	STEP1		DCAMS	0000	false	1
3	false	true	N/A	1	STEP2		DCAMS	0016	false	1
4	false	false	N/A	1	STEP3		DCAMS	S222	true	0

Step 5 You can restart the job from a specific step or select specific steps to re-run.

- **To restart the task from a specific step and run it to the end:**

1. Click the box to the left of the step from which you want to restart the task, as shown in the following illustration. For example, to restart the job from step 2 to the end, select 2.



2. Click the down-arrow in **Actions on selected rows...** and select **Select steps to end to rerun**.
3. Return to the z/OS Task main screen and click the **Re-run** button.

- **To re-run one or more specific steps:**

1. Click the box to the left of the steps you want to re-run.
2. Click the down-arrow in **Actions on selected rows...** and select **Select steps to rerun**.
3. Return to the z/OS Task main screen and click the **Re-run** button.

- **To re-run a range of steps:**

1. Click the boxes to the left of the first and last steps you want to re-run. For example, to run steps 1 through 3, click 1 and 3.
2. Click the down-arrow in **Actions on selected rows...** and select **Select steps inclusive to rerun**.
3. Return to the z/OS Task main screen and click the **Re-run** button.

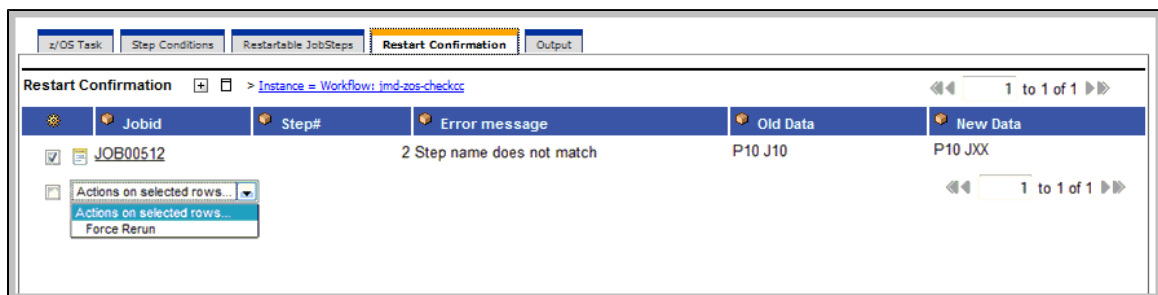
- **To start over before clicking on the Re-run button:**

1. Click the down-arrow in **Actions on selected rows...**
2. Select **Deselect steps for rerun**.

Step 6 Once you have made your restart selection, Opwise reruns the task.

Step 7 If you have made changes to the JCL, the task goes into a status of **Confirmation Required** on the Activity screen. Confirm the change follows:

1. From the Activity screen, click on the task name to open the record.
2. Click the **Restart Confirmation** tab. This displays a record for each change that was made to the JCL, with a description in the Error Message column, as shown in the following example.



3. To confirm the change(s), click the box to the left of each listed change and click the **Actions on selected rows**.
4. Select **Force Rerun**.
5. Return to the z/OS Task main screen and click the **Re-run** button.

Step 8 After the rerun is complete, the Failed column should show false for each step.

Step#	Run Step	Restartable	Depends On	Attempt	Step Name	Pstep Name	Program	Step Code	Failed	CPU Time
1	false	true	N/A	1	OPSTP00		UAGRERUN	0000	false	1
2	true	true	N/A	1	STEP10		WMSSETRC	0001	false	0
3	true	true	N/A	1	STEP20		WMSSETRC	0008	false	0
4	true	true	N/A	1	STEP30		WMSSETRC	FLSH	false	



Note

If you repeat the above process on the same task instance, the previous list of JCL changes, if any, is wiped clean and replaced with the most recent list of changes.

z/OS Restartable JobSteps Tab Column Descriptions

The following table describes each column on the z/OS Restartable Job Steps screen.

Column Name	Description
Step #	Number assigned to this step by Opwise.
Run Step	Indicates whether this step ran during the last run or restart of this job. For example, if you just re-started the job from step 4, steps 1, 2, and 3 would indicate false, and steps 4 to the end would indicate true.
Restartable	When the JCL job fails, Opwise determines the latest step you can restart from. A step indicating True and highlighted in green means you can restart from this step.
Depends On	Indicates which step or steps must be completed successfully before you can run this step.
Attempt	Number of times this step has been run or attempted to run.
Step Name	Extracted from the JCL. The name of the JCL step.
Pstep Name	Process step name from within the JCL step.
Program	Name of the program being executed by the step.
Step Code	Extracted from the JCL. Exit code for this step of the program.
Failed	True or false. Indicates whether or not this step failed. True means the step failed.
CPU Time	Number of CPU seconds it took for the JCL step to run.
IO Total	Total input/output operations for this step.
Memory Peak	Peak amount of memory used during the execution of this step.

z/OS Restart Confirmation Tab Column Descriptions

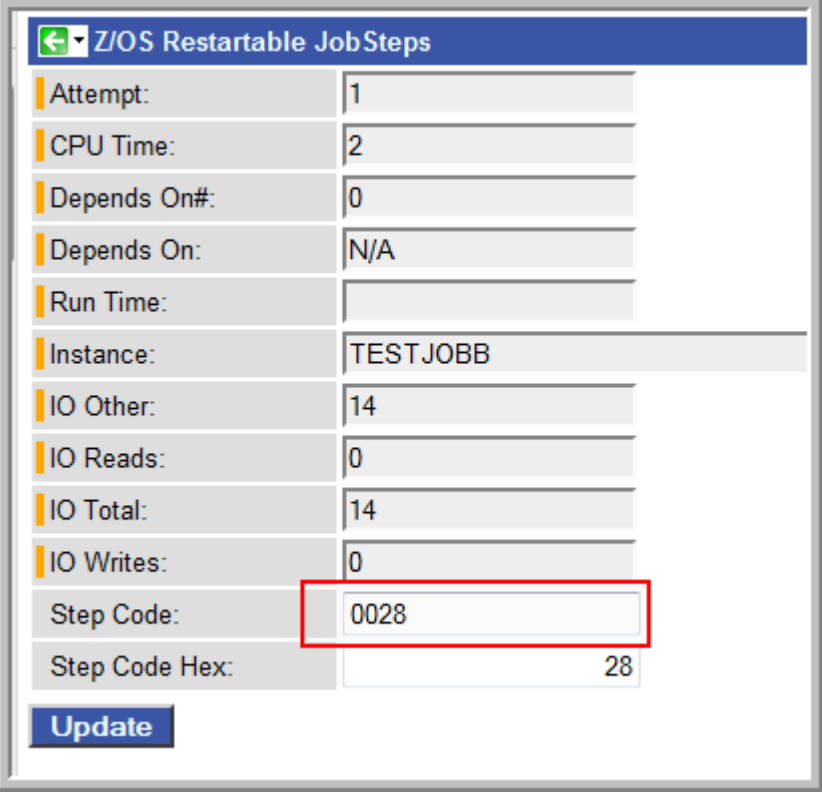
The following table describes each column on the Z/OS Restartable Job Steps screen.

Column Name	Description
Jobid	Number assigned to this step by Opwise.
Step#	JCL step number that was modified.

Error message	Description of the change.
Old Data	JCL before the change.
New Data	JCL after the change.

Interactively Ignoring a Stepcode to Force a Task to Complete

If the exit code on a previous step is causing a step failure and you have determined that you want to finish the job run anyway, you can change the exit code to force finish the task.

Step 1	Open the task instance from the Activity screen.
Step 2	<p>From the Restartable Steps tab, click on the step whose exit code you want to change. Opwise opens the record for this step, as shown in the following example:</p> 
Step 3	Enter the new code in the Step Code field and click Update .
Step 4	Return to the main task screen and click Rerun .

CA7/CA11 Toleration

Non-Restartable Steps

Opwise can read and interpret JCL step names that have been customized for CA11 and UCC. This allows you to launch your existing CA11 and UCC jobs from Opwise without modifying the JCL. When Opwise encounters one of the step names listed below in your JCL, Opwise will skip the step during a restart:

- CA11NR - CA11 Non-Restartable
- UCC11NR - UCC11 Non-Restartable
- OPSNR000 - Opwise Non-Restartable

In the example shown below, steps 4 to 12 each have one of the above DD Names and are therefore non-restartable steps.

z/OS Task	Step Conditions	Restartable JobSteps	Restart Confirmation	Output						
Restartable JobSteps + Instance = Manually Launched > null%3DNULL 1 to 13 of 13										
Step#	Run Step	Restartable	Depends On	Attempt	Step Name	Pstep Name	Program	Step Code	Failed	CPU Time
1	false	true	N/A	1	OPSSTP00		UAGRERUN	0000	false	1
2	false	true	N/A	1	DEL		IEFBR14	0000	false	0
3	false	true	N/A	1	ALLOC		IEFBR14	0000	false	0
4	false	false	Step# 2	1	EQ0000C1	PS1	UDM	FLSH	false	
5	false	false	Step# 2	1	EQ0000C2		ISRSUPC	FLSH	false	
6	false	false	Step# 2	1	EQ0000C3	PS1	UDM	FLSH	false	
7	false	false	Step# 2	1	EQ0000C4		ISRSUPC	FLSH	false	
8	false	false	Step# 2	1	EQ0000C5	PS1	UDM	FLSH	false	
9	false	false	Step# 2	1	EQ0000C6		ISRSUPC	FLSH	false	
10	false	false	Step# 2	1	EQ0000C7	PS1	UDM	FLSH	false	
11	false	false	Step# 2	1	EQ0000C8		ISRSUPC	FLSH	false	
12	false	false	Step# 2	1	EQ0000C9	PS1	UDM	FLSH	false	
13	false	true	N/A	1	CLEANUP		IEFBR14	FLSH	false	

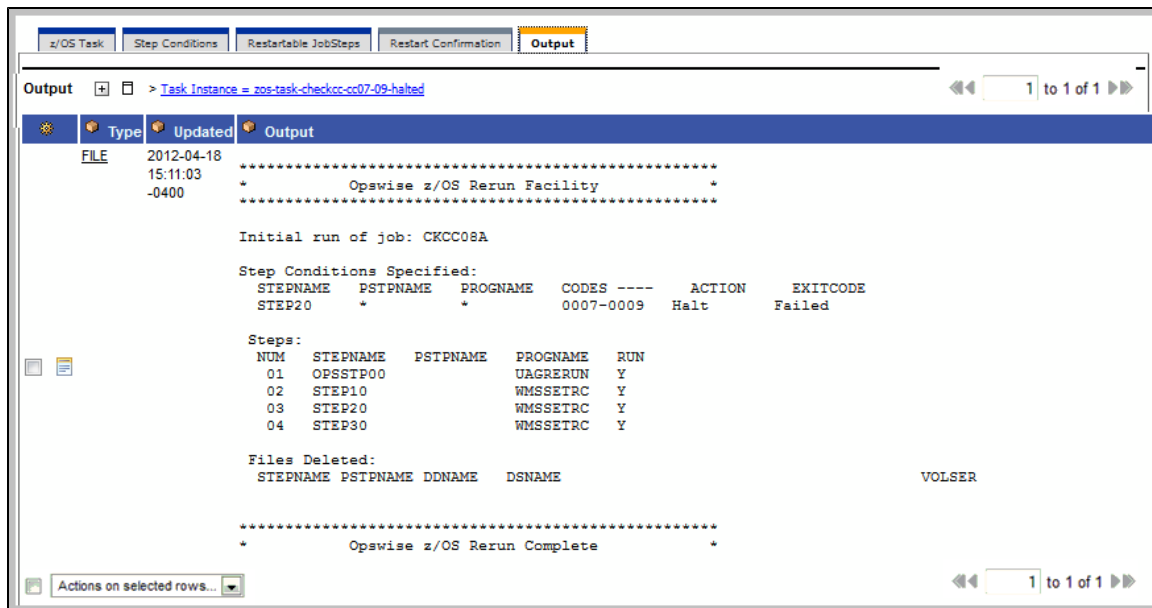
SCHID - Overriding the CA7 Schedule ID

The Schedule ID field allows you to override the CA7 SCHID, or Schedule ID. For example, the JCL shown below contains CA7 Scheduled Overrides statements #JI and #JEND. This JCL will set CLASS=A if the SCHID is between 1 thru 39, and set CLASS=B if the SCHID is between 40 thru 79. The user can set the SCHID by entering it into the Schedule ID field on the z/OS Task definition screen. The Opwise agent scans for #JI and #JEND, and generates the appropriate JCL, as shown in the example below.

```
//SCHID JOB (IMS,001),JIM,MSGCLASS=X,MSGLEVEL=(1,1),NOTIFY=&SYSUID,
#JI, ID=1-39
//          CLASS=A
#JEND
#JI, ID=40-79
//          CLASS=B
#JEND
//S1      EXEC PGM=IGWSPZAP
//SYSLIB DD DSN=OPS01.JS01.LOAD,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN   DD *
DUMPT WMSSETRC WMSSETRC
/*
//
```

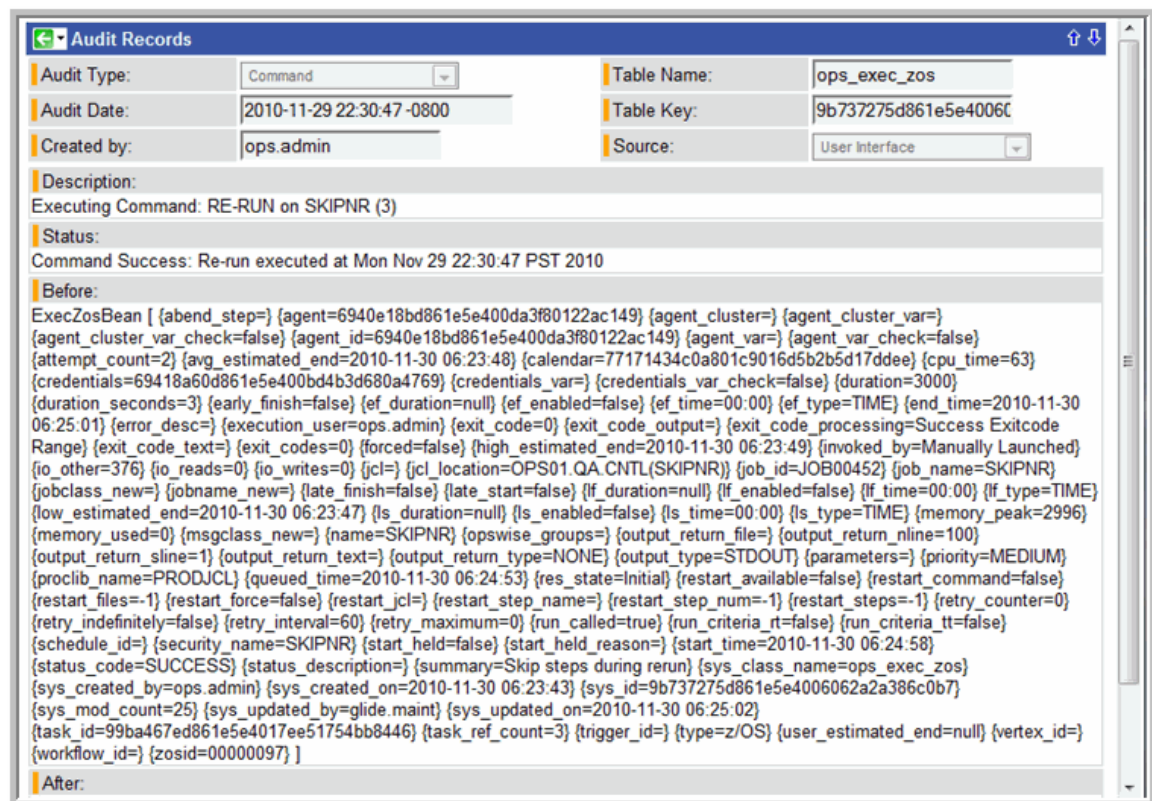
Viewing Ops Rerun Reports

Opwise keeps a detailed record of task restarts. This data is written to the **Output** tab on the task instance record, as shown in the sample below:



Viewing Audit Trails on a Restart

Opwise maintains detailed audit records on all system activity. The sample below shows an audit record for a re-run on a z/OS task called SKIPNR.



Indesca Task

- [Before You Begin](#)
- [Creating a New Indesca Task](#)
- [Output Redirection](#)
- [Indesca Task Field Descriptions](#)
- [Specifying When a Task Runs](#)
- [Monitoring Task Execution](#)

Before You Begin

The Indesca task allows you to run a platform-specific application on a machine where Indesca is running. From the Opwise perspective, Indesca serves as an agent process. Indesca runs on any supported platform: z/OS, Linux/Unix, and Windows.

To run an Indesca task, you must first complete the following tasks:

- Install Indesca on the target machine.
- Launch the Indesca agent. When the agent connects with the Controller, it automatically creates an [agent resource definition](#) in the Opwise database.
- Optionally, customize the agent heartbeat and credentials, as described in [Indesca Agent Definition Field Descriptions](#).

Creating a New Indesca Task

Step 1 From the navigation pane, select **Automation Center > Tasks > Indesca Tasks**. The Indesca Tasks List screen displays.

Step 2 Click **New**. The Indesca Task Definition screen displays.

Step 3 Using the field descriptions provided below as a guide, complete the fields as needed.

Step 4 Click the **Submit** button to save the record and return to the menu, or right-click on the title bar and select **Save** to save the record and remain on the current display.

Step 5 If appropriate, repeat these steps for any additional tasks you want to add.

Task run statistics will appear after the first time this task has been launched.

Output Redirection

Universal Automation Center Agent (UAG) processes Indesca, File Transfer/Infitran, and SAP task types differently than Windows and Linux/Unix task types. Indesca, File Transfer/Infitran, and SAP command lines are sent to the user process via standard input, so any redirection operators entered as task command input are not processed as expected.

If you want to direct output from an Indesca task to your file system, the **-uagstdio** command option lets you specify the same output redirection commands that are available for Windows and Linux/Unix task types. UAG will apply the user-specified value for **-uagstdio** directly to the command image.

The I/O redirection commands that you can use with **-uagstdio** are dependent on the OS/command shell. You should be able to set up any redirection that the OS/command shell supports (just as with Windows and Unix/Linux task types).

The syntax of **-uagstdio** is similar to Universal Command, Universal Data Mover, and Universal Connector [command line options](#); option followed by value.

For the Indesca task type, you can specify **uagstdio** in either of the following fields:

- [Command](#)
- [Indesca Options](#)

-uagstdio Examples

```
-uagstdio >C:\INDESCAOUT\indesca.out
```

If the **-uagstdio** value contains spaces, it must be enclosed in double quotation marks (" "):

```
-uagstdio ">C:\INDESCAOUT\indesca.out 2>C:\INDESCAOUT\indesca.err"
```

If the quoted value itself requires double quotation marks, they must be doubled (""):

```
-uagstdio ">C:\tmp\"\"indesca output\""\indesca.out 2>C:\tmp\"\"indesca output\""\indesca.err"
```

Indesca Task Field Descriptions

The table below describes the fields, buttons, and tabs on the task definition and task instance screens. Color coding is provided that differentiates the following three types of fields:

- Fields that display on the task definition and task instance screens are shown in black.
- Fields that display only on the task definition screen are shown in green.
- Fields that display only on the task instance screen are shown in maroon.

Field Name	Description
Task/Instance Name	Required. Name used within Opwise to identify this task. Up to 40 alphanumeric; variables supported. It is the user's responsibility to develop a workable naming scheme for tasks.
Invoked by	Task instance only; system-supplied. How the task instance was launched. One of the following: <ul style="list-style-type: none"> • Trigger: (Trigger Name) - The instance was launched by the named trigger. • Workflow: (Workflow Name) - The instance was launched by the named workflow. • Manually Launched - The instance was launched by a user. To determine the name of the user: <ol style="list-style-type: none"> 1. From the Activity or Task Instances screen, click the task instance name to open the record. 2. The Execution User field identifies the user who launched the task instance.
Task	Task instance only; system-supplied. Hover over the paper icon to display more information about the task instance. Click the paper icon to display the task definition record.
Execution User	Task instance only; system-supplied. If the task was launched manually, the ID of the user who launched it.
Instance Reference Id	Task instance only; system-supplied. Opwise increments this number each time the task is run.

Utility Credentials	Login credentials the Opwise agent will use to access the Indesca server machine.
Version	Task definition only; system-supplied. The version number of the current record, which is incremented by Opwise every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Credentials Variable	Optional. If enabled, the Utility Credentials field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: <code>\${variable name}</code> . The variable must be a supported type as described in Variables and Functions .
Utility Agent	Required. The name of the Windows, Linux/Unix, or z/OS agent resource that will communicate with the Indesca agent. If you do not specify an agent, you must specify an agent cluster (see below).
Utility Agent Variable	Optional. If enabled, the Utility Agent field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: <code>\$(variable name)</code> . The variable must be a supported type as described in Variables and Functions .
Utility Agent Cluster	Optional. You can specify an agent cluster in addition to or in place of a specific agent. An agent cluster is a group of agents, one of which Opwise will choose to run this task. If you specify an agent and an agent cluster, Opwise Automation Center first tries to run the task on the specific agent. If the agent is not available, Opwise reverts to the agent cluster. See Agent Clusters for more information.
Cluster Variable	Optional. If enabled, the Agent Cluster field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: <code>\$(variable name)</code> . The variable must be a supported type as described in Variables and Functions .
Indesca Agent	Required. Depending on the value in the Indesca Agent Option field (below), this field might contain a record name from the Indesca Agent table, a variable that will be resolved when the task is launched, or the hostname of a machine where the Indesca agent is running.
Indesca Credentials	Required. The login credentials that Opwise Automation Center will use to access the remote machine where the Indesca Agent is running. See Credentials .
Indesca Agent Option	Specifies how the name of the Indesca Agent is being supplied in the Indesca Agent field. Options: <ul style="list-style-type: none"> • Indesca Agent - The Indesca agent record is selected from the Indesca Agent table. • Indesca Agent Variable - The Indesca Agent field contains a variable that will be resolved when the task is launched. • Indesca Agent Hostname - The Indesca Agent field contains the hostname where the Indesca agent is running. The hostname must be accessible to Opwise.
Indesca Credentials Variable	Optional. If enabled, the Indesca (Agent) Credentials field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: <code>\$(variablename)</code> . The variable must be a supported type as described in Variables and Functions .
Hold on Start	If enabled, when the task is launched it appears in the Activity display with a status of Held. The task runs when the user releases it.
Hold Reason	Information about why the task will be put on hold when it starts.

Task Description	User-supplied description of this record.
Member of Business Services	User-defined. Allows you to select one or more Business Services that this record definition belongs to. Click the lock icon to unlock the field and select Business Services .
Status	Task instance only; system-supplied. See Task Instance Statuses .
Exit Code	Task instance only; system-supplied. The exit code captured by the agent when executing the task (for example, a command or script).
Status Description	Task instance only; system-supplied. Provides additional information, if any, about the status of the task.
Queued Time	Task instance only; system-supplied. The time the task was queued for processing.
Process ID	Task instance only; system-supplied. The ID of the process that was launched.
Start Time	Task instance only; system-supplied. The date and time the task started.
CPU Time	Task instance only; system-supplied. The amount of CPU time the task took to run.
End Time	Task instance only; system-supplied. The date and time the task instance completed.
Duration	Task instance only; system-supplied. The amount of time the task took to run.
User Estimated Duration	Task definition only; optional. The estimated amount of time it should normally take to run this task. Opwise uses this information to calculate the User Estimated End Time on a task instance record.
Command or Script	Specifies whether a single command or a script is being executed. Options: <ul style="list-style-type: none"> • Command (default) • Script
Command	Required (if Command is selected in Command or Script field). Command being executed on the remote machine. Variables supported.
Script Options	Optional. One or more command line options to pass to the script file.
Script File	Path and filename of the script file that will be executed on the remote machine.
Indesca Options	Optional. Any Indesca options needed by the program to execute properly. Variables supported.

Runtime Directory	Optional. The directory from which the application should be executed. Variables supported.
Exit Code Processing	Required. Specifies how Opwise Automation Center should determine whether the executed command failed or completed successfully. Options: <ul style="list-style-type: none"> • Success Exitcode Range - The command is considered completed successfully if its exit code falls within the range specified in the Exit Codes field (see below). • Failure Exitcode Range - The command is considered failed if its exit code falls within the range specified in the Exit Codes field (see below). • Success Output Contains - The command is considered completed successfully if its output contains the text specified in the Scan Output For field (see below). • Failure Output Contains - The command is considered failed if its output contains the text specified in the Scan Output For field (see below). • Step Conditions - The command is considered completed successfully/failed if any of its specified condition codes falls within the range specified under the Step Conditions tab (see Creating Step Conditions).
Output Type	Required if Exit Code Processing = Success Output Contains or Failure Output Contains. This field specifies the type of output. Options: <ul style="list-style-type: none"> • Standard Output (STDOUT) • Standard Error (STDERR) • File
Exit Codes	Required if Exit Code Processing = Success Exitcode Range or Failure Exit Code Range. This field specifies the range. Format: Numeric. Use commas to list a series of exitcodes; use hyphens to specify a range. Example: 1,5, 22-30.
Scan Output For	Required if Exit Code Processing = Success Output Contains or Failure Output Contains. This field specifies the text for which Opwise should scan the output file. Opwise will process this field as a regular expression.
Output File (Exit Code processing)	Required if Output Type = File. This field specifies the path and file name of the output file that should be scanned for the text in the Scan Output For field.
Automatic Output Retrieval	Optional. Allows you to specify whether you want Opwise to automatically retrieve any output from the job and attach it to the task instance record. Options: <ul style="list-style-type: none"> • None - Do not attach any output to the task instance record. • Standard Output - Attach all standard output. • Standard Error - Attach standard error output. • File - Attach the file specified in the Output File field.
Output File (Automatic Output Retrieval)	Required if Automatic Output Retrieval=File. This field specifies the path and filename containing the output that you want automatically retrieved and attached to the task instance.
Start Line	Optional. Allows you to instruct Opwise to retrieve data beginning at the line indicated. If a Start Line value is not specified on the screen, the default is 1.
Number of Lines	Optional. Allows you to limit the retrieved data to the number of lines specified. If a Number of Lines value is not specified, the default is the value of the Retrieve Output Default Maximum Lines Opwise system property.
Scan Text	Optional. Instructs Opwise to scan the data for the text specified and retrieve only that. Opwise will process this field as a regular expression.

Late Start	If enabled, and if the task instance starts after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late start (see Late Start Type). To determine whether a task instance started late, open the task instance and locate the Started Late field; the field is checked if the instance started after the specified time. This field only appears on the task instance if the user specified a Late Start in the task definition.
Late Start Type	Required if Late Start is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it starts after the specified time. • Duration - Flag the task if it starts a certain amount of time after the programmed start time. The task must have a specific start time.
Late Start Time	Time after which the task start time is considered late. Use hh:mm, 24-hour time
Late Start Duration	Duration (amount of relative time) after which the task is considered to have started late. For a task within a workflow, the duration is the period between the time the workflow starts and the time the task itself starts. For example, a task might have a Late Start Duration of 60 minutes. If the workflow starts at 9:00 a.m. but the task itself does not start until 10:30, the task has started late. For a task that is not within a workflow, Late Start Duration has meaning only if the task has been held upon starting. For example, if a task has a Late Start Duration of 60 minutes and the Hold on Start field is enabled, if the task is not released from hold within the amount of time specified in the Late Start Duration field, the task has started late.
Started Late	Task instance only; system-supplied. This field is flagged if the task started later than the time specified in the Late Start fields.
Late Finish	If enabled, and if the task instance finishes after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late finish (see Late Finish Type). To determine whether a task instance finished late, open the task instance and locate the Finished Late field; the field is checked if the instance finished after the specified time or lasted longer than expected. This field only appears on the task instance if the user specified a Late Finish in the task definition.
Late Finish Type	Required if Late Finish is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it finishes after the specified time (see Late Finish Time). • Duration - Flag the task if it finishes a certain amount of time after the programmed finish time (see Late Finish Duration). The task must have a specific finish time.
Late Finish Time	If Late Finish Type is Time, use this to specify the time after which the task finish time is considered late. Use hh:mm, 24-hour time.
Late Finish Duration	If Late Finish Type is Duration, use this to specify the longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Finished Late	Task instance only; system-supplied. This field is flagged if the task finished later than the time or duration specified in the Late Finish fields.
Early Finish	If enabled, and if the task instance finishes before the time or period specified, the task instance is flagged as early. You can specify a time or duration to determine an early finish (see Early Finish Type). To determine whether a task instance finished early, open the task instance and locate the Finished Early field; the field is checked if the instance finished before the specified time or did not last as long as expected. This field only appears on the task instance if the user added Early Finish specifications to the task definition.

Early Finish Type	Required if Early Finish is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it finishes before the specified time (see Early Finish Time). • Duration - Flag the task if it finishes a certain amount of time before the programmed finish time (see Early Finish Duration). The task must have a specific finish time.
Early Finish Time	If Early Finish Type is Time, use this to specify the time before which the task finish time is considered early. That is, enter a time at which the task should still be running. Use hh:mm, 24-hour time.
Early Finish Duration	If Early Finish Type is Duration, use this to specify the shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Finished Early	Task instance only; system-supplied. This field is flagged if the task finished earlier than the time specified in the Early Finish fields.
Task Priority	Task instance only; the priority of this task instance, as set by the user via the Set Priority command. Options are: HIGH, MEDIUM, LOW.
User Estimated End Time	Task instance only; system-supplied. If the user entered information into the User Estimated Duration field in the task definition, Opwise uses this information to calculate an end time for the task instance, based on the date/time the task instance started.
Shortest Estimated End Time	Task instance only; system-supplied.
Average Estimated End Time	Task instance only; system-supplied.
Longest Estimated End Time	Task instance only; system-supplied.
Maximum Retries	User-defined. The maximum number of times Opwise should retry this task after it has started and gone to a failed state.
Retry Indefinitely	User-defined. Enabled or disabled. Indicates whether Opwise should continue trying indefinitely to run this task. If you enable this field, it overrides any value placed in the Maximum Retries field (above).
Retry Interval	User-defined. The number of seconds between each retry.
Current Retry Count	Task instance only; system-supplied. Displays, only for a running task instance, the current number of times that Opwise has retried the task after it first went to failure status.
Virtual Resource Priority	Priority for acquiring a resource when two or more tasks are waiting for the resource. This priority applies to all resources required by the task. Options: 1 (high) - 20 (low). Default is 10.
Hold Resources on Failure	If enabled, the task instance will continue to hold Renewable resources if the task instance fails. Renewable resources will be returned only if the task instance status is either Complete, Finished, or Skipped.

Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Launch Task button	Manually launches the task.
View Instances button	Displays a list of task instances for which there has been a status change or a modification to the task instance record within the last 30 days (an Updated on Last 30 Days filter has been pre-selected for the list).
Delete button	Deletes the current record.
View Parent button	Task instance only; Displays this task's parent task (workflow), if any.
Show Details button	Task instance only; displays detailed information about this task instance.
Retrieve Output button	Task instance only; see Retrieving Output .
Skip button	Task instance only; for tasks loaded into the schedule that haven't run yet. Allows you to tell Opwise to skip this task. See Skipping a Task .
Hold button	Task instance only; see Putting a Task on Hold .
Force Finish button	Task instance only; see Force Finishing a Task .
Re-run button	Task instance only; see Re-running a Task .
Variables tab	Displays all variables associated with this record.
Actions tab	<p>Allows you to specify actions that Opwise will take automatically based on events that occur during the execution of this task. Events are task instance status, exit codes, late start, late finish, and early finish. Actions are:</p> <ul style="list-style-type: none"> • Abort Action - Abort the task if certain events occur. For details, see Setting Abort Actions. • Email Notification - Send an email if certain events occur. For details, see Creating Email Notifications. • Set Variable - Used in workflows to set a variable based on the occurrence of certain events. For details, see Setting Variables within a Workflow. • SNMP Notification - Send an email if certain events occur. For details, see Creating SNMP Notifications. • System Operation - Run an Opwise system operation based on specified conditions. For details, see Setting System Operations.
Task Virtual Resources tab	Lists Virtual Resources to which this task is assigned.
Mutually Exclusive Tasks tab	Displays all tasks that have been set to be mutually exclusive of this task.

Triggers tab	Displays a list of all triggers that have been defined to launch this task. Also allows you to add new triggers. If you add a new trigger from this location, Opswise automatically constructs a default trigger name as follows: <current task name>#TRIGGER#. You can change the default name if desired. For instructions on creating triggers, see Creating Triggers .
Notes tab	Displays all notes associated with this task.
Versions tab	Stores copies of all previous versions of the current record. See Record Versioning .
Output tab	Task instance only. Displays output generated from the process, if any, based on specifications provided by the user in the Automatic Output Retrieval fields in the task definition.

Specifying When a Task Runs

You can run the task as part of a [workflow](#), [specify triggers](#) that run the task automatically based on times or events, or [run the task manually](#).

Monitoring Task Execution

You can monitor all system activity from the [Activity screen](#) and can view activity history from the [Activity History screen](#).

SAP Task

- Overview
- Before You Begin
- Creating a New SAP Task
- Output Redirection
- SAP Task Field Descriptions
- Universal Connector Commands
- Built-In Variables
- Specifying When a Task Runs
- Monitoring Task Execution

Overview

**Note**

These instructions assume the user has a working knowledge of SAP.

The SAP task allows you to send commands to an SAP system and gather status information and output back from SAP. The SAP task uses Stonebranch's proprietary Universal Connector *for Use with SAP® ERP* (USAP) to communicate with SAP. Universal Connector allows Opswise to connect to an SAP system and manage SAP background processing tasks.

Before You Begin

To run an SAP task, you must first complete the following:

- Identify an Opswise Linux/Unix agent that will interface with the SAP system.
- Define an SAP connection in the Opswise database.

Creating a New SAP Task

Step 1 From the navigation pane, select **Automation Center > Tasks > SAP Tasks**. The SAP Tasks List screen displays.

Step 2 Click **New**. The SAP Task Definition screen displays.

The screenshot displays the 'SAP Task' configuration window. The title bar includes tabs for 'Variables', 'Actions', 'Task Virtual Resources', 'Mutually Exclusive Tasks', 'Triggers', 'Notes', and 'Versions'. The main area contains the following fields and controls:

- Task Name:** Required field, empty text box.
- Version:** Text box containing '1'.
- Utility Agent:** Text box with a search icon.
- Utility Agent Variable:** Check box, unchecked.
- SAP Connection:** Text box with a search icon.
- SapLang:** Text box.
- Hold on Start:** Check box, unchecked.
- Task Description:** Large text area.
- Member of Business Services:** Lock icon.
- User Estimated Duration:** Time picker showing '00:00:00' in hh:mm:ss format.
- Command Group:** Dropdown menu set to 'Run'.
- Definition or Model:** Dropdown menu set to 'USAP Definition File'.
- Script Library or File System:** Dropdown menu set to 'Script Library'.
- Script:** Large text area.
- Start Immediately:** Check box, unchecked.
- SAP Target Server:** Text box.
- Print Application Log:** Check box, checked.
- Print Application RC:** Check box, checked.
- Use Application RC:** Check box, unchecked.
- SAP Command Options:** Expandable text area.
- Runtime Directory:** Text box.
- Exit Code Processing:** Dropdown menu set to 'Success Exitcode Range'.
- Exit Codes:** Text box containing '0'.
- Automatic Output Retrieval:** Dropdown menu set to 'Standard Output/Error'.
- Start Line:** Text box containing '1'.
- Number of Lines:** Text box containing '100'.
- Scan Text:** Text box.
- Late Start:** Check box, unchecked.
- Late Finish:** Check box, unchecked.
- Early Finish:** Check box, unchecked.
- Maximum Retries:** Text box containing '0'.
- Retry Interval (Seconds):** Text box containing '60'.
- Virtual Resource Priority:** Dropdown menu set to '10'.
- Retry Indefinitely:** Check box, unchecked.
- Hold Resources on Failure:** Check box, unchecked.

A **Submit** button is located at the bottom left of the form.

Step 3 Using the field descriptions provided below as a guide, complete the fields as needed.

Step 4 Click the **Submit** button to save the record and return to the menu, or right-click on the title bar and select **Save** to save the record and remain on the current display.

Step 5 If appropriate, repeat these steps for any additional tasks you want to add.

Task run statistics appear after the first time this task has been launched.

Output Redirection

Universal Automation Center Agent (UAG) processes SAP, Indesca, and File Transfer/Inftran task types differently than Windows and Linux/Unix task types. SAP, Indesca, and File Transfer/Inftran command lines are sent to the user process via standard input, so any redirection operators entered as task command input are not processed as expected.

If you want to direct output from an SAP task to your file system, the **-uagstdio** command option lets you specify the same output redirection commands that are available for Windows and Linux/Unix task types. UAG will apply the user-specified value for **-uagstdio** directly to the command image.

The I/O redirection commands that you can use with **-uagstdio** are dependent on the OS/command shell. You should be able to set up any redirection that the OS/command shell supports (just as with Windows and Unix/Linux task types).

The syntax of **-uagstdio** is similar to Universal Data Mover, Universal Command, and Universal Connector [command line options](#); option followed by value.

For the SAP task type, you can specify **uagstdio** in the following field:

- [SAP Command Options](#)

-uagstdio Examples

```
-uagstdio >C:\SAPOUT\sap.out
```

If the **-uagstdio** value contains spaces, it must be enclosed in double quotation marks ("):

```
-uagstdio ">C:\SAPOUT\sap.out 2>C:\SAPOUT\sap.err"
```

If the quoted value itself requires double quotation marks, they must be doubled (""):

```
-uagstdio ">C:\tmp\"\"sap output\""\sap.out 2>C:\tmp\"\"sap output\""\sap.err"
```

SAP Task Field Descriptions

The table below describes the fields, buttons, and tabs on the task definition and task instance screens. Color coding is provided that differentiates the following three types of fields:

- Fields that display on the task definition and task instance screens are shown in black.
- Fields that display only on the task definition screen are shown in green.
- Fields that display only on the task instance screen are shown in maroon.

Field Name	Description
Task/Instance Name	Required. Name used within Opwise to identify this task. Up to 40 alphanumeric; variables supported. It is the user's responsibility to develop a workable naming scheme for tasks.
Invoked by	Task instance only; system-supplied. How the task instance was launched. One of the following: <ul style="list-style-type: none"> • Trigger: (Trigger Name) - The instance was launched by the named trigger. • Workflow: (Workflow Name) - The instance was launched by the named workflow. • Manually Launched - The instance was launched by a user. To determine the name of the user: <ol style="list-style-type: none"> 1. From the Activity or Task Instances screen, click the task instance name to open the record. 2. The Execution User field identifies the user who launched the task instance.
Task	Task instance only; system-supplied. Hover over the paper icon to display more information about the task instance. Click the paper icon to display the task definition record.
Execution User	Task instance only; system-supplied. If the task was launched manually, the ID of the user who launched it.

Instance Reference Id	Task instance only; system-supplied. Opwise increments this number each time the task is run.
Credentials	Credentials under which an Agent runs this task. These credentials override any credentials provided in the Agent resource definition for any Agent running this task.
Version	Task definition only; system-supplied. The version number of the current record, which is incremented by Opwise every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Credentials Variable	Optional. If enabled, the Credentials field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: \${variable name}. The variable must be a supported type as described in Variables and Functions .
Utility Agent	Required. Name of the Linux/Unix or Windows agent that will communicate with the SAP system. If you do not specify an agent, you must specify an agent cluster (see below).
Utility Agent Variable	Optional. If enabled, the Utility Agent field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: \${variable name}. The variable must be a supported type as described in Variables and Functions .
Utility Agent Cluster	Optional. You can specify an agent cluster in addition to or in place of a specific agent. An agent cluster is a group of agents, one of which Opwise will choose to run this task. If you specify an agent and an agent cluster, Opwise Automation Center first tries to run the task on the specific agent. If the agent is not available, Opwise reverts to the agent cluster. See Agent Clusters for more information.
Cluster Variable	Optional. If enabled, the Agent Cluster field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: \${variable name}. The variable must be a supported type as described in Variables and Functions .
SAP Connection	Required. The name of the SAP connection defined using the SAP connection screen. The SAP connection specifies information about the SAP server. Type in a name, or click the magnifying glass to browse to an existing SAP server definition or create a new one.
SAP Credentials	Login credentials Opwise will use to access the SAP system. The credentials are stored in the Opwise credentials table; see Credentials .
SAPLang	SAP logon language used when executing the SAP task. Valid values are: <ul style="list-style-type: none"> Any valid 1-character SAP language identifier Any valid 2-character ISO language identifier (no value). SAP will use the default language set up for the user. If there is no such default, the default is EN (English).
SAP Credentials Variable	Optional. If enabled, the Credentials field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: \${variable name}. The variable must be a supported type as described in Variables and Functions .
Hold on Start	If enabled, when the task is launched it appears in the Activity display with a status of Held. The task runs when the user releases it.

Hold Reason	Information about why the task will be put on hold when it starts.
Task Description	User-supplied description of this record.
Status	Task instance only; system-supplied. See Task Instance Statuses .
Exit Code	Task instance only; system-supplied. The exit code captured by the agent when executing the task (for example, a command or script).
Status Description	Task instance only; system-supplied. Provides additional information, if any, about the status of the task.
Member of Business Services	User-defined. Allows you to select one or more Business Services that this record definition belongs to. Click the lock icon to unlock the field and select Business Services .
User Estimated Duration	Task definition only; optional. The estimated amount of time it should normally take to run this task. Opwise uses this information to calculate the User Estimated End Time on a task instance record.
Command Group	See Universal Connector Commands , below, for a description of all supported commands and their contingent fields (options).
SAP Command Options	Use this field to specify any additional command options supported by Universal Connector (USAP) .
Runtime Directory	Optional. The directory from which the application should be executed. Variables supported.
Exit Code Processing	Required. Specifies how Opwise Automation Center should determine whether the executed command failed or completed successfully. Options: <ul style="list-style-type: none"> • Success Exitcode Range - The command is considered completed successfully if its exit code falls within the range specified in the Exit Codes field (see below). • Failure Exitcode Range - The command is considered failed if its exit code falls within the range specified in the Exit Codes field (see below). • Success Output Contains - The command is considered completed successfully if its output contains the text specified in the Scan Output For field (see below). • Failure Output Contains - The command is considered failed if its output contains the text specified in the Scan Output For field (see below). • Step Conditions - The command is considered completed successfully/failed if any of its specified condition codes falls within the range specified under the Step Conditions tab (see Creating Step Conditions).
Output Type	Required if Exit Code Processing = Success Output Contains or Failure Output Contains. This field specifies the type of output. Options: <ul style="list-style-type: none"> • Standard Output (STDOUT) • Standard Error (STDERR) • File
Exit Codes	Required if Exit Code Processing = Success Exitcode Range or Failure Exit Code Range. This field specifies the range. Format: Numeric. Use commas to list a series of exitcodes; use hyphens to specify a range. Example: 1,5, 22-30.
Scan Output For	Required if Exit Code Processing = Success Output Contains or Failure Output Contains. This field specifies the text for which Opwise should scan the output file. Opwise will process this field as a regular expression.

Output File (Exit Code Processing)	Required if Output Type = File. This field specifies the path and file name of the output file that should be scanned for the text in the Scan Output For field.
Automatic Output Retrieval	Optional. Allows you to specify whether you want Opswise to automatically retrieve any output from the job and attach it to the task instance record. Options: <ul style="list-style-type: none"> • None - Do not attach any output to the task instance record. • Standard Output - Attach all standard output. • Standard Error - Attach standard error output. • File - Attach the file specified in the Output File field.
Output File (Automatic Output Retrieval)	Required if Automatic Output Retrieval=File. This field specifies the path and filename containing the output that you want automatically retrieved and attached to the task instance.
Start Line	Optional. Allows you to instruct Opswise to retrieve data beginning at the line indicated. If a Start Line value is not specified on the screen, the default is 1.
Number of Lines	Optional. Allows you to limit the retrieved data to the number of lines specified. If a Number of Lines value is not specified, the default is the value of the Retrieve Output Default Maximum Lines Opswise system property.
Scan Text	Optional. Instructs Opswise to scan the data for the text specified and retrieve only that. Opswise will process this field as a regular expression.
Late Start	If enabled, and if the task instance starts after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late start (see Late Start Type). To determine whether a task instance started late, open the task instance and locate the Started Late field; the field is checked if the instance started after the specified time. This field only appears on the task instance if the user specified a Late Start in the task definition.
Late Start Type	Required if Late Start is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it starts after the specified time. • Duration - Flag the task if it starts a certain amount of time after the programmed start time. The task must have a specific start time.
Late Start Time	Time after which the task start time is considered late. Use hh:mm, 24-hour time
Late Start Duration	Duration (amount of relative time) after which the task is considered to have started late. For a task within a workflow, the duration is the period between the time the workflow starts and the time the task itself starts. For example, a task might have a Late Start Duration of 60 minutes. If the workflow starts at 9:00 a.m. but the task itself does not start until 10:30, the task has started late. For a task that is not within a workflow, Late Start Duration has meaning only if the task has been held upon starting. For example, if a task has a Late Start Duration of 60 minutes and the Hold on Start field is enabled, if the task is not released from hold within the amount of time specified in the Late Start Duration field, the task has started late.
Started Late	Task instance only; system-supplied. This field is flagged if the task started later than the time specified in the Late Start fields.
Late Finish	If enabled, and if the task instance finishes after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late finish (see Late Finish Type). To determine whether a task instance finished late, open the task instance and locate the Finished Late field; the field is checked if the instance finished after the specified time or lasted longer than expected. This field only appears on the task instance if the user specified a Late Finish in the task definition.

Late Finish Type	Required if Late Finish is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it finishes after the specified time (see Late Finish Time). • Duration - Flag the task if it finishes a certain amount of time after the programmed finish time (see Late Finish Duration). The task must have a specific finish time.
Late Finish Time	If Late Finish Type is Time, use this to specify the time after which the task finish time is considered late. Use hh:mm, 24-hour time.
Late Finish Duration	If Late Finish Type is Duration, use this to specify the longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Finished Late	Task instance only; system-supplied. This field is flagged if the task finished later than the time or duration specified in the Late Finish fields.
Early Finish	If enabled, and if the task instance finishes before the time or period specified, the task instance is flagged as early. You can specify a time or duration to determine an early finish (see Early Finish Type). To determine whether a task instance finished early, open the task instance and locate the Finished Early field; the field is checked if the instance finished before the specified time or did not last as long as expected. This field only appears on the task instance if the user added Early Finish specifications to the task definition.
Early Finish Type	Required if Early Finish is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it finishes before the specified time (see Early Finish Time). • Duration - Flag the task if it finishes a certain amount of time before the programmed finish time (see Early Finish Duration). The task must have a specific finish time.
Early Finish Time	If Early Finish Type is Time, use this to specify the time before which the task finish time is considered early. That is, enter a time at which the task should still be running. Use hh:mm, 24-hour time.
Early Finish Duration	If Early Finish Type is Duration, use this to specify the shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Finished Early	Task instance only; system-supplied. This field is flagged if the task finished earlier than the time specified in the Early Finish fields.
Maximum Retries	User-defined. The maximum number of times Opwise should retry this task after it has started and gone to a failed state.
Retry Indefinitely	User-defined. Enabled or disabled. Indicates whether Opwise should continue trying indefinitely to run this task. If you enable this field, it overrides any value placed in the Maximum Retries field (above).
Retry Interval	User-defined. The number of seconds between each retry.
Current Retry Count	Task instance only; system-supplied. Displays, only for a running task instance, the current number of times that Opwise has retried the task after it first went to failure status.
First Time Ran	Task definition only; system-supplied. Displays after the first time the task runs. The date and time this task first ran.

Lowest Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. The shortest amount of time this task has taken to run.
Last Time Ran	Task definition only; system-supplied. Displays after the first time the task runs. The most recent date and time the task ran.
Average Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. Shows the average amount of time this task takes to run.
Number of Instances	Task definition only; system-supplied. Displays after the first time the task runs. Shows the number of times this task has run.
Highest Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. The longest amount of time this task has taken to run.
Last Instance Duration	Task definition only; system-supplied. Displays after the first time the task runs. The amount of time the task took to run the last time it ran.
Virtual Resource Priority	<p>Priority for acquiring a resource when two or more tasks are waiting for the resource. This priority applies to all resources required by the task.</p> <p>Options: 1 (high) - 20 (low).</p> <p>Default is 10.</p>
Hold Resources on Failure	If enabled, the task instance will continue to hold Renewable resources if the task instance fails. Renewable resources will be returned only if the task instance status is either Complete, Finished, or Skipped.
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Launch Task button	Manually launches the task.
View Instances button	Displays a list of task instances for which there has been a status change or a modification to the task instance record within the last 30 days (an Updated on Last 30 Days filter has been pre-selected for the list).
Delete button	Deletes the current record.
View Parent button	Task instance only; Displays this task's parent task (workflow), if any.
Show Details button	Task instance only; displays detailed information about this task instance.
Retrieve Output button	Task instance only; see Retrieving Output .

Skip button	Task instance only; for tasks loaded into the schedule that haven't run yet. Allows you to tell Opwise to skip this task. See Skipping a Task .
Hold button	Task instance only; see Putting a Task on Hold .
Force Finish button	Task instance only; see Force Finishing a Task .
Re-run button	Task instance only; see Re-running a Task .
Output tab	Task instance only. Displays output generated from the process, if any, based on specifications provided by the user in the Automatic Output Retrieval fields in the task definition.
Variables tab	Displays all variables associated with this record.
Actions tab	Allows you to specify actions that Opwise will take automatically based on events that occur during the execution of this task. Events are task instance status, exit codes, late start, late finish, and early finish. Actions are: <ul style="list-style-type: none"> • Abort Action - Abort the task if certain events occur. For details, see Setting Abort Actions. • Email Notification - Send an email if certain events occur. For details, see Creating Email Notifications. • Set Variable - Used in workflows to set a variable based on the occurrence of certain events. For details, see Setting Variables within a Workflow. • SNMP Notification - Send an email if certain events occur. For details, see Creating SNMP Notifications. • System Operation - Run an Opwise system operation based on specified conditions. For details, see Setting System Operations.
Task Virtual Resources tab	Lists Virtual Resources to which this task is assigned.
Mutually Exclusive Tasks tab	Displays all tasks that have been set to be mutually exclusive of this task.
Triggers tab	Displays a list of all triggers that have been defined to launch this task. Also allows you to add new triggers. If you add a new trigger from this location, Opwise automatically constructs a default trigger name as follows: <current task name>#TRIGGER#. You can change the default name if desired. For instructions on creating triggers, see Creating Triggers .
Notes tab	Displays all notes associated with this task.
Versions Tab	Stores copies of all previous versions of the current record. See Record Versioning .

Universal Connector Commands

The following table identifies supported Universal Connector commands, describes the actions that each command performs, and lists each command's related options, which display on the SAP Task definition screen when that command is selected.

Command Name	Description	Options
Run	Performs the following actions:	<ul style="list-style-type: none"> • Definition or Model Specifies how the new SAP job will be created, based either on a USAP Definition File or an SAP Model Job.

1. Defines a new SAP, job based on either a USAP Definition file or an SAP Model Job.
2. Starts the defined job.
3. Waits for the job to complete.
4. Prints the job's joblog to standard error and the spoolists to standard output.
5. Purges the job from the SAP system.

- **Script Library or File System**

Specifies whether the USAP definition file exists in the file system of the machine where the agent is running or in the [Script Library](#).

- **Script**

Required (if Script is selected in **Command or Script** field). Name of the script that has been uploaded into the [Script Library](#) and will be executed by this task.

- **Definition File**

If you selected USAP Definition File above, use this field to provide the path and file name of the file.

- **SAP Job Name**

Job name of the SAP job. [Variables](#) supported.

- **SAP Job ID**

Job ID of the SAP job. [Variables](#) supported.

- **Target Job Name**

If you selected SAP Model Job above, use this field to provide the name of the new SAP job being created. If you leave this field blank, Opwise uses the same name as the SAP Model Job.

- **Start Immediately**

Enabled or disabled. Enabling the Start Immediately flag will cause the job to fail if SAP resources are not available to start the job immediately (for example, a background work process). Otherwise, the job will wait for SAP resources to become available.

- **SAP Target Server**

Name of an SAP instance at which a background job should be run. The name has the following format:

```
[host name]_[SAP System name]_[SAP
System number]
```

Where host name is the name of the server computer on which the instance is running, as specified in the system profile parameter SAPLOCALHOST.

Example:

```
hs0123_C11_55
```

- **Print Application Log**

Enabled or disabled. Specifies whether or not the job's application log, if one was generated, is returned.

- **Print Application RC**

Enabled or disabled. Specifies whether or not the job's application return codes, if they were set, are returned.

		<ul style="list-style-type: none"> • Use Application RC Specifies whether or not the SAP job's application return codes will be used to determine the return code for the Opwise task.
Run Process Chain	<p>Performs the following actions:</p> <ol style="list-style-type: none"> 1. Starts a process chain. 2. Waits for the process chain to complete. 3. Returns the process chain log. 4. Returns process logs. 5. Returns process spool lists. 	<ul style="list-style-type: none"> • Chain ID ID of the process chain to run.
Run InfoPackage	<p>Performs the following actions:</p> <ol style="list-style-type: none"> 1. Starts an InfoPackage. 2. Wait for the InfoPackage request to complete. 3. Returns status messages for the completed Infopackage request. 	<ul style="list-style-type: none"> • InfoPackage Name of the InfoPackage to run. • InfoPackage Job Name Name of the SAP batch job that processes the InfoPackage request.
Submit	Defines a new SAP job.	<ul style="list-style-type: none"> • Definition or Model Specifies how the new SAP job will be created, based either on a USAP Definition File or an SAP Model Job. • Script Library or File System Specifies whether the USAP definition file exists in the file system of the machine where the agent is running or in the Script Library. • Script Required (if Script is selected in Command or Script field). Name of the script that has been uploaded into the Script Library and will be executed by this task. • Definition File If you selected USAP Definition File above, use this field to provide the path and file name of the file. • SAP Job Name Job name of the SAP job. Variables supported. • SAP Job ID Job ID of the SAP job. Variables supported. • Target Job Name If you selected SAP Model Job above, use this field to provide the name of the new SAP job being created. If you leave this field blank, Opwise uses the same name as the SAP Model Job. • Start Enabled or disabled. Specifies whether or not the newly-defined SAP job should be started.

- **Start Immediately**

Enabled or disabled. Enabling the Start Immediately flag will cause the job to fail if SAP resources are not available to start the job immediately (for example, a background work process). Otherwise, the job will wait for SAP resources to become available.

- **SAP Target Server**

Name of an SAP instance at which a background job should be run. The name has the following format:

```
[host name]_[SAP System name]_[SAP System number]
```

Where host name is the name of the server computer on which the instance is running, as specified in the system profile parameter SAPLOCALHOST.

Example:

```
hs0123_C11_55
```

- **Wait**

Specifies whether Opwise should wait for the SAP process chain to complete processing.

- **Print Job Log**

Enabled or disabled. Specifies whether or not the job's joblog is returned.

- **Print Spooled Output**

Enabled or disabled. Specifies whether or not the spoollists of all job steps are returned.

- **Print Application Log**

Enabled or disabled. Specifies whether or not the job's application log, if one was generated, is returned.

- **Print Application RC**

Enabled or disabled. Specifies whether or not the job's application return codes, if they were set, are returned.

- **Use Application RC**

Specifies whether or not the SAP job's application return codes will be used to determine the return code for the Opwise task.

- **SAP ABAP Program Name**

Name of an ABAP program in an SAP system to which the model variant belongs.

- **SAP Variant Name**

Pre-existing SAP variant name to use as the model variant.

- **Target Variant Name**

One or more replacement variants for ABAP program job steps in an SAP job.

Modify	Modifies an SAP job that already exists in an SAP system. A USAP job definition file is used to specify the modifications.	<ul style="list-style-type: none">• Script Library or File System Specifies whether the USAP definition file exists in the file system of the machine where the agent is running or in the Script Library.• Script Required (if Script is selected in Command or Script field). Name of the script that has been uploaded into the Script Library and will be executed by this task.• Definition File If you selected USAP Definition File above, use this field to provide the path and file name of the file.• SAP Job ID Job ID of the SAP job. Variables supported.
--------	--	---

Start	Starts a currently defined SAP job.	<ul style="list-style-type: none"> • SAP Job Name Job name of the SAP job. <i>Variables</i> supported. • SAP Job ID Job ID of the SAP job. <i>Variables</i> supported. • Start Immediately Enabled or disabled. Enabling the Start Immediately flag will cause the job to fail if SAP resources are not available to start the job immediately (for example, a background work process). Otherwise, the job will wait for SAP resources to become available. • SAP Target Server Name of an SAP instance at which a background job should be run. The name has the following format: <div style="border: 1px solid black; padding: 5px; margin: 10px 0; text-align: center;">[host name]_[SAP System name]_[SAP System number]</div>Where host name is the name of the server computer on which the instance is running, as specified in the system profile parameter SAPLOCALHOST. Example: <div style="border: 1px solid black; padding: 5px; margin: 10px 0; text-align: center;">hs0123_C11_55</div> • Wait Specifies whether Opwise should wait for the SAP process chain to complete processing. • Print Job Log Enabled or disabled. Specifies whether or not the job's joblog is returned. • Print Spooled Output Enabled or disabled. Specifies whether or not the spoolists of all job steps are returned. • Print Application Log Enabled or disabled. Specifies whether or not the job's application log, if one was generated, is returned. • Print Application RC Enabled or disabled. Specifies whether or not the job's application return codes, if they were set, are returned. • Use Application RC Specifies whether or not the SAP job's application return codes will be used to determine the return code for the Opwise task.
-------	-------------------------------------	---

<p>Start Process Chain</p>	<p>Starts the specified process chain on the SAP system.</p>	<ul style="list-style-type: none"> • Chain ID ID of process chain to start. • Restart Specification to restart failed and cancelled processes (R or X) in the specified process chain instance. • Log ID Log ID for process chain instance to be re-started. • Wait Specifies whether Opwise should wait for the SAP process chain to complete processing. • Print Job Log Enabled or disabled. Specifies whether or not the job's joblog is returned. • Print Spooled Output Enabled or disabled. Specifies whether or not the spoollists of all job steps are returned. • Print Application Log Enabled or disabled. Specifies whether or not the job's application log, if one was generated, is returned. • Print Application RC Enabled or disabled. Specifies whether or not the job's application return codes, if they were set, are returned. • Use Application RC Specifies whether or not the SAP job's application return codes will be used to determine the return code for the Opwise task.
<p>Start InfoPackage</p>	<p>Starts the specified InfoPackage on the SAP system.</p>	<ul style="list-style-type: none"> • InfoPackage Name of the InfoPackage to start. • InfoPackage Job Name Name of the SAP batch job that processes the InfoPackage request. • Wait Specifies whether Opwise should wait for the SAP InfoPackage to complete processing.

Wait	Reconnects to a started job and monitors it through completion.	<ul style="list-style-type: none"> • SAP Job Name Job name of the SAP job. Variables supported. • SAP Job ID Job ID of the SAP job. Variables supported. • Print Job Log Enabled or disabled. Specifies whether or not the job's joblog is returned. • Print Spooled Output Enabled or disabled. Specifies whether or not the spoolists of all job steps are returned. • Print Application Log Enabled or disabled. Specifies whether or not the job's application log, if one was generated, is returned. • Print Application RC Enabled or disabled. Specifies whether or not the job's application return codes, if they were set, are returned. • Use Application RC Specifies whether or not the SAP job's application return codes will be used to determine the return code for the Opwise task.
Wait Process Chain	Waits for a Process Chain to complete.	<ul style="list-style-type: none"> • Chain ID ID of process chain to be monitored to completion. • Log ID Log ID for process chain instance to be monitored to completion.
Wait InfoPackage	Waits for an InfoPackage to complete.	<ul style="list-style-type: none"> • Request ID Request ID of the InfoPackage that is to be monitored.
Abort	Cancels a running SAP job.	<ul style="list-style-type: none"> • SAP Job Name Job name of the SAP job. Variables supported. • SAP Job ID Job ID of the SAP job. Variables supported.
Interrupt Process Chain	Removes the specified process chain from the schedule.	<ul style="list-style-type: none"> • Chain ID ID of process chain that is to be interrupted.
Purge Job	Deletes a defined SAP job, its joblog, and all of its spoolists. This command is not available on SAP 3.1 and SAP 4.0.	<ul style="list-style-type: none"> • SAP Job Name Job name of the SAP job. Variables supported. • SAP Job ID Job ID of the SAP job. Variables supported.

Purge Variant	Deletes a variant from an SAP system.	<ul style="list-style-type: none">• SAP ABAP Program Name Name of the ABAP program for which the variant will be deleted.• SAP Variant Name Name of the variant to be deleted.
Raise Event	Raises the specified SAP background processing event.	<ul style="list-style-type: none">• SAP Event Name of the event.• SAP Event Parameter Optional parameter value for the event.

<p>Display</p>	<p>Displays the data specified in the Display Command field. The data is written to standard output.</p>	<ul style="list-style-type: none"> • Display Command <p>One of the following:</p> <ul style="list-style-type: none"> • Job Log - Displays the job log for a specified SAP job. • Spool List - Displays the spoolist for a job step. • Status - Displays the current status for an SAP job. • Variants - Displays the variants available for the specified ABAP program • Variant - Displays the contents of a specified variant. Note: Requires XBP interface 2.0 or greater. • Job Definition - Displays the definition of the specified SAP job. • Select - Displays a variety of attributes for a list of SAP jobs that match the specified criteria. • System Log - Displays a portion of an SAP syslog that meets the specified date/time constraints. • Intercept Table - Displays the contents of the job intercept criteria table for the connected SAP system. • Intercepted Jobs - Displays intercepted jobs for the connected SAP system. • Reports - Displays a list of ABAP reports that match the specified criteria. • Commands - Displays a list of SAP external commands that match the specified criteria. • Output Devices - Displays a list of SAP output devices that match the specified criteria. • Print Formats - Displays a list of print formats that are available for the specified printer. • Selection Screen - Displays information about the selection fields of an ABAP program. • Event History - Displays a list of events that were logged in an SAP system's event history. The retrieved events can optionally be set to "Confirmed". • Criteria Manager Profiles - Displays a list of Criteria Manager profiles. • Criteria Manager Criteria - Displays the criteria hierarchy of a particular profile in XML format. • Process Chains - Displays a list of process chains from the SAP system that meet the specified criteria. • Process Chain - Displays the list of processes contained within the specified process chain. • Process Chain Log - Displays the SAP log associated with the process chain. • Process Chain Start Condition - Displays the SAP start condition for specified process chain. • Process Chain Status - Displays the current status of the process chain. • InfoPackages - Displays a list of InfoPackages on the SAP system that meet the specified criteria. • InfoPackage Status - Displays the current status for the InfoPackage instance identified by the request ID.
<p>Generate Variant Definition</p>	<p>Generates a USAP variant definition file based on a model SAP variant. The generated definition file is written to standard output. Requires XBP interface 2.0 or greater.</p>	<ul style="list-style-type: none"> • SAP ABAP Program Name <p>Name of an ABAP program in an SAP system to which the model variant belongs.</p> <ul style="list-style-type: none"> • SAP Variant Name <p>Pre-existing SAP variant name to use as the model variant.</p>

<p>Generate Job Definition</p>	<p>Generates a USAP job definition file based on a model SAP job. The generated definition file is written to standard output.</p>	<ul style="list-style-type: none"> • SAP Job Name Job name of the SAP job. Variables supported. • SAP Job ID Job ID of the SAP job. Variables supported.
<p>Create CM Profile</p>	<p>Creates a new Criteria Manager profile.</p>	<ul style="list-style-type: none"> • Script Library or File System Specifies whether the USAP definition file exists in the file system of the machine where the agent is running or in the Script Library. • Script Required (if Script is selected in Command or Script field). Name of the script that has been uploaded into the Script Library and will be executed by this task. • SAP Criteria Manager XML File Name of the file that contains the Criteria Manager information. • Event Select State Event status of the events which should be read. • SAP Event Name of the event. • SAP Event Parameter Optional parameter value for the event. • Confirm Returned Events Specification for whether or not the status of returned events should be changed in the SAP system.

<p>Set CM Criteria</p>	<p>Sets the criteria for a profile.</p>	<ul style="list-style-type: none"> • Script Library or File System Specifies whether the USAP definition file exists in the file system of the machine where the agent is running or in the Script Library. • Script Required (if Script is selected in Command or Script field). Name of the script that has been uploaded into the Script Library and will be executed by this task. • SAP Criteria Manager XML File Name of the file that contains the Criteria Manager information. • SAP Criteria Manager Profile ID ID of the profile. • SAP Criteria Manager Profile Type Type of profile. For the default criteria types provided by SAP, the values are: <ul style="list-style-type: none"> • EVTHIS - Identifies a criteria type for event history. • EVHIRO - Identifies a criteria type for the reorganization of raised events. • INTERC - Identifies a criteria type for job interception. • Event Select State Event status of the events which should be read. • SAP Event Name of the event. • SAP Event Parameter Optional parameter value for the event. • Confirm Returned Events Specification for whether or not the status of returned events should be changed in the SAP system.
------------------------	---	--

<p>Activate CM Profile</p>	<p>Activates a criteria profile of the specified type.</p>	<ul style="list-style-type: none"> • SAP Criteria Manager Profile ID ID of the profile. • SAP Criteria Manager Profile Type Type of profile. For the default criteria types provided by SAP, the values are: <ul style="list-style-type: none"> • EVTHIS - Identifies a criteria type for event history. • EVHIRO - Identifies a criteria type for the reorganization of raised events. • INTERC - Identifies a criteria type for job interception. • Event Select State Event status of the events which should be read. • SAP Event Name of the event. • SAP Event Parameter Optional parameter value for the event. • Confirm Returned Events Specification for whether or not the status of returned events should be changed in the SAP system.
<p>Deactivate CM Profile</p>	<p>Deactivates a criteria profile of the specified type.</p>	<ul style="list-style-type: none"> • SAP Criteria Manager Profile Type Type of profile. For the default criteria types provided by SAP, the values are: <ul style="list-style-type: none"> • EVTHIS - Identifies a criteria type for event history. • EVHIRO - Identifies a criteria type for the reorganization of raised events. • INTERC - Identifies a criteria type for job interception. • Event Select State Event status of the events which should be read. • SAP Event Name of the event. • SAP Event Parameter Optional parameter value for the event. • Confirm Returned Events Specification for whether or not the status of returned events should be changed in the SAP system.

Delete CM Profile	Deletes a criteria profile from an SAP system.	<ul style="list-style-type: none"> • SAP Criteria Manager Profile ID ID of the profile. • SAP Criteria Manager Profile Type Type of profile. For the default criteria types provided by SAP, the values are: <ul style="list-style-type: none"> • EVTHIS - Identifies a criteria type for event history. • EVHIRO - Identifies a criteria type for the reorganization of raised events. • INTERC - Identifies a criteria type for job interception. • Event Select State Event status of the events which should be read. • SAP Event Name of the event. • SAP Event Parameter Optional parameter value for the event. • Confirm Returned Events Specification for whether or not the status of returned events should be changed in the SAP system.
-------------------	--	---

Built-In Variables

The built-in variables outlined below can be used in an SAP task to pass data where appropriate:

- [Task and Task Instance Variables](#)
- [Script Variables](#)
- [SAP Task Variables](#)
- [Agent Variables](#)

Specifying When a Task Runs

You can run the task as part of a [workflow](#), [specify triggers](#) that run the task automatically based on times or events, or [run the task manually](#).

Monitoring Task Execution

You can monitor all system activity from the [Activity screen](#) and can view activity history from the [Activity History screen](#).

File Transfer Task

- Overview
- FTP and SFTP File Transfer Tasks
 - Creating an FTP or SFTP File Transfer Task
 - FTP and SFTP File Transfer Task Field Descriptions
- INFITRAN File Transfer Tasks
 - Creating an INFITRAN File Transfer Task
 - Output Redirection
 - INFITRAN File Transfer Task Field Descriptions
- Specifying When a Task Runs
- Monitoring Task Execution
- Code Pages

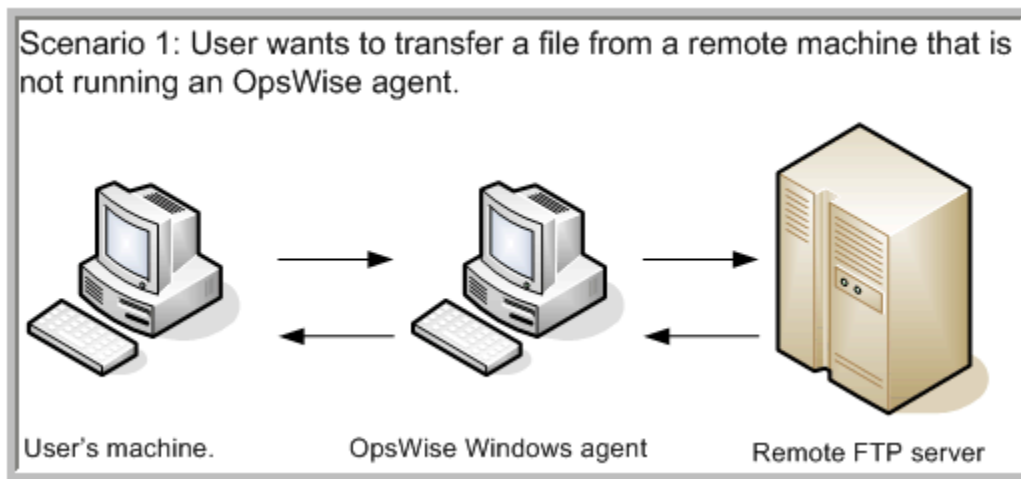
Overview

The File Transfer task allows you to execute file transfers on remote machines using any of the following protocols:

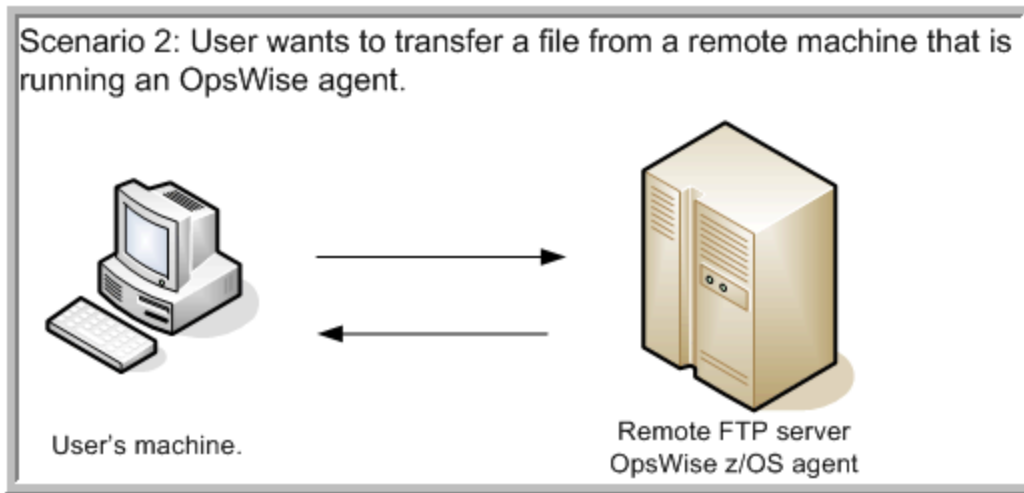
- FTP
- SFTP
- INFITRAN

To run a File Transfer task, you need an Opswise Linux/Unix, z/OS, or Windows agent to communicate with the File Transfer server. The agent can but does not have to be running on the same machine as the File Transfer server. The following diagrams provide sample configurations for executing file transfers using an Opswise File Transfer task.

In the example shown below, the user wants to transfer a file from a remote File Transfer Server on a machine that does not have an Opswise agent running on it. In this case, the File Transfer task definition provides an address and login credentials for the machine where the Opswise agent is running as well as address and login credentials for the machine where the File Transfer server is running.



In the example shown below, the user wants to transfer a file from a remote File Transfer Server on a z/OS machine that does have an Opswise agent running on it. In this case, the login credentials for the Opswise agent machine and the File Transfer server machine are the same.



FTP and SFTP File Transfer Tasks

The screens for FTP and SFTP File Transfer tasks are the same; the screen for [INFITRAN](#) file transfer tasks differs considerably.

Using **SFTP** requires that you supply a valid credential that specifies the location of the SSL Private key on your agent. In the Opswise Credentials form for the user you will use, you supply the location for the private key in the field "Key Location (File Transfer only)". This location must exist on the agent where you intend to run the SFTP task. For Linux/Unix, Automation Center does not at this time support password authentication for SFTP Transfer.

Make sure you have your private/public keys properly set up and working before you configure Opswise to use it. For example, to validate the keys, log into your destination server from your agent server using ssh.

Creating an FTP or SFTP File Transfer Task

Step 1 From the navigation pane, select **Automation Center > Tasks > File Transfer Tasks**. The File Transfer Tasks List screen displays.

Step 2 Click **New**. The File Transfer Task Definition screen displays. (This example shows the FTP file transfer method.)

Step 3 In the Transfer Type field, select FTP or SFTP. Using the field descriptions provided below as a guide, complete the fields as needed.

Step 4 Click the **Submit** button to save the record and return to the menu, or right-click on the title bar and select **Save** to save the record and remain on the current display.

Step 5 If appropriate, repeat these steps for any additional tasks you want to add.

Task run statistics appear after the first time this task has been launched.

FTP and SFTP File Transfer Task Field Descriptions

The table below describes the fields, buttons, and tabs on the task definition and task instance screens. Color coding is provided that differentiates the following three types of fields:

- Fields that display on the task definition and task instance screens are shown in black.
- Fields that display only on the task definition screen are shown in green.
- Fields that display only on the task instance screen are shown in maroon.

Field Name	Description
Task/Instance Name	Required. Name used within Opswise to identify this task. Up to 40 alphanumeric; variables supported. It is the user's responsibility to develop a workable naming scheme for tasks.
Version	Task definition only; system-supplied. The version number of the current record, which is incremented by Opswise every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning.

Invoked by	<p>Task instance only; system-supplied. How the task instance was launched. One of the following:</p> <ul style="list-style-type: none"> • Trigger: (Trigger Name) - The instance was launched by the named trigger. • Workflow: (Workflow Name) - The instance was launched by the named workflow. • Manually Launched - The instance was launched by a user. To determine the name of the user: <ol style="list-style-type: none"> 1. From the Activity or Task Instances screen, click the task instance name to open the record. 2. The Execution User field identifies the user who launched the task instance.
Task	Task instance only; system-supplied. Hover over the paper icon to display more information about the task instance. Click the paper icon to display the task definition record.
Execution User	Task instance only; system-supplied. If the task was launched manually, the ID of the user who launched it.
Instance Reference Id	Task instance only; system-supplied. Opwise increments this number each time the task is run.
z/OS ID	Task instance only; z/OS only. The z/OS execID, used internally by the z/OS agent to identify each z/OS task.
Task Description	User-supplied description of this record.
Hold on Start	If enabled, when the task is launched it appears in the Activity display with a status of Held. The task runs when the user releases it.
Hold Reason	Information about why the task will be put on hold when it starts.
Member of Business Services	User-defined. Allows you to select one or more Business Services that this record definition belongs to. Click the lock icon to unlock the field and select Business Services .
User Estimated Duration	Task definition only; optional. The estimated amount of time it should normally take to run this task. Opwise uses this information to calculate the User Estimated End Time on a task instance record.
Status	Task instance only; system-supplied. See Task Instance Statuses .
Exit Code	Task instance only; system-supplied. The exit code captured by the agent when executing the task (for example, a command or script).
Status Description	Task instance only; system-supplied. Provides additional information, if any, about the status of the task.
Start Time	Task instance only; system-supplied. The date and time the task started.
Duration	Task instance only; system-supplied. The amount of time the task took to run.
End Time	Task instance only; system-supplied. The date and time the task instance completed.

Transfer Type	Type of File Transfer server. Options: <ul style="list-style-type: none"> • FTP • SFTP • INFITRAN
Command	File Transfer command being executed. Options: <ul style="list-style-type: none"> • GET - Copies a remote file to the local computer. • PUT - Copies a local file to the remote computer. • MGET - Copies multiple remote files to the local computer. • MPUT - Copies multiple local files to the remote computer. • DELETE - Deletes the specified file from the remote computer. • MDELETE - Deletes the specified file(s) from the remote computer. • MKDIR - Creates the specified directory on the remote computer. • RMDIR - Removes the specified directory from the remote computer.
Subcommands (z/OS only)	For z/OS, any subcommands used in the JCL statement.
Job Card (z/OS only)	For z/OS, the job card information for the JCL statement. Example: <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <pre>//File TransferJOB01 JOB (File Transfer,001),FANNY,MSGCLASS=X,MSGLEVEL=(1,1),NOTIFY=&SYSUID,CLASS=A</pre> </div>
Agent	Optional. The name of the agent resource definition that identifies the machine where the operation will run. If you do not specify an agent, you must specify an agent cluster (see below).
Transfer Method	Transfer method. Options: <ul style="list-style-type: none"> • Active • Passive • Extended Passive
Agent Variable	Optional. If enabled, the Agent field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: <code>\${variable name}</code> . The variable must be a supported type as described in Variables and Functions .
Transfer Mode	Mode of data transfer. Options: <ul style="list-style-type: none"> • Binary • ASCII
Agent Cluster	Optional. You can specify an agent cluster in addition to or in place of a specific agent. An agent cluster is a group of agents, one of which Opwise will choose to run this task. If you specify an agent and an agent cluster, Opwise Automation Center first tries to run the task on the specific agent. If the agent is not available, Opwise reverts to the agent cluster. See Agent Clusters for more information.
Remote Server	Required if FTP or SFTP is selected in the Transfer Type: field. Name or IP address of the File Transfer server. This machine may or may not be the same as the Opwise agent machine. <p>You also can specify a non-standard FTP or SFTP port:</p> <ul style="list-style-type: none"> • For FTP, specify the port number separated from the host name with a space: "some.server.com 2222". • For SFTP, specify the port number separated from the host name with a colon: "some.server.com:2222".

Agent Cluster Variable	Optional. If enabled, the Agent Cluster field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: \${variable name}. The variable must be a supported type as described in Variables and Functions .
FTP Credentials	Login credentials the Opwise agent will use to access the FTP or SFTP server machine. If the File Transfer server and Opwise agent are running on the same machine, enter the same credentials as those you entered in the Credentials field.
Cluster Broadcast	Task definition only; optional. You can specify a Cluster Broadcast in place of a specific agent and/or Agent Cluster. When you specify an agent cluster in the Cluster Broadcast field, Opwise Automation Center runs the task on all the agents in the cluster. Each instance of the task running on its own agent becomes a separate task instance record in the database and displays separately in the Activity monitor. See Agent Clusters for more information about defining agent clusters.
FTP Credentials Variable	Optional. If enabled, the FTP Credentials field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: \${variable name}. The variable must be a supported type as described in Variables and Functions .
Credentials	Credentials under which an Agent runs this task. These credentials override any credentials provided in the Agent resource definition for any Agent running this task.
Credentials Variable	Optional. If enabled, the Credentials field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: \${variable name}. The variable must be a supported type as described in Variables and Functions .
Local Filename	Required if Transfer Type = FTP or SFTP. Path and file name on the local server. That is, the "transfer from" file name.
Remote Filename	Required if Transfer Type = FTP or SFTP. Path and file name on the remote server. That is, the "transfer to" file name.
Maximum Retries	User-defined. The maximum number of times Opwise should retry this task after it has started and gone to a failed state.
Retry Indefinitely	User-defined. Enabled or disabled. Indicates whether Opwise should continue trying indefinitely to run this task. If you enable this field, it overrides any value placed in the Maximum Retries field (above).
Retry Interval	User-defined. The number of seconds between each retry.
Current Retry Count	Task instance only; system-supplied. Displays, only for a running task instance, the current number of times that Opwise has retried the task after it first went to failure status.
User Estimated End Time	Task instance only; system-supplied. If the user entered information into the User Estimated Duration field in the task definition, Opwise uses this information to calculate an end time for the task instance, based on the date/time the task instance started.
Shortest Estimated End Time	Task instance only; system-supplied.
Average Estimated End Time	Task instance only; system-supplied.

Longest Estimated End Time	Task instance only; system-supplied.
Late Start	If enabled, and if the task instance starts after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late start (see Late Start Type). To determine whether a task instance started late, open the task instance and locate the Started Late field; the field is checked if the instance started after the specified time. This field only appears on the task instance if the user specified a Late Start in the task definition.
Late Start Type	Required if Late Start is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it starts after the specified time. • Duration - Flag the task if it starts a certain amount of time after the programmed start time. The task must have a specific start time.
Late Start Time	Time after which the task start time is considered late. Use hh:mm, 24-hour time
Late Start Duration	Duration (amount of relative time) after which the task is considered to have started late. For a task within a workflow, the duration is the period between the time the workflow starts and the time the task itself starts. For example, a task might have a Late Start Duration of 60 minutes. If the workflow starts at 9:00 a.m. but the task itself does not start until 10:30, the task has started late. For a task that is not within a workflow, Late Start Duration has meaning only if the task has been held upon starting. For example, if a task has a Late Start Duration of 60 minutes and the Hold on Start field is enabled, if the task is not released from hold within the amount of time specified in the Late Start Duration field, the task has started late.
Started Late	Task instance only; system-supplied. This field is flagged if the task started later than the time specified in the Late Start fields.
Late Finish	If enabled, and if the task instance finishes after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late finish (see Late Finish Type). To determine whether a task instance finished late, open the task instance and locate the Finished Late field; the field is checked if the instance finished after the specified time or lasted longer than expected. This field only appears on the task instance if the user specified a Late Finish in the task definition.
Late Finish Type	Required if Late Finish is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it finishes after the specified time (see Late Finish Time). • Duration - Flag the task if it finishes a certain amount of time after the programmed finish time (see Late Finish Duration). The task must have a specific finish time.
Late Finish Time	If Late Finish Type is Time, use this to specify the time after which the task finish time is considered late. Use hh:mm, 24-hour time.
Late Finish Duration	If Late Finish Type is Duration, use this to specify the longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Finished Late	Task instance only; system-supplied. This field is flagged if the task finished later than the time or duration specified in the Late Finish fields.
Early Finish	If enabled, and if the task instance finishes before the time or period specified, the task instance is flagged as early. You can specify a time or duration to determine an early finish (see Early Finish Type). To determine whether a task instance finished early, open the task instance and locate the Finished Early field; the field is checked if the instance finished before the specified time or did not last as long as expected. This field only appears on the task instance if the user added Early Finish specifications to the task definition.

Early Finish Type	<p>Required if Early Finish is enabled. Options are:</p> <ul style="list-style-type: none"> • Time - Flag the task if it finishes before the specified time (see Early Finish Time). • Duration - Flag the task if it finishes a certain amount of time before the programmed finish time (see Early Finish Duration). The task must have a specific finish time.
Early Finish Time	If Early Finish Type is Time, use this to specify the time before which the task finish time is considered early. That is, enter a time at which the task should still be running. Use hh:mm, 24-hour time.
Early Finish Duration	If Early Finish Type is Duration, use this to specify the shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Finished Early	Task instance only; system-supplied. This field is flagged if the task finished earlier than the time specified in the Early Finish fields.
Virtual Resource Priority	<p>Priority for acquiring a resource when two or more tasks are waiting for the resource. This priority applies to all resources required by the task.</p> <p>Options: 1 (high) - 20 (low).</p> <p>Default is 10.</p>
Hold Resources on Failure	If enabled, the task instance will continue to hold Renewable resources if the task instance fails. Renewable resources will be returned only if the task instance status is either Complete, Finished, or Skipped.
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Launch Task button	Manually launches the task.
View Instances button	Displays a list of task instances for which there has been a status change or a modification to the task instance record within the last 30 days (an Updated on Last 30 Days filter has been pre-selected for the list).
Delete button	Deletes the current record.
View Parent button	Task instance only; Displays this task's parent task (workflow), if any.
Show Details button	Task instance only; displays detailed information about this task instance.
Retrieve Output button	Task instance only; see Retrieving Output.
Skip button	Task instance only; for tasks loaded into the schedule that haven't run yet. Allows you to tell Opwise to skip this task. See Skipping a Task.

Hold button	Task instance only; see Putting a Task on Hold .
Force Finish button	Task instance only; see Force Finishing a Task .
Output tab	Task instance only. Displays output generated from the process, if any, based on specifications provided by the user in the Automatic Output Retrieval fields in the task definition.
Variables tab	Displays all variables associated with this record.
Actions tab	<p>Allows you to specify actions that Opwise will take automatically based on events that occur during the execution of this task. Events are task instance status, exit codes, late start, late finish, and early finish. Actions are:</p> <ul style="list-style-type: none"> • Abort Action - Abort the task if certain events occur. For details, see Setting Abort Actions. • Email Notification - Send an email if certain events occur. For details, see Creating Email Notifications. • Set Variable - Used in workflows to set a variable based on the occurrence of certain events. For details, see Setting Variables within a Workflow. • SNMP Notification - Send an email if certain events occur. For details, see Creating SNMP Notifications. • System Operation - Run an Opwise system operation based on specified conditions. For details, see Setting System Operations.
Task Virtual Resources tab	Lists Virtual Resources to which this task is assigned.
Mutually Exclusive Tasks tab	Displays all tasks that have been set to be mutually exclusive of this task.
Triggers tab	Displays a list of all triggers that have been defined to launch this task. Also allows you to add new triggers. If you add a new trigger from this location, Opwise automatically constructs a default trigger name as follows: <current task name>#TRIGGER#. You can change the default name if desired. For instructions on creating triggers, see Creating Triggers .
Notes tab	Displays all notes associated with this task.
Versions tab	Stores copies of all previous versions of the current record. See Record Versioning .

INFITRAN File Transfer Tasks

The screen for INFITRAN file transfers differs considerably from [FTP](#) and [SFTP](#).

Creating an INFITRAN File Transfer Task

Step 1	From the navigation pane, select Automation Center > Tasks > File Transfer Tasks .
---------------	---

Step 2 Click **New**. The File Transfer Task screen displays.

The screenshot shows the 'File Transfer Task' configuration window. The title bar includes tabs for 'File Transfer Task', 'Variables', 'Actions', 'Task Virtual Resources', 'Mutually Exclusive Tasks', 'Triggers', 'Notes', and 'Versions'. The main area contains the following fields and controls:

- Task Name:** [Text input field]
- Version:** [Text input field with value '1']
- Task Description:** [Text area]
- Hold on Start:**
- Member of Business Services:**
- User Estimated Duration:** [Time picker showing 00:00:00]
- Transfer Type:** [Dropdown menu showing INFITRAN]
- Utility Agent:** [Text input field]
- Utility Agent Variable:**
- Utility Agent Cluster:** [Text input field]
- Utility Agent Cluster Variable:**
- Utility Credentials:** [Text input field]
- Utility Credentials Variable:**
- Transfer Mode:** [Dropdown menu showing Binary]
- Encrypt:** [Dropdown menu showing NO]
- Compress:** [Dropdown menu showing NO]
- Codepage:** [Dropdown menu showing -- None --]
- File Creation Option:** [Dropdown menu showing -- None --]
- Trim Trailing Spaces:**
- Network Fault Tolerant:**
- Runtime Directory:** [Text input field]
- Source Filename(s):** [Text input field]
- Source File System:** [Dropdown menu showing -- None --]
- Source Infitran Agent:** [Text input field]
- Source Infitran Agent Option:** [Dropdown menu showing Infitran Agent]
- Source Credentials:** [Text input field]
- Source Credentials Variable:**
- Append Source Open Options:** [Text input field]
- Append Infitran Options:** [Text area]
- Destination Filename(s):** [Text input field]
- Destination File System:** [Dropdown menu showing -- None --]
- Destination Infitran Agent:** [Text input field]
- Destination Infitran Agent Option:** [Dropdown menu showing Infitran Agent]
- Destination Credentials:** [Text input field]
- Destination Credentials Variable:**
- Append Destination Open Options:** [Text input field]
- Exit Code Processing:** [Dropdown menu showing Success Exitcode Range]
- Exit Codes:** [Text input field with value 0]
- Late Start:**
- Late Finish:**
- Early Finish:**
- Maximum Retries:** [Text input field with value 0]
- Retry Interval (Seconds):** [Text input field with value 60]
- Virtual Resource Priority:** [Dropdown menu showing 10]
- Retry Indefinitely:**
- Hold Resources on Failure:**

A **Submit** button is located at the bottom left of the form.

Step 3 In the Transfer Type field, select INFITRAN. Using the field descriptions provided below as a guide, complete the fields as needed.

Step 4 Click the **Submit** button to save the record and return to the menu, or, right-click on the title bar and select **Save** to save the record and remain on the current display.

Step 5 If appropriate, repeat these steps for any additional tasks you want to add.

Task run statistics appear after the first time this task has been launched.

Output Redirection

Universal Automation Center Agent (UAG) processes File Transfer/Infitran, Indesca, and SAP task types differently than Windows and Linux/Unix task types. File Transfer/Infitran, Indesca, and SAP command lines are sent to the user process via standard input, so any redirection operators entered as task command input are not processed as expected.

If you want to direct output from a File Transfer/Infitran task to your file system, the **-uagstdio** command option lets you specify the same output redirection commands that are available for Windows and Linux/Unix task types. UAG will apply the user-specified value for **-uagstdio** directly to the command image.

The I/O redirection commands that you can use with **-uagstdio** are dependent on the OS/command shell. You should be able to set up any

redirection that the OS/command shell supports (just as with Windows and Unix/Linux task types).

The syntax of **-uagstdio** is similar to Universal Data Mover, Universal Command, and Universal Connector [command line options](#); option followed by value.

For the File Transfer/Infitran task type, you can specify **uagstdio** in the following field:

- [Append Infitran Options](#)

-uagstdio Examples

```
-uagstdio >C:\INFITRANAOUT\infitran.out
```

If the **-uagstdio** value contains spaces, it must be enclosed in double quotation marks ("):

```
-uagstdio ">C:\INFITRANAOUT\infitran.out 2>C:\INDFITRANAOUT\infitran.err"
```

If the quoted value itself requires double quotation marks, they must be doubled (""):

```
-uagstdio ">C:\tmp\"\"infitran output\"\"\infitran.out 2>C:\tmp\"\"infitran output\"\"\infitran.err"
```

INFITRAN File Transfer Task Field Descriptions

The table below describes the fields, buttons, and tabs on the task definition and task instance screens. Color coding is provided that differentiates the following three types of fields:

- Fields that display on the task definition and task instance screens are shown in black.
- Fields that display only on the task definition screen are shown in green.
- Fields that display only on the task instance screen are shown in maroon.

Field Name	Description
Task/Instance Name	Required. Name used within Opswise to identify this task. Up to 40 alphanumeric; variables supported. It is the user's responsibility to develop a workable naming scheme for tasks.
Version	Task definition only; system-supplied. The version number of the current record, which is incremented by Opswise every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning.
Description	User-supplied description of this record.
Invoked by	Task instance only; system-supplied. How the task instance was launched. One of the following: <ul style="list-style-type: none"> • Trigger: (Trigger Name) - The instance was launched by the named trigger. • Workflow: (Workflow Name) - The instance was launched by the named workflow. • Manually Launched - The instance was launched by a user. To determine the name of the user: <ol style="list-style-type: none"> 1. From the Activity or Task Instances screen, click the task instance name to open the record. 2. The Execution User field identifies the user who launched the task instance.
Task	Task instance only; system-supplied. Hover over the paper icon to display more information about the task instance. Click the paper icon to display the task definition record.

Execution User	Task instance only; system-supplied. If the task was launched manually, the ID of the user who launched it.
Instance Reference Id	Task instance only; system-supplied. Opswise increments this number each time the task is run.
Hold on Start	If enabled, when the task is launched it appears in the Activity display with a status of Held. The task runs when the user releases it.
Hold Reason	Information about why the task will be put on hold when it starts.
Member of Business Services	User-defined. Allows you to select one or more Business Services that this record definition belongs to. Click the lock icon to unlock the field and select Business Services .
User Estimated Duration	Task definition only; optional. The estimated amount of time it should normally take to run this task. Opswise uses this information to calculate the User Estimated End Time on a task instance record.
Status	Task instance only; system-supplied. See Task Instance Statuses .
Exit Code-Task Instance	Task instance only; system-supplied. The exit code captured by the agent when executing the task (for example, a command or script).
Status Description	Task instance only; system-supplied. Provides additional information, if any, about the status of the task.
Start Time	Task instance only; system-supplied. The date and time the task started.
Duration	Task instance only; system-supplied. The amount of time the task took to run.
End Time	Task instance only; system-supplied. The date and time the task instance completed.
Transfer Type	Type of File Transfer server. Options: <ul style="list-style-type: none"> • FTP • SFTP • INFITRAN
Utility Agent	Required. The name of the Windows, Linux/Unix, or z/OS agent resource that will communicate with the Indesca agent. If you do not specify an agent, you must specify an agent cluster (see below).
Transfer Mode	Mode of data transfer. Options: <ul style="list-style-type: none"> • Binary • ASCII
Utility Agent Variable	Optional. If enabled, the Utility Agent field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: \${variable name}. The variable must be a supported type as described in Variables and Functions .

Encrypt	<p>The method of encryption that Opwise will use in the transfer. Options:</p> <ul style="list-style-type: none"> • YES • NO (none) • RC4-SHA • RC4-MD5 • AES256-SHA • AES128-SHA • DES-CBC3-SHA • DES-CBC-SHA • NULL-SHA • NULL-MD5 • NULL-NULL
Utility Agent Cluster	<p>Optional. You can specify an agent cluster in addition to or in place of a specific agent. An agent cluster is a group of agents, one of which Opwise will choose to run this task. If you specify an agent and an agent cluster, Opwise Automation Center first tries to run the task on the specific agent. If the agent is not available, Opwise reverts to the agent cluster. See Agent Clusters for more information.</p>
Compress	<p>The type of data compression used in the transfer, if any. Options:</p> <ul style="list-style-type: none"> • YES • NO • ZLIB • HASP
Utility Agent Cluster Variable	<p>Optional. If enabled, the Agent Cluster field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: <code>\${variable name}</code>. The variable must be a supported type as described in Variables and Functions.</p>
Codepage	<p>Options: (see Code Pages, below)</p>
Utility Credentials	<p>Login credentials the Opwise agent will use to access the Indesca server machine.</p>
File Creation Option	<p>Specifies whether the transferred file should be created (new), appended, or replace any existing file. Options:</p> <ul style="list-style-type: none"> • None • APPEND • NEW • REPLACE
Utility Credentials Variable	<p>Optional. If enabled, the Utility Credentials field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: <code>\${variable name}</code>. The variable must be a supported type as described in Variables and Functions.</p>
Trim Trailing Spaces	<p>Enabled or not. Specifies whether Opwise should trim trailing spaces from lines on an ASCII transfer.</p>
Network Fault Tolerant	<p>Enable if the session is network fault tolerant.</p>
Runtime Directory	<p>Optional. The directory from which the application should be executed. Variables supported.</p>
Source Filename	<p>Required if Transfer Type = INFITRAN. Path and file name on the source Infitran server.</p>

Destination Filename	Required if Transfer Type = INFITRAN. Path and file name on the destination INFITRAN server.
Source File System	Type of file system on the source server. Options: <ul style="list-style-type: none"> • None • DSN • HFS • LIB
Destination File System	Type of file system on the destination server. Options: <ul style="list-style-type: none"> • None • DSN • HFS • LIB
Source Infitran Agent	Required if Transfer Type = INFITRAN. Name of the agent resource defined in Opwise that describes the source Indesca agent machine (primary transfer server).
Destination Infitran Agent	Required if Transfer Type = INFITRAN. Name of the agent resource defined in Opwise that provides details about the destination Indesca agent machine (secondary transfer server).
Source Infitran Agent Option	Defines how you will specify the Source Infitran Agent. Options: <ul style="list-style-type: none"> • Infitran Agent - The source agent is an Indesca/Infitran agent defined in Opwise. • Infitran Agent Variable - The source agent will be defined by setting the variable in the Source Infitran Agent field. • Infitran Agent Hostname - The source agent runs on the hostname specified in the Source Infitran Agent field.
Destination Infitran Agent Option	Defines how you will specify the Destination Infitran Agent. Options: <ul style="list-style-type: none"> • Infitran Agent - The destination agent is an Indesca/Infitran agent defined in Opwise. • Infitran Agent Variable - The destination agent will be defined by setting the variable in the Destination Infitran Agent field. • Infitran Agent Hostname - The destination agent runs on the hostname specified in the Destination Infitran Agent field.
Source Credentials	Specifies the source user ID and password (local to the host on which the server is running) under which the transfer operation is being carried out.
Destination Credentials	Specifies the destination user ID and password (local to the host on which the server is running) under which the transfer operation is being carried out.
Source Credentials Variable	Optional. If enabled, the Source Credentials field (see above) converts from a reference field (where you browse and select a credentials record from the Opwise credentials table) into a text field that allows you to enter a variable. Use the format: \${variable name}. The variable must be a supported type as described in Variables and Functions .
Destination Credentials Variable	Optional. If enabled, the Destination Credentials field (see above) converts from a reference field (where you browse and select a credentials record from the Opwise credentials table) into a text field that allows you to enter a variable. Use the format: \${variable name}. The variable must be a supported type as described in Variables and Functions .
Append Source Open Options	Optional. Any additional free form open command options for the source (primary) transfer server.

Append Destination Open Option	Optional. Any additional free form open command options for the destination (secondary) transfer server.
Append Infitran Options	Optional. Any additional free form Universal Data Mover command options.
Exit Code Processing	<p>Required. Specifies how Opwise Automation Center should determine whether the executed command failed or completed successfully. Options:</p> <ul style="list-style-type: none"> • Success Exitcode Range - The command is considered completed successfully if its exit code falls within the range specified in the Exit Codes field (see below). • Failure Exitcode Range - The command is considered failed if its exit code falls within the range specified in the Exit Codes field (see below). • Success Output Contains - The command is considered completed successfully if its output contains the text specified in the Scan Output For field (see below). • Failure Output Contains - The command is considered failed if its output contains the text specified in the Scan Output For field (see below). • Step Conditions - The command is considered completed successfully/failed if any of its specified condition codes falls within the range specified under the Step Conditions tab (see Creating Step Conditions).
Output Type-Exit Code	<p>Required if Exit Code Processing = Success Output Contains or Failure Output Contains. This field specifies the type of output. Options:</p> <ul style="list-style-type: none"> • Standard Output (STDOUT) • Standard Error (STDERR) • File
Exit Codes	<p>Required if Exit Code Processing = Success Exitcode Range or Failure Exit Code Range. This field specifies the range. Format: Numeric. Use commas to list a series of exitcodes; use hyphens to specify a range. Example: 1,5, 22-30.</p>
Scan Output For	<p>Required if Exit Code Processing = Success Output Contains or Failure Output Contains. This field specifies the text for which Opwise should scan the output file. Opwise will process this field as a regular expression.</p>
Output File-Exit Codes	<p>Required if Output Type = File. This field specifies the path and file name of the output file that should be scanned for the text in the Scan Output For field.</p>
Maximum Retries	<p>User-defined. The maximum number of times Opwise should retry this task after it has started and gone to a failed state.</p>
Retry Indefinitely	<p>User-defined. Enabled or disabled. Indicates whether Opwise should continue trying indefinitely to run this task. If you enable this field, it overrides any value placed in the Maximum Retries field (above).</p>
Retry Interval	<p>User-defined. The number of seconds between each retry.</p>
Current Retry Count	<p>Task instance only; system-supplied. Displays, only for a running task instance, the current number of times that Opwise has retried the task after it first went to failure status.</p>
User Estimated End Time	<p>Task instance only; system-supplied. If the user entered information into the User Estimated Duration field in the task definition, Opwise uses this information to calculate an end time for the task instance, based on the date/time the task instance started.</p>
Shortest Estimated End Time	<p>Task instance only; system-supplied.</p>

Average Estimated End Time	Task instance only; system-supplied.
Longest Estimated End Time	Task instance only; system-supplied.
Late Start	If enabled, and if the task instance starts after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late start (see Late Start Type). To determine whether a task instance started late, open the task instance and locate the Started Late field; the field is checked if the instance started after the specified time. This field only appears on the task instance if the user specified a Late Start in the task definition.
Late Start Type	Required if Late Start is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it starts after the specified time. • Duration - Flag the task if it starts a certain amount of time after the programmed start time. The task must have a specific start time.
Late Start Time	Time after which the task start time is considered late. Use hh:mm, 24-hour time
Late Start Duration	Duration (amount of relative time) after which the task is considered to have started late. For a task within a workflow, the duration is the period between the time the workflow starts and the time the task itself starts. For example, a task might have a Late Start Duration of 60 minutes. If the workflow starts at 9:00 a.m. but the task itself does not start until 10:30, the task has started late. For a task that is not within a workflow, Late Start Duration has meaning only if the task has been held upon starting. For example, if a task has a Late Start Duration of 60 minutes and the Hold on Start field is enabled, if the task is not released from hold within the amount of time specified in the Late Start Duration field, the task has started late.
Started Late	Task instance only; system-supplied. This field is flagged if the task started later than the time specified in the Late Start fields.
Late Finish	If enabled, and if the task instance finishes after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late finish (see Late Finish Type). To determine whether a task instance finished late, open the task instance and locate the Finished Late field; the field is checked if the instance finished after the specified time or lasted longer than expected. This field only appears on the task instance if the user specified a Late Finish in the task definition.
Late Finish Type	Required if Late Finish is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it finishes after the specified time (see Late Finish Time). • Duration - Flag the task if it finishes a certain amount of time after the programmed finish time (see Late Finish Duration). The task must have a specific finish time.
Late Finish Time	If Late Finish Type is Time, use this to specify the time after which the task finish time is considered late. Use hh:mm, 24-hour time.
Late Finish Duration	If Late Finish Type is Duration, use this to specify the longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Finished Late	Task instance only; system-supplied. This field is flagged if the task finished later than the time or duration specified in the Late Finish fields.

Early Finish	If enabled, and if the task instance finishes before the time or period specified, the task instance is flagged as early. You can specify a time or duration to determine an early finish (see Early Finish Type). To determine whether a task instance finished early, open the task instance and locate the Finished Early field; the field is checked if the instance finished before the specified time or did not last as long as expected. This field only appears on the task instance if the user added Early Finish specifications to the task definition.
Early Finish Type	Required if Early Finish is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it finishes before the specified time (see Early Finish Time). • Duration - Flag the task if it finishes a certain amount of time before the programmed finish time (see Early Finish Duration). The task must have a specific finish time.
Early Finish Time	If Early Finish Type is Time, use this to specify the time before which the task finish time is considered early. That is, enter a time at which the task should still be running. Use hh:mm, 24-hour time.
Early Finish Duration	If Early Finish Type is Duration, use this to specify the shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Finished Early	Task instance only; system-supplied. This field is flagged if the task finished earlier than the time specified in the Early Finish fields.
Virtual Resource Priority	Priority for acquiring a resource when two or more tasks are waiting for the resource. This priority applies to all resources required by the task. Options: 1 (high) - 20 (low). Default is 10.
Hold Resources on Failure	If enabled, the task instance will continue to hold Renewable resources if the task instance fails. Renewable resources will be returned only if the task instance status is either Complete, Finished, or Skipped.
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Launch Task button	Manually launches the task.
View Instances button	Displays a list of task instances for which there has been a status change or a modification to the task instance record within the last 30 days (an Updated on Last 30 Days filter has been pre-selected for the list).
Delete button	Deletes the current record.
View Parent button	Task instance only; Displays this task's parent task (workflow), if any.
Show Details button	Task instance only; displays detailed information about this task instance.

Skip button	Task instance only; for tasks loaded into the schedule that haven't run yet. Allows you to tell Opwise to skip this task. See Skipping a Task .
Hold button	Task instance only; see Putting a Task on Hold .
Force Finish button	Task instance only; see Force Finishing a Task .
Re-run button	Task instance only; see Re-running a Task .
Output tab	Task instance only. Displays output generated from the process, if any, based on specifications provided by the user in the Automatic Output Retrieval fields in the task definition.
Variables tab	Displays all variables associated with this record.
Actions tab	Allows you to specify actions that Opwise will take automatically based on events that occur during the execution of this task. Events are task instance status, exit codes, late start, late finish, and early finish. Actions are: <ul style="list-style-type: none"> • Abort Action - Abort the task if certain events occur. For details, see Setting Abort Actions. • Email Notification - Send an email if certain events occur. For details, see Creating Email Notifications. • Set Variable - Used in workflows to set a variable based on the occurrence of certain events. For details, see Setting Variables within a Workflow. • SNMP Notification - Send an email if certain events occur. For details, see Creating SNMP Notifications. • System Operation - Run an Opwise system operation based on specified conditions. For details, see Setting System Operations.
Task Virtual Resources tab	Lists Virtual Resources to which this task is assigned.
Mutually Exclusive Tasks tab	Displays all tasks that have been set to be mutually exclusive of this task.
Triggers tab	Displays a list of all triggers that have been defined to launch this task. Also allows you to add new triggers. If you add a new trigger from this location, Opwise automatically constructs a default trigger name as follows: <current task name>#TRIGGER#. You can change the default name if desired. For instructions on creating triggers, see Creating Triggers .
Notes tab	Displays all notes associated with this task.
Versions tab	Stores copies of all previous versions of the current record. See Record Versioning .

Specifying When a Task Runs

You can run the task as part of a [workflow](#), [specify triggers](#) that run the task automatically based on times or events, or [run the task manually](#).

Monitoring Task Execution

You can monitor all system activity from the [Activity screen](#) and can view activity history from the [Activity History screen](#).

Code Pages

The following table identifies all supported code pages for an [INFITRAN File Transfer task](#).

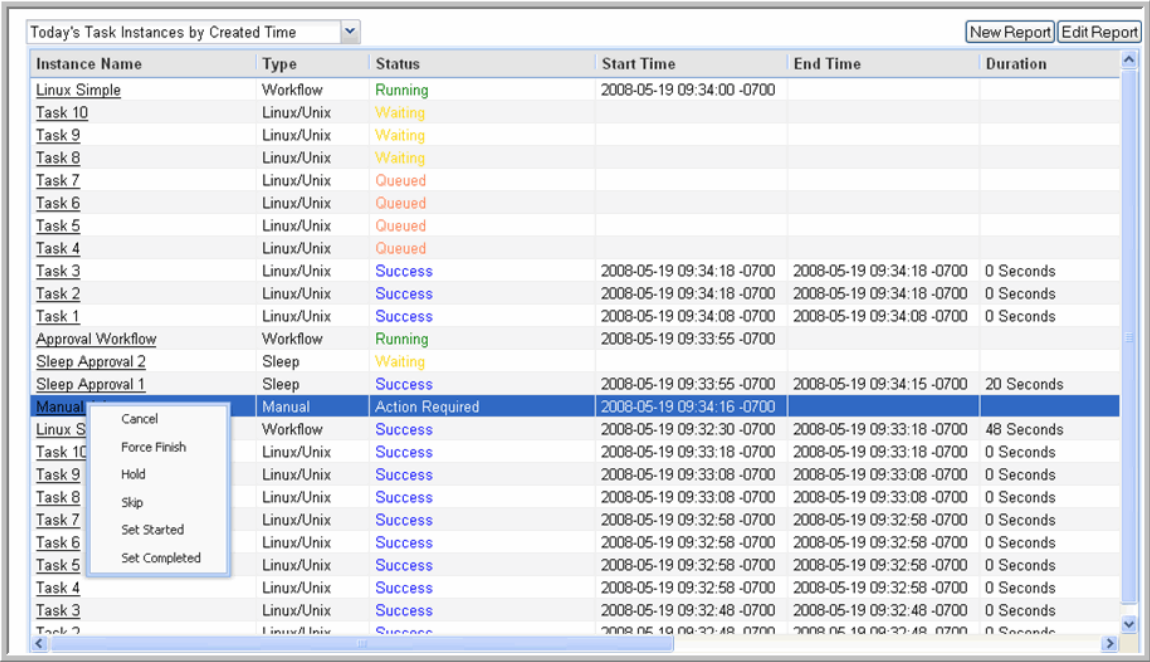
ISO8859-1	op437	IBM Portugal 037
ISO8859-2	op737	IBM German 273
ISO8859-3	op775	IBM Danish and Norwegian 277
ISO8859-4	op850	IBM Sweden and Finland 278
ISO8859-5	op852	IBM Italian 280
ISO8859-6	op855	IBM Spanish 284
ISO8859-7	op857	IBM International 500
ISO8859-8	op860	IBM Greek 875
ISO8859-9	op861	IBM Latin-1 1047
ISO8859-10	op862	IBM Portugal 1140
ISO8859-13	op863	IBM German 1141
ISO8859-14	op864	IBM Danish 1142
ISO8859-15	op865	IBM Finish 1143
	op866	IBM Italian 1144
	op869	IBM Spanish 1145
	op874	IBM UK 1146
	op 1250	IBM Swiss 1148
	op 1251	IBM Greek 4971
	op 1252	
	op 1253	
	op 1254	
	op 1255	
	op 1256	
	op 1257	
	op 1258	

Manual Task

- Overview
- Creating a New Manual Task
- Manual Task Field Descriptions
- Specifying When a Task Runs
- Monitoring Task Execution

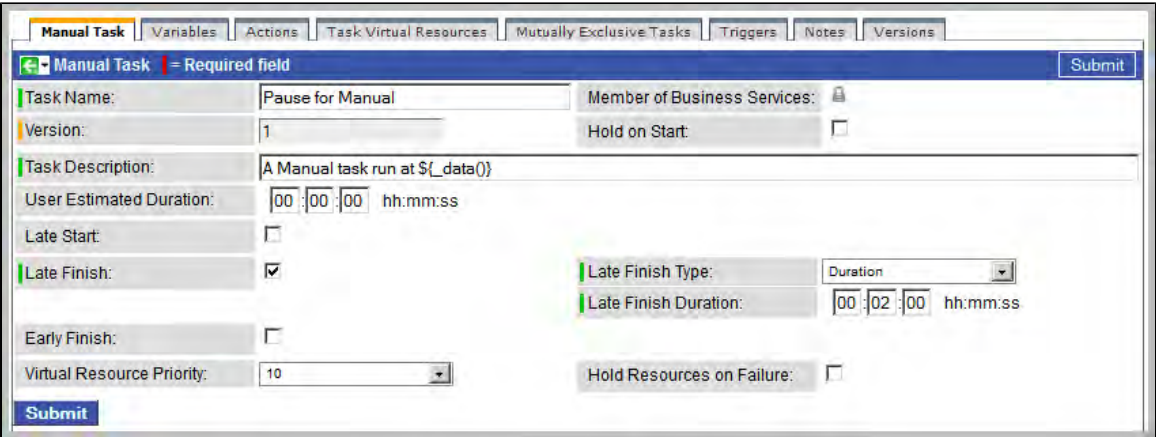
Overview

Manual tasks are used to create a pause in the workflow during which the user must take some action. The processing on a Manual task within a workflow is described below.

Step 1	While its upstream tasks are running, the Manual task remains in WAITING status. When the Manual task launches, it goes immediately into ACTION REQUIRED status, meaning a user must perform some manual activity. Opwise also sets the Started Time in the Manual task instance to the time the task goes into the ACTION REQUIRED status.
Step 2	<p>Optionally, the user can re-set the Started Time on the Manual task by issuing the Set Started command as follows:</p> <ol style="list-style-type: none"> 1. From the Activity display or the Workflow monitor, right click on the Manual task. 2. Select Set Started or, if you are using the Workflow monitor, select Commands > Set Started.  <p>Opwise re-sets the Started Time.</p>
Step 3	<p>When you have completed the activities called for in the Manual task, you need to indicate that the task is completed and that the workflow can continue, as follows:</p> <ol style="list-style-type: none"> 1. From the Activity display or the Workflow monitor, right click on the Manual task. 2. Select Set Completed or select Commands > Set Completed if you are using the Workflow monitor. The Manual task goes into SUCCESS status, the End Time is set, and the workflow continues. 3. If the Manual task is not completed but you want the workflow to continue anyway, select Force Finish.

You can also set a Manual task to STARTED or COMPLETED status from the [Command Line Interface \(CLI\)](#).

Creating a New Manual Task

Step 1	From the navigation pane, select Automation Center > Tasks > Manual Tasks . The Manual Tasks List screen displays.
Step 2	Click New . The Manual Task Definition screen displays. 
Step 3	Using the field descriptions provided below as a guide, complete the fields as needed.
Step 4	Click the Submit button to save the record and return to the menu, or right-click on the title bar and select Save to save the record and remain on the current display.
Step 5	If appropriate, repeat these steps for any additional tasks you want to add.

Task run statistics will show at the bottom after the first time this task has been launched.

Manual Task Field Descriptions

The table below describes the fields, buttons, and tabs on the task definition and task instance screens. Color coding is provided that differentiates the following three types of fields:

- Fields that display on the task definition and task instance screens are shown in black.
- Fields that display only on the task definition screen are shown in green.
- Fields that display only on the task instance screen are shown in maroon.

Field Name	Description
Task/Instance Name	Required. Name used within Opwise to identify this task. Up to 40 alphanumeric; variables supported. It is the user's responsibility to develop a workable naming scheme for tasks.
Invoked by	Task instance only; system-supplied. How the task instance was launched. One of the following: <ul style="list-style-type: none"> • Trigger: (Trigger Name) - The instance was launched by the named trigger. • Workflow: (Workflow Name) - The instance was launched by the named workflow. • Manually Launched - The instance was launched by a user. To determine the name of the user: <ol style="list-style-type: none"> 1. From the Activity or Task Instances screen, click the task instance name to open the record. 2. The Execution User field identifies the user who launched the task instance.
Task	Task instance only; system-supplied. Hover over the paper icon to display more information about the task instance. Click the paper icon to display the task definition record.
Instance Reference Id	Task instance only; system-supplied. Opwise increments this number each time the task is run.
Member of Business Services	User-defined. Allows you to select one or more Business Services that this record definition belongs to. Click the lock icon to unlock the field and select Business Services.

Version	Task definition only; system-supplied. The version number of the current record, which is incremented by Opwise every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Task Description	User-supplied description of this record.
Status	Task instance only; system-supplied. See Task Instance Statuses .
Exit Code	Task instance only; system-supplied. The exit code captured by the agent when executing the task (for example, a command or script).
Status Description	Task instance only; system-supplied. Provides additional information, if any, about the status of the task.
Start Time	Task instance only; system-supplied. The date and time the task started.
Duration	Task instance only; system-supplied. The amount of time the task took to run.
End Time	Task instance only; system-supplied. The date and time the task instance completed.
User Estimated End Time	Task instance only; system-supplied. If the user entered information into the User Estimated Duration field in the task definition, Opwise uses this information to calculate an end time for the task instance, based on the date/time the task instance started.
Shortest Estimated End Time	Task instance only; system-supplied.
Average Estimated End Time	Task instance only; system-supplied.
Longest Estimated End Time	Task instance only; system-supplied.
Hold on Start	If enabled, when the task is launched it appears in the Activity display with a status of Held. The task runs when the user releases it.
Hold Reason	Information about why the task will be put on hold when it starts.
User Estimated Duration	Task definition only; optional. The estimated amount of time it should normally take to run this task. Opwise uses this information to calculate the User Estimated End Time on a task instance record.
Late Start	If enabled, and if the task instance starts after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late start (see Late Start Type). To determine whether a task instance started late, open the task instance and locate the Started Late field; the field is checked if the instance started after the specified time. This field only appears on the task instance if the user specified a Late Start in the task definition.

Late Start Type	<p>Required if Late Start is enabled. Options are:</p> <ul style="list-style-type: none"> • Time - Flag the task if it starts after the specified time. • Duration - Flag the task if it starts a certain amount of time after the programmed start time. The task must have a specific start time.
Late Start Time	Time after which the task start time is considered late. Use hh:mm, 24-hour time
Late Start Duration	<p>Duration (amount of relative time) after which the task is considered to have started late. For a task within a workflow, the duration is the period between the time the workflow starts and the time the task itself starts. For example, a task might have a Late Start Duration of 60 minutes. If the workflow starts at 9:00 a.m. but the task itself does not start until 10:30, the task has started late.</p> <p>For a task that is not within a workflow, Late Start Duration has meaning only if the task has been held upon starting. For example, if a task has a Late Start Duration of 60 minutes and the Hold on Start field is enabled, if the task is not released from hold within the amount of time specified in the Late Start Duration field, the task has started late.</p>
Started Late	Task instance only; system-supplied. This field is flagged if the task started later than the time specified in the Late Start fields.
Late Finish	If enabled, and if the task instance finishes after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late finish (see Late Finish Type). To determine whether a task instance finished late, open the task instance and locate the Finished Late field; the field is checked if the instance finished after the specified time or lasted longer than expected. This field only appears on the task instance if the user specified a Late Finish in the task definition.
Late Finish Type	<p>Required if Late Finish is enabled. Options are:</p> <ul style="list-style-type: none"> • Time - Flag the task if it finishes after the specified time (see Late Finish Time). • Duration - Flag the task if it finishes a certain amount of time after the programmed finish time (see Late Finish Duration). The task must have a specific finish time.
Late Finish Time	If Late Finish Type is Time, use this to specify the time after which the task finish time is considered late. Use hh:mm, 24-hour time.
Late Finish Duration	If Late Finish Type is Duration, use this to specify the longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Finished Late	Task instance only; system-supplied. This field is flagged if the task finished later than the time or duration specified in the Late Finish fields.
Early Finish	If enabled, and if the task instance finishes before the time or period specified, the task instance is flagged as early. You can specify a time or duration to determine an early finish (see Early Finish Type). To determine whether a task instance finished early, open the task instance and locate the Finished Early field; the field is checked if the instance finished before the specified time or did not last as long as expected. This field only appears on the task instance if the user added Early Finish specifications to the task definition.
Early Finish Type	<p>Required if Early Finish is enabled. Options are:</p> <ul style="list-style-type: none"> • Time - Flag the task if it finishes before the specified time (see Early Finish Time). • Duration - Flag the task if it finishes a certain amount of time before the programmed finish time (see Early Finish Duration). The task must have a specific finish time.

Early Finish Time	If Early Finish Type is Time, use this to specify the time before which the task finish time is considered early. That is, enter a time at which the task should still be running. Use hh:mm, 24-hour time.
Early Finish Duration	If Early Finish Type is Duration, use this to specify the shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Finished Early	Task instance only; system-supplied. This field is flagged if the task finished earlier than the time specified in the Early Finish fields.
First Time Ran	Task definition only; system-supplied. Displays after the first time the task runs. The date and time this task first ran.
Lowest Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. The shortest amount of time this task has taken to run.
Last Time Ran	Task definition only; system-supplied. Displays after the first time the task runs. The most recent date and time the task ran.
Average Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. Shows the average amount of time this task takes to run.
Number of Instances	Task definition only; system-supplied. Displays after the first time the task runs. Shows the number of times this task has run.
Highest Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. The longest amount of time this task has taken to run.
Last Instance Duration	Task definition only; system-supplied. Displays after the first time the task runs. The amount of time the task took to run the last time it ran.
Virtual Resource Priority	Priority for acquiring a resource when two or more tasks are waiting for the resource. This priority applies to all resources required by the task. Options: 1 (high) - 20 (low). Default is 10.
Hold Resources on Failure	If enabled, the task instance will continue to hold Renewable resources if the task instance fails. Renewable resources will be returned only if the task instance status is either Complete, Finished, or Skipped.
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Launch Task button	Manually launches the task.
View Instances button	Displays a list of task instances for which there has been a status change or a modification to the task instance record within the last 30 days (an Updated on Last 30 Days filter has been pre-selected for the list).

Delete button	Deletes the current record.
View Parent button	Task instance only; Displays this task's parent task (workflow), if any.
Show Details button	Task instance only; displays detailed information about this task instance.
Retrieve Output button	Task instance only; see Retrieving Output .
Skip button	Task instance only; for tasks loaded into the schedule that haven't run yet. Allows you to tell Opwise to skip this task. See Skipping a Task .
Hold button	Task instance only; see Putting a Task on Hold .
Force Finish button	Task instance only; see Force Finishing a Task .
Re-run button	Task instance only; see Re-running a Task .
Output tab	Task instance only. Displays output generated from the process, if any, based on specifications provided by the user in the Automatic Output Retrieval fields in the task definition.
Variables tab	Displays all variables associated with this record.
Actions tab	<p>Allows you to specify actions that Opwise will take automatically based on events that occur during the execution of this task. Events are task instance status, exit codes, late start, late finish, and early finish. Actions are:</p> <ul style="list-style-type: none"> • Abort Action - Abort the task if certain events occur. For details, see Setting Abort Actions. • Email Notification - Send an email if certain events occur. For details, see Creating Email Notifications. • Set Variable - Used in workflows to set a variable based on the occurrence of certain events. For details, see Setting Variables within a Workflow. • SNMP Notification - Send an email if certain events occur. For details, see Creating SNMP Notifications. • System Operation - Run an Opwise system operation based on specified conditions. For details, see Setting System Operations.
Task Virtual Resources tab	Lists Virtual Resources to which this task is assigned.
Mutually Exclusive Tasks tab	Displays all tasks that have been set to be mutually exclusive of this task.
Triggers tab	Displays a list of all triggers that have been defined to launch this task. Also allows you to add new triggers. If you add a new trigger from this location, Opwise automatically constructs a default trigger name as follows: <current task name>#TRIGGER#. You can change the default name if desired. For instructions on creating triggers, see Creating Triggers .
Notes tab	Displays all notes associated with this task.
Versions Tab	Stores copies of all previous versions of the current record. See Record Versioning .

Specifying When a Task Runs

You can run the task as part of a [workflow](#), [specify triggers](#) that run the task automatically based on times or events, or [run the task manually](#).

Monitoring Task Execution

You can monitor all system activity from the [Activity screen](#) and can view activity history from the [Activity History screen](#).

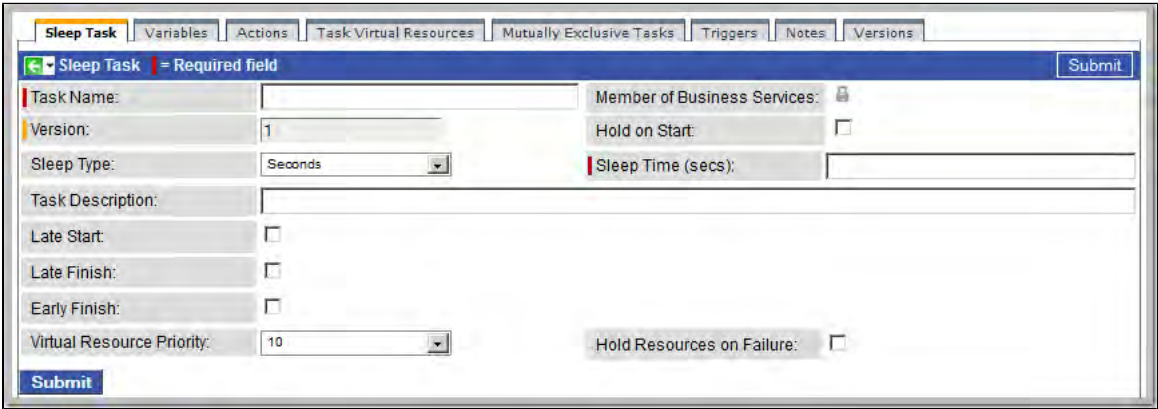
Sleep Task

- Overview
- Creating a New Sleep Task
- Sleep Task Field Descriptions
- Specifying When a Task Runs
- Monitoring Task Execution

Overview

The Sleep task allows you to execute a sleep command for a specified period of time or until a specific time. This task is helpful, for example, if you need to impose a pause in the processing of a workflow.

Creating a New Sleep Task

Step 1	From the navigation pane, select Automation Center > Tasks > Sleep Tasks . The Sleep Tasks List screen displays.
Step 2	Click New . The Sleep Task Definition screen displays.
	
Step 3	Using the field descriptions provided below as a guide, complete the fields as needed.
Step 4	Click the Submit button to save the record and return to the menu, or right-click on the title bar and select Save to save the record and remain on the current display.
Step 5	If appropriate, repeat these steps for any additional tasks you want to add.

Task run statistics appear after the first time this task has been launched.

Sleep Task Field Descriptions

The table below describes the fields, buttons, and tabs on the task definition and task instance screens. Color coding is provided that differentiates the following three types of fields:

- Fields that display on the task definition and task instance screens are shown in black.
- Fields that display only on the task definition screen are shown in green.
- Fields that display only on the task instance screen are shown in maroon.

Field Name	Description
Task/Instance Name	Required. Name used within Opwise to identify this task. Up to 40 alphanumeric; variables supported. It is the user's responsibility to develop a workable naming scheme for tasks.

Invoked by	<p>Task instance only; system-supplied. How the task instance was launched. One of the following:</p> <ul style="list-style-type: none"> • Trigger: (Trigger Name) - The instance was launched by the named trigger. • Workflow: (Workflow Name) - The instance was launched by the named workflow. • Manually Launched - The instance was launched by a user. To determine the name of the user: <ol style="list-style-type: none"> 1. From the Activity or Task Instances screen, click the task instance name to open the record. 2. The Execution User field identifies the user who launched the task instance.
Task	<p>Task instance only; system-supplied. Hover over the paper icon to display more information about the task instance. Click the paper icon to display the task definition record.</p>
Instance Reference Id	<p>Task instance only; system-supplied. Opwise increments this number each time the task is run.</p>
Member of Business Services	<p>User-defined. Allows you to select one or more Business Services that this record definition belongs to. Click the lock icon to unlock the field and select Business Services.</p>
Version	<p>Task definition only; system-supplied. The version number of the current record, which is incremented by Opwise every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning.</p>
Sleep Type	<p>User-supplied. The type of sleep command you want to execute. Options:</p> <ul style="list-style-type: none"> • Seconds - Use the Sleep Time (secs) field to specify the number of seconds. • Duration - Use the Sleep Duration field to specify the number of days, hours, minutes, and/or seconds. • Time - Use the Sleep Until Time (hh:mm) field to specify the time you want the sleep command to complete.
Sleep Time	<p>Required if Sleep Type = Seconds. Number of seconds the Sleep should last.</p>
Sleep Duration	<p>User-supplied. If Sleep Type = Duration, the number of hours, minutes and/or seconds the Sleep should last.</p>
Sleep Until Time	<p>Required if Sleep Type = Time. Time that the Sleep command should go to complete status. (Use 24-hour time.)</p>
Task Description	<p>User-supplied description of this record.</p>
Status	<p>Task instance only; system-supplied. See Task Instance Statuses.</p>
Exit Code	<p>Task instance only; system-supplied. The exit code captured by the agent when executing the task (for example, a command or script).</p>
Status Description	<p>Task instance only; system-supplied. Provides additional information, if any, about the status of the task.</p>
Start Time	<p>Task instance only; system-supplied. The date and time the task started.</p>
End Time	<p>Task instance only; system-supplied. The date and time the task instance completed.</p>
Duration	<p>Task instance only; system-supplied. The amount of time the task took to run.</p>

Hold on Start	If enabled, when the task is launched it appears in the Activity display with a status of Held. The task runs when the user releases it .
Hold Reason	Information about why the task will be put on hold when it starts.
Late Start	If enabled, and if the task instance starts after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late start (see Late Start Type). To determine whether a task instance started late, open the task instance and locate the Started Late field; the field is checked if the instance started after the specified time. This field only appears on the task instance if the user specified a Late Start in the task definition.
Late Start Type	Required if Late Start is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it starts after the specified time. • Duration - Flag the task if it starts a certain amount of time after the programmed start time. The task must have a specific start time.
Late Start Time	Time after which the task start time is considered late. Use hh:mm, 24-hour time
Late Start Duration	Duration (amount of relative time) after which the task is considered to have started late. For a task within a workflow, the duration is the period between the time the workflow starts and the time the task itself starts. For example, a task might have a Late Start Duration of 60 minutes. If the workflow starts at 9:00 a.m. but the task itself does not start until 10:30, the task has started late. For a task that is not within a workflow, Late Start Duration has meaning only if the task has been held upon starting. For example, if a task has a Late Start Duration of 60 minutes and the Hold on Start field is enabled, if the task is not released from hold within the amount of time specified in the Late Start Duration field, the task has started late.
Started Late	Task instance only; system-supplied. This field is flagged if the task started later than the time specified in the Late Start fields.
Late Finish	If enabled, and if the task instance finishes after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late finish (see Late Finish Type). To determine whether a task instance finished late, open the task instance and locate the Finished Late field; the field is checked if the instance finished after the specified time or lasted longer than expected. This field only appears on the task instance if the user specified a Late Finish in the task definition.
Late Finish Type	Required if Late Finish is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it finishes after the specified time (see Late Finish Time). • Duration - Flag the task if it finishes a certain amount of time after the programmed finish time (see Late Finish Duration). The task must have a specific finish time.
Late Finish Time	If Late Finish Type is Time, use this to specify the time after which the task finish time is considered late. Use hh:mm, 24-hour time.
Late Finish Duration	If Late Finish Type is Duration, use this to specify the longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Finished Late	Task instance only; system-supplied. This field is flagged if the task finished later than the time or duration specified in the Late Finish fields.

Early Finish	If enabled, and if the task instance finishes before the time or period specified, the task instance is flagged as early. You can specify a time or duration to determine an early finish (see Early Finish Type). To determine whether a task instance finished early, open the task instance and locate the Finished Early field; the field is checked if the instance finished before the specified time or did not last as long as expected. This field only appears on the task instance if the user added Early Finish specifications to the task definition.
Early Finish Type	Required if Early Finish is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it finishes before the specified time (see Early Finish Time). • Duration - Flag the task if it finishes a certain amount of time before the programmed finish time (see Early Finish Duration). The task must have a specific finish time.
Early Finish Time	If Early Finish Type is Time, use this to specify the time before which the task finish time is considered early. That is, enter a time at which the task should still be running. Use hh:mm, 24-hour time.
Early Finish Duration	If Early Finish Type is Duration, use this to specify the shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Finished Early	Task instance only; system-supplied. This field is flagged if the task finished earlier than the time specified in the Early Finish fields.
First Time Ran	Task definition only; system-supplied. Displays after the first time the task runs. The date and time this task first ran.
Last Time Ran	Task definition only; system-supplied. Displays after the first time the task runs. The most recent date and time the task ran.
Number of Instances	Task definition only; system-supplied. Displays after the first time the task runs. Shows the number of times this task has run.
Virtual Resource Priority	Priority for acquiring a resource when two or more tasks are waiting for the resource. This priority applies to all resources required by the task. Options: 1 (high) - 20 (low). Default is 10.
Hold Resources on Failure	If enabled, the task instance will continue to hold Renewable resources if the task instance fails. Renewable resources will be returned only if the task instance status is either Complete, Finished, or Skipped.
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Launch Task button	Manually launches the task.
View Instances button	Displays a list of task instances for which there has been a status change or a modification to the task instance record within the last 30 days (an Updated on Last 30 Days filter has been pre-selected for the list).

Delete button	Deletes the current record.
View Parent button	Task instance only; Displays this task's parent task (workflow), if any.
Show Details button	Task instance only; displays detailed information about this task instance.
Skip button	Task instance only; for tasks loaded into the schedule that haven't run yet. Allows you to tell Opwise to skip this task. See Skipping a Task .
Hold button	Task instance only; see Putting a Task on Hold .
Force Finish button	Task instance only; see Force Finishing a Task .
Re-run button	Task instance only; see Re-running a Task .
Output tab	Task instance only. Displays output generated from the process, if any, based on specifications provided by the user in the Automatic Output Retrieval fields in the task definition.
Variables tab	Displays all variables associated with this record.
Actions tab	<p>Allows you to specify actions that Opwise will take automatically based on events that occur during the execution of this task. Events are task instance status, exit codes, late start, late finish, and early finish. Actions are:</p> <ul style="list-style-type: none"> • Abort Action - Abort the task if certain events occur. For details, see Setting Abort Actions. • Email Notification - Send an email if certain events occur. For details, see Creating Email Notifications. • Set Variable - Used in workflows to set a variable based on the occurrence of certain events. For details, see Setting Variables within a Workflow. • SNMP Notification - Send an email if certain events occur. For details, see Creating SNMP Notifications. • System Operation - Run an Opwise system operation based on specified conditions. For details, see Setting System Operations.
Task Virtual Resources tab	Lists Virtual Resources to which this task is assigned.
Mutually Exclusive Tasks tab	Displays all tasks that have been set to be mutually exclusive of this task.
Triggers tab	Displays a list of all triggers that have been defined to launch this task. Also allows you to add new triggers. If you add a new trigger from this location, Opwise automatically constructs a default trigger name as follows: <current task name>#TRIGGER#. You can change the default name if desired. For instructions on creating triggers, see Creating Triggers .
Notes tab	Displays all notes associated with this task.
Versions Tab	Stores copies of all previous versions of the current record. See Record Versioning .

Specifying When a Task Runs

You can run the task as part of a [workflow](#), [specify triggers](#) that run the task automatically based on times or events, or [run the task manually](#).

Monitoring Task Execution

You can monitor all system activity from the [Activity screen](#) and can view activity history from the [Activity History screen](#).

SQL Task

- [Overview](#)
- [Built-In Variables](#)
- [Creating a New SQL Task](#)
- [SQL Task Field Descriptions](#)
- [Specifying When a Task Runs](#)
- [Monitoring Task Execution](#)

Overview

The SQL task allows you to execute one or a series of SQL statements against the database specified in the task.

**Note**

Before you can run a SQL task, you first must create a [Database Connection](#), which defines the information needed to locate and access the database.

Built-In Variables

The built-in variables outlined below can be used in a SQL task to pass data where appropriate:

- [Task and Task Instance Variables](#)
- [SQL and Stored Procedure Variables](#)

Creating a New SQL Task

Step 1 From the navigation pane, select **Automation Center > Tasks > SQL Tasks**. The SQL Tasks List screen displays.

Step 2 Click **New**. The SQL Task Definition screen displays.

Step 3 Using the field descriptions provided below as a guide, complete the fields as needed.

Step 4 Click the **Submit** button to save the record and return to the menu, or right-click on the title bar and select **Save** to save the record and remain on the current display.

Step 5 If appropriate, repeat these steps for any additional tasks you want to add.

Task run statistics appear after the first time this task has been launched.

SQL Task Field Descriptions

The table below describes the fields, buttons, and tabs on the task definition and task instance screens. Color coding is provided that differentiates the following three types of fields:

- Fields that display on the task definition and task instance screens are shown in black.
- Fields that display only on the task definition screen are shown in green.
- Fields that display only on the task instance screen are shown in maroon.

Field Name	Description
Task/Instance Name	Required. Name used within Opwise to identify this task. Up to 40 alphanumeric; variables supported. It is the user's responsibility to develop a workable naming scheme for tasks.
Member of Business Services	User-defined. Allows you to select one or more Business Services that this record definition belongs to. Click the lock icon to unlock the field and select Business Services.

Version	Task definition only; system-supplied. The version number of the current record, which is incremented by Opwise every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Invoked by	Task instance only; system-supplied. How the task instance was launched. One of the following: <ul style="list-style-type: none"> • Trigger: (Trigger Name) - The instance was launched by the named trigger. • Workflow: (Workflow Name) - The instance was launched by the named workflow. • Manually Launched - The instance was launched by a user. To determine the name of the user: <ol style="list-style-type: none"> 1. From the Activity or Task Instances screen, click the task instance name to open the record. 2. The Execution User field identifies the user who launched the task instance.
Task	Task instance only; system-supplied. Hover over the paper icon to display more information about the task instance. Click the paper icon to display the task definition record.
Execution User	Task instance only; system-supplied. If the task was launched manually, the ID of the user who launched it.
Instance Reference Id	Task instance only; system-supplied. Opwise increments this number each time the task is run.
Hold on Start	If enabled, when the task is launched it appears in the Activity display with a status of Held. The task runs when the user releases it.
Hold Reason	Information about why the task will be put on hold when it starts.
Database Connection	Required. Name of the Opwise Automation Center database connection that defines the database. Click the magnifying glass to browse for an existing database connection or add a new one.
Database Connection Variable	Optional. If enabled, the Database Connection field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: <code>\${variable name}</code> . The variable must be a supported type as described in Variables and Functions .
Credentials	Credentials under which an Agent runs this task. These credentials override any credentials provided in the Agent resource definition for any Agent running this task.
Credentials Variable	Optional. If enabled, the Credentials field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: <code>\${variable name}</code> . The variable must be a supported type as described in Variables and Functions .
Task Description	User-supplied description of this record.
Status	Task instance only; system-supplied. See Task Instance Statuses .
Exit Code	Task instance only; system-supplied. The vendor-specific exception code for the SQL exception.
Status Description	Task instance only; system-supplied. Provides additional information, if any, about the status of the task.
Start Time	Task instance only; system-supplied. The date and time the task started.

Duration	Task instance only; system-supplied. The amount of time the task took to run.
End Time	Task instance only; system-supplied. The date and time the task instance completed.
Rows Retrieved	Task instance only; system-supplied. The number of rows retrieved by the SQL procedure.
SQL State	Task instance only; system-supplied. Resolves to a return code that indicates the outcome of the most recently executed SQL statement.
SQL Error Message	Task instance only; system-supplied. Any error messages returned by the SQL procedure.
User Estimated Duration	Task definition only; optional. The estimated amount of time it should normally take to run this task. Opwise uses this information to calculate the User Estimated End Time on a task instance record.
SQL Command	Required. SQL command being executed against the database. Variables supported.
Result Processing	Specifies how Opwise Automation Center should determine whether the SQL command failed or completed successfully. Options: <ul style="list-style-type: none"> • Skip Result Processing • Success Exitcode Range - The SQL command is considered completed successfully if its exitcode falls within the range specified in the Exit Codes field (see below). • Failure Exitcode Range - The SQL command is considered failed if its exitcode falls within the range specified in the Exit Codes field (see below). • Success Result Set Contains - The SQL command is considered completed successfully depending on the value in a specific database column (see Column Name, Operator and Value fields below). • Failure Result Set Contains - The SQL command is considered failed depending on the value in a specific database column (see Column Name, Operator and Value fields below).
Exit Codes (Result Processing)	Required if Result Processing = Success Exitcode Range or Failure Exit Code Range. Specifies the range. Format: Numeric. Use commas to list a series of exit codes; use hyphens to specify a range. Example: 1,5, 22-30.
Column Name	Required if Result Processing = Success Result Set Contains or Failure Result Set Contains. Specifies the name of a database column that is being checked for a specific value.
Operator	Operator being used for the comparison. Options: =, !=, >, >=, <, <=, regex.
Value	Value being compared, using the operator specified.
Auto Cleanup	When data is retrieved as the result of a SQL task, the data is written into a database table. If Auto Cleanup is enabled, the data is discarded upon the successful completion of the task (or workflow if the task is contained within a workflow).
Maximum Rows	Optional. If necessary, specify a limit to the number of rows you want returned by the SQL statement. This value overrides any value you specify in the database connection .

Late Start	If enabled, and if the task instance starts after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late start (see Late Start Type). To determine whether a task instance started late, open the task instance and locate the Started Late field; the field is checked if the instance started after the specified time. This field only appears on the task instance if the user specified a Late Start in the task definition.
Late Start Type	Required if Late Start is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it starts after the specified time. • Duration - Flag the task if it starts a certain amount of time after the programmed start time. The task must have a specific start time.
Late Start Time	Time after which the task start time is considered late. Use hh:mm, 24-hour time
Late Start Duration	Duration (amount of relative time) after which the task is considered to have started late. For a task within a workflow, the duration is the period between the time the workflow starts and the time the task itself starts. For example, a task might have a Late Start Duration of 60 minutes. If the workflow starts at 9:00 a.m. but the task itself does not start until 10:30, the task has started late. For a task that is not within a workflow, Late Start Duration has meaning only if the task has been held upon starting. For example, if a task has a Late Start Duration of 60 minutes and the Hold on Start field is enabled, if the task is not released from hold within the amount of time specified in the Late Start Duration field, the task has started late.
Started Late	Task instance only; system-supplied. This field is flagged if the task started later than the time specified in the Late Start fields.
Late Finish	If enabled, and if the task instance finishes after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late finish (see Late Finish Type). To determine whether a task instance finished late, open the task instance and locate the Finished Late field; the field is checked if the instance finished after the specified time or lasted longer than expected. This field only appears on the task instance if the user specified a Late Finish in the task definition.
Late Finish Type	Required if Late Finish is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it finishes after the specified time (see Late Finish Time). • Duration - Flag the task if it finishes a certain amount of time after the programmed finish time (see Late Finish Duration). The task must have a specific finish time.
Late Finish Time	If Late Finish Type is Time, use this to specify the time after which the task finish time is considered late. Use hh:mm, 24-hour time.
Late Finish Duration	If Late Finish Type is Duration, use this to specify the longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Finished Late	Task instance only; system-supplied. This field is flagged if the task finished later than the time or duration specified in the Late Finish fields.
Early Finish	If enabled, and if the task instance finishes before the time or period specified, the task instance is flagged as early. You can specify a time or duration to determine an early finish (see Early Finish Type). To determine whether a task instance finished early, open the task instance and locate the Finished Early field; the field is checked if the instance finished before the specified time or did not last as long as expected. This field only appears on the task instance if the user added Early Finish specifications to the task definition.

Early Finish Type	Required if Early Finish is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it finishes before the specified time (see Early Finish Time). • Duration - Flag the task if it finishes a certain amount of time before the programmed finish time (see Early Finish Duration). The task must have a specific finish time.
Early Finish Time	If Early Finish Type is Time, use this to specify the time before which the task finish time is considered early. That is, enter a time at which the task should still be running. Use hh:mm, 24-hour time.
Early Finish Duration	If Early Finish Type is Duration, use this to specify the shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Finished Early	Task instance only; system-supplied. This field is flagged if the task finished earlier than the time specified in the Early Finish fields.
Maximum Retries	User-defined. The maximum number of times Opwise should retry this task after it has started and gone to a failed state.
Retry Indefinitely	User-defined. Enabled or disabled. Indicates whether Opwise should continue trying indefinitely to run this task. If you enable this field, it overrides any value placed in the Maximum Retries field (above).
Retry Interval	User-defined. The number of seconds between each retry.
Current Retry Count	Task instance only; system-supplied. Displays, only for a running task instance, the current number of times that Opwise has retried the task after it first went to failure status.
User Estimated End Time	Task instance only; system-supplied. If the user entered information into the User Estimated Duration field in the task definition, Opwise uses this information to calculate an end time for the task instance, based on the date/time the task instance started.
Shortest Estimated End Time	Task instance only; system-supplied.
Average Estimated End Time	Task instance only; system-supplied.
Longest Estimated End Time	Task instance only; system-supplied.
First Time Ran	Task definition only; system-supplied. Displays after the first time the task runs. The date and time this task first ran.
Lowest Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. The shortest amount of time this task has taken to run.
Last Time Ran	Task definition only; system-supplied. Displays after the first time the task runs. The most recent date and time the task ran.
Average Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. Shows the average amount of time this task takes to run.

Number of Instances	Task definition only; system-supplied. Displays after the first time the task runs. Shows the number of times this task has run.
Highest Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. The longest amount of time this task has taken to run.
Last Instance Duration	Task definition only; system-supplied. Displays after the first time the task runs. The amount of time the task took to run the last time it ran.
Virtual Resource Priority	Priority for acquiring a resource when two or more tasks are waiting for the resource. This priority applies to all resources required by the task. Options: 1 (high) - 20 (low). Default is 10.
Hold Resources on Failure	If enabled, the task instance will continue to hold Renewable resources if the task instance fails. Renewable resources will be returned only if the task instance status is either Complete, Finished, or Skipped.
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Launch Task button	Manually launches the task.
View Instances button	Displays a list of task instances for which there has been a status change or a modification to the task instance record within the last 30 days (an Updated on Last 30 Days filter has been pre-selected for the list).
Delete button	Deletes the current record.
View Parent button	Task instance only; Displays this task's parent task (workflow), if any.
Show Details button	Task instance only; displays detailed information about this task instance.
Retrieve Output button	Task instance only; see Retrieving Output .
Skip button	Task instance only; for tasks loaded into the schedule that haven't run yet. Allows you to tell Opwise to skip this task. See Skipping a Task .
Hold button	Task instance only; see Putting a Task on Hold .
Force Finish button	Task instance only; see Force Finishing a Task .

Re-run button	Task instance only; see Re-running a Task .
SQL Result Set tab	Task instance only. Stores results of executed SQL statements, if any.
SQL Warning Set tab	Task instance only. Contains warnings returned by executed SQL statements, if any.
Variables tab	Displays all variables associated with this record.
Actions tab	<p>Allows you to specify actions that Opwise will take automatically based on events that occur during the execution of this task. Events are task instance status, exit codes, late start, late finish, and early finish. Actions are:</p> <ul style="list-style-type: none"> • Abort Action - Abort the task if certain events occur. For details, see Setting Abort Actions. • Email Notification - Send an email if certain events occur. For details, see Creating Email Notifications. • Set Variable - Used in workflows to set a variable based on the occurrence of certain events. For details, see Setting Variables within a Workflow. • SNMP Notification - Send an email if certain events occur. For details, see Creating SNMP Notifications. • System Operation - Run an Opwise system operation based on specified conditions. For details, see Setting System Operations.
Task Virtual Resources tab	Lists Virtual Resources to which this task is assigned.
Mutually Exclusive Tasks tab	Displays all tasks that have been set to be mutually exclusive of this task.
Triggers tab	Displays a list of all triggers that have been defined to launch this task. Also allows you to add new triggers. If you add a new trigger from this location, Opwise automatically constructs a default trigger name as follows: <current task name>#TRIGGER#. You can change the default name if desired. For instructions on creating triggers, see Creating Triggers .
Notes tab	Displays all notes associated with this task.
Versions Tab	Stores copies of all previous versions of the current record. See Record Versioning .

Specifying When a Task Runs

You can run the task as part of a [workflow](#), [specify triggers](#) that run the task automatically based on times or events, or [run the task manually](#).

Monitoring Task Execution

You can monitor all system activity from the [Activity screen](#) and can view activity history from the [Activity History screen](#).

Stored Procedure Task

- Overview
- Built-In Variables
- Creating a New Stored Procedure Task
- Stored Procedure Task Field Descriptions
- Adding Stored Procedure Parameters
 - Adding a Parameter
 - Stored Procedure Parameter Field Descriptions
 - Deleting a Parameter
 - Viewing a Parameter
- Specifying When a Task Runs
- Monitoring Task Execution

Overview

A Stored Procedure task allows you to execute a stored procedure against the database specified in the task.

**Note**

Before you can run a Stored Procedure task, you first must create a [Database Connection](#), which defines the information needed to locate and access the database.

Built-In Variables

The built-in variables outlined below can be used in a Stored Procedure task to pass data where appropriate:

- [Task and Task Instance Variables](#)
- [SQL and Stored Procedure Variables](#)

Creating a New Stored Procedure Task

Step 1 From the navigation pane, select **Automation Center > Tasks > Stored Procedure Tasks**. The Stored Procedure Tasks List screen displays.

Step 2 Click **New**. The Stored Procedure Task Definition screen displays.

Step 3 Using the field descriptions provided below as a guide, complete the fields as needed.

Step 4 Click the **Submit** button to save the record and return to the menu, or right-click on the title bar and select **Save** to save the record and remain on the current display.

Step 5 If appropriate, repeat these steps for any additional tasks you want to add.

Task run statistics appear after the first time this task has been launched.

Stored Procedure Task Field Descriptions

The table below describes the fields, buttons, and tabs on the task definition and task instance screens. Color coding is provided that differentiates the following three types of fields:

- Fields that display on the task definition and task instance screens are shown in black.
- Fields that display only on the task definition screen are shown in green.
- Fields that display only on the task instance screen are shown in maroon.

Field Name	Description
Task/Instance Name	Required. Name used within Opwise to identify this task. Up to 40 alphanumeric; variables supported. It is the user's responsibility to develop a workable naming scheme for tasks.
Invoked by	Task instance only; system-supplied. How the task instance was launched. One of the following: <ul style="list-style-type: none"> • Trigger: (Trigger Name) - The instance was launched by the named trigger. • Workflow: (Workflow Name) - The instance was launched by the named workflow. • Manually Launched - The instance was launched by a user. To determine the name of the user: <ol style="list-style-type: none"> 1. From the Activity or Task Instances screen, click the task instance name to open the record. 2. The Execution User field identifies the user who launched the task instance.
Task	Task instance only; system-supplied. Hover over the paper icon to display more information about the task instance. Click the paper icon to display the task definition record.
Execution User	Task instance only; system-supplied. If the task was launched manually, the ID of the user who launched it.

Instance Reference Id	Task instance only; system-supplied. Opwise increments this number each time the task is run.
Member of Business Services	User-defined. Allows you to select one or more Business Services that this record definition belongs to. Click the lock icon to unlock the field and select Business Services .
Version	Task definition only; system-supplied. The version number of the current record, which is incremented by Opwise every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Hold on Start	If enabled, when the task is launched it appears in the Activity display with a status of Held. The task runs when the user releases it.
Hold Reason	Information about why the task will be put on hold when it starts.
Database Connection	Required. Name of the Opwise Automation Center database connection that defines the database. Click the magnifying glass to browse for an existing database connection or add a new one.
Database Connection Variable	Optional. If enabled, the Database Connection field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: <code>#{variable name}</code> . The variable must be a supported type as described in Variables and Functions .
Credentials	Credentials under which an Agent runs this task. These credentials override any credentials provided in the Agent resource definition for any Agent running this task.
Credentials Variable	Optional. If enabled, the Credentials field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: <code>#{variable name}</code> . The variable must be a supported type as described in Variables and Functions .
Task Description	User-supplied description of this record.
Status	Task instance only; system-supplied. See Task Instance Statuses .
Exit Code	Task instance only; system-supplied. The vendor-specific exception code for the SQL exception.
Status Description	Task instance only; system-supplied. Provides additional information, if any, about the status of the task.
Start Time	Task instance only; system-supplied. The date and time the task started.
Duration	Task instance only; system-supplied. The amount of time the task took to run.
End Time	Task instance only; system-supplied. The date and time the task instance completed.
Rows Retrieved	Task instance only; system-supplied. The number of rows retrieved by the SQL procedure.

SQL State	Task instance only; system-supplied. Resolves to a return code that indicates the outcome of the most recently executed SQL statement.
SQL Error Message	Task instance only; system-supplied. Any error messages returned by the SQL procedure.
User Estimated Duration	Task definition only; optional. The estimated amount of time it should normally take to run this task. Opwise uses this information to calculate the User Estimated End Time on a task instance record.
Stored Procedure Name	Required. Name of the file containing the stored procedure being executed against the database. Variables supported.
Result Processing	Specifies how Opwise should determine whether the Stored Procedure failed or completed successfully. Options: <ul style="list-style-type: none"> • Skip Result Processing • Success Exitcode Range - The Stored Procedure is considered completed successfully if its exitcode falls within the range specified in the Exit Codes field (see below). • Failure Exitcode Range - The Stored Procedure is considered failed if its exitcode falls within the range specified in the Exit Codes field (see below). • Success Result Set Contains - The Stored Procedure is considered completed successfully depending on the value in a specific database column (see Column Name, Operator and Value fields below). • Failure Result Set Contains - The Stored Procedure is considered failed depending on the value in a specific database column (see Column Name, Operator and Value fields below). • Success Output Parameter - The Stored Procedure is considered completed successfully if its output parameter satisfies the condition specified in the associated Parameter Position, Operator, and Value fields. • Failure Output Parameter - The Stored Procedure is considered failed if its output parameter satisfies the condition specified in the associated Parameter Position, Operator, and Value fields.
Parameter Position	Position of this parameter within a list of parameters.
Operator	Operator being used for the comparison. Options: =, !=, >, >=, <, <=, regex.
Value	Value being compared, using the operator specified.
Exit Codes (Result Processing)	Required if Result Processing = Success Exitcode Range or Failure Exit Code Range. Specifies the range. Format: Numeric. Use commas to list a series of exit codes; use hyphens to specify a range. Example: 1,5, 22-30.
Column Name	Required if Result Processing = Success Result Set Contains or Failure Result Set Contains. Specifies the name of a database column that is being checked for a specific value.
Operator	Operator being used for the comparison. Options: =, !=, >, >=, <, <=, regex.
Value	Value being compared, using the operator specified.
Auto Cleanup	When data is retrieved as the result of a SQL task, the data is written into a database table. If Auto Cleanup is enabled, the data is discarded upon the successful completion of the task (or workflow if the task is contained within a workflow).
Maximum Rows	Optional. If necessary, specify a limit to the number of rows you want returned by the SQL statement. This value overrides any value you specify in the database connection.

Late Start	If enabled, and if the task instance starts after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late start (see Late Start Type). To determine whether a task instance started late, open the task instance and locate the Started Late field; the field is checked if the instance started after the specified time. This field only appears on the task instance if the user specified a Late Start in the task definition.
Late Start Type	Required if Late Start is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it starts after the specified time. • Duration - Flag the task if it starts a certain amount of time after the programmed start time. The task must have a specific start time.
Late Start Time	Time after which the task start time is considered late. Use hh:mm, 24-hour time
Late Start Duration	Duration (amount of relative time) after which the task is considered to have started late. For a task within a workflow, the duration is the period between the time the workflow starts and the time the task itself starts. For example, a task might have a Late Start Duration of 60 minutes. If the workflow starts at 9:00 a.m. but the task itself does not start until 10:30, the task has started late. For a task that is not within a workflow, Late Start Duration has meaning only if the task has been held upon starting. For example, if a task has a Late Start Duration of 60 minutes and the Hold on Start field is enabled, if the task is not released from hold within the amount of time specified in the Late Start Duration field, the task has started late.
Started Late	Task instance only; system-supplied. This field is flagged if the task started later than the time specified in the Late Start fields.
Late Finish	If enabled, and if the task instance finishes after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late finish (see Late Finish Type). To determine whether a task instance finished late, open the task instance and locate the Finished Late field; the field is checked if the instance finished after the specified time or lasted longer than expected. This field only appears on the task instance if the user specified a Late Finish in the task definition.
Late Finish Type	Required if Late Finish is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it finishes after the specified time (see Late Finish Time). • Duration - Flag the task if it finishes a certain amount of time after the programmed finish time (see Late Finish Duration). The task must have a specific finish time.
Late Finish Time	If Late Finish Type is Time, use this to specify the time after which the task finish time is considered late. Use hh:mm, 24-hour time.
Late Finish Duration	If Late Finish Type is Duration, use this to specify the longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Finished Late	Task instance only; system-supplied. This field is flagged if the task finished later than the time or duration specified in the Late Finish fields.
Early Finish	If enabled, and if the task instance finishes before the time or period specified, the task instance is flagged as early. You can specify a time or duration to determine an early finish (see Early Finish Type). To determine whether a task instance finished early, open the task instance and locate the Finished Early field; the field is checked if the instance finished before the specified time or did not last as long as expected. This field only appears on the task instance if the user added Early Finish specifications to the task definition.

Early Finish Type	Required if Early Finish is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it finishes before the specified time (see Early Finish Time). • Duration - Flag the task if it finishes a certain amount of time before the programmed finish time (see Early Finish Duration). The task must have a specific finish time.
Early Finish Time	If Early Finish Type is Time, use this to specify the time before which the task finish time is considered early. That is, enter a time at which the task should still be running. Use hh:mm, 24-hour time.
Early Finish Duration	If Early Finish Type is Duration, use this to specify the shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Finished Early	Task instance only; system-supplied. This field is flagged if the task finished earlier than the time specified in the Early Finish fields.
Maximum Retries	User-defined. The maximum number of times Opwise should retry this task after it has started and gone to a failed state.
Retry Indefinitely	User-defined. Enabled or disabled. Indicates whether Opwise should continue trying indefinitely to run this task. If you enable this field, it overrides any value placed in the Maximum Retries field (above).
Retry Interval	User-defined. The number of seconds between each retry.
Current Retry Count	Task instance only; system-supplied. Displays, only for a running task instance, the current number of times that Opwise has retried the task after it first went to failure status.
User Estimated End Time	Task instance only; system-supplied. If the user entered information into the User Estimated Duration field in the task definition, Opwise uses this information to calculate an end time for the task instance, based on the date/time the task instance started.
Shortest Estimated End Time	Task instance only; system-supplied.
Average Estimated End Time	Task instance only; system-supplied.
Longest Estimated End Time	Task instance only; system-supplied.
First Time Ran	Task definition only; system-supplied. Displays after the first time the task runs. The date and time this task first ran.
Lowest Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. The shortest amount of time this task has taken to run.
Last Time Ran	Task definition only; system-supplied. Displays after the first time the task runs. The most recent date and time the task ran.
Average Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. Shows the average amount of time this task takes to run.

Number of Instances	Task definition only; system-supplied. Displays after the first time the task runs. Shows the number of times this task has run.
Highest Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. The longest amount of time this task has taken to run.
Last Instance Duration	Task definition only; system-supplied. Displays after the first time the task runs. The amount of time the task took to run the last time it ran.
Virtual Resource Priority	Priority for acquiring a resource when two or more tasks are waiting for the resource. This priority applies to all resources required by the task. Options: 1 (high) - 20 (low). Default is 10.
Hold Resources on Failure	If enabled, the task instance will continue to hold Renewable resources if the task instance fails. Renewable resources will be returned only if the task instance status is either Complete, Finished, or Skipped.
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Launch Task button	Manually launches the task.
View Instances button	Displays a list of task instances for which there has been a status change or a modification to the task instance record within the last 30 days (an Updated on Last 30 Days filter has been pre-selected for the list).
Delete button	Deletes the current record.
View Parent button	Task instance only; Displays this task's parent task (workflow), if any.
Show Details button	Task instance only; displays detailed information about this task instance.
Retrieve Output button	Task instance only; see Retrieving Output .
Skip button	Task instance only; for tasks loaded into the schedule that haven't run yet. Allows you to tell Opwise to skip this task. See Skipping a Task .
Hold button	Task instance only; see Putting a Task on Hold .
Force Finish button	Task instance only; see Force Finishing a Task .

Re-run button	Task instance only; see Re-running a Task .
Stored Procedure Parameters tab	See "Adding Stored Procedure Parameters" below.
SQL Result Set tab	Task instance only. Stores results of executed SQL statements, if any.
SQL Warning Set tab	Task instance only. Contains warnings returned by executed SQL statements, if any.
Variables tab	Displays all variables associated with this record.
Actions tab	<p>Allows you to specify actions that Opwise will take automatically based on events that occur during the execution of this task. Events are task instance status, exit codes, late start, late finish, and early finish. Actions are:</p> <ul style="list-style-type: none"> • Abort Action - Abort the task if certain events occur. For details, see Setting Abort Actions. • Email Notification - Send an email if certain events occur. For details, see Creating Email Notifications. • Set Variable - Used in workflows to set a variable based on the occurrence of certain events. For details, see Setting Variables within a Workflow. • SNMP Notification - Send an email if certain events occur. For details, see Creating SNMP Notifications. • System Operation - Run an Opwise system operation based on specified conditions. For details, see Setting System Operations.
Task Virtual Resources tab	Lists Virtual Resources to which this task is assigned.
Mutually Exclusive Tasks tab	Displays all tasks that have been set to be mutually exclusive of this task.
Triggers tab	Displays a list of all triggers that have been defined to launch this task. Also allows you to add new triggers. If you add a new trigger from this location, Opwise automatically constructs a default trigger name as follows: <current task name>#TRIGGER#. You can change the default name if desired. For instructions on creating triggers, see Creating Triggers .
Notes tab	Displays all notes associated with this task.
Versions Tab	Stores copies of all previous versions of the current record. See Record Versioning .

Adding Stored Procedure Parameters

You can enter one or more parameters for each stored procedure, as described below.

Adding a Parameter

Step 1	Open the Stored Procedure task to which you want to add the parameter.
Step 2	Click on the Stored Procedure Parameters tab or scroll to the parameters section. Opwise displays a list of parameters, if any.
Step 3	Click the New button.

Step 4 Use the field descriptions provided below to fill in the screen. An example parameter entry is shown below:

Step 5 Click the **Submit** button.

Stored Procedure Parameter Field Descriptions

Field Name	Description
Parameter Position	The position of this parameter within a list of parameters.
Parameter Mode	Mode of this parameter. Options: <ul style="list-style-type: none"> • Input • Output • Input/Output
Parameter Type	Type of parameter. Options: <ul style="list-style-type: none"> • VARCHAR • SMALLINT • INTEGER • BIGINT • FLOAT • REAL • DOUBLE • NUMERIC • DECIMAL • DATE • TIME • TIMESTAMP • VARBINARY • BOOLEAN
Value is Null	The value for the parameter is a database NULL value; applies to the input part of a stored procedure parameter. That is, if a value in a database is undefined, it is NULL, which means it has no set value. An input value can be NULL and is represented by selecting "Value is Null".
Input Value	Input value of the parameter, if any.
Description	Description of this parameter.

Deleting a Parameter

To delete a single parameter, display the parameter and click the **Delete** button.

To delete one or more parameters:

Step 1	From the parameters list, click the box associated with the parameter or parameters you want to delete.
Step 2	From the Actions on selected rows menu, select Delete .

Viewing a Parameter

Step 1	From the parameters list, scroll to the parameter you want to read.
Step 2	Click the underlined field displayed in the leftmost column. Opswise displays the contents of the parameter.

Specifying When a Task Runs

You can run the task as part of a [workflow](#), [specify triggers](#) that run the task automatically based on times or events, or [run the task manually](#).

Monitoring Task Execution

You can monitor all system activity from the [Activity screen](#) and can view activity history from the [Activity History screen](#).

Email Task

- Overview
- Creating a New Email Task
- Email Task Field Descriptions
- Specifying When a Task Runs
- Monitoring Task Execution

Overview

The Email task allows you to create and send emails. In order to execute Email tasks, you first need to define an [Email Connection](#), which defines the server information needed to create and send emails.

Creating a New Email Task

Step 1 From the navigation pane, select **Automation Center > Tasks > Email Tasks**. The Email Tasks List screen displays.

Step 2 Click **New**. The Email Task Definition screen displays.

The screenshot shows the 'Email Task' definition form. At the top, there are tabs for 'Email Task', 'Variables', 'Actions', 'Task Virtual Resources', 'Mutually Exclusive Tasks', 'Triggers', 'Notes', and 'Versions'. The 'Email Task' tab is active, and the title bar says 'Email Task | = Required field' with a 'Submit' button on the right. The form fields are as follows:

- Task Name:** [Text input field]
- Version:** [Text input field with value '1']
- Email Template:** [Text input field with search icon]
- Hold on Start:** [Checkbox, currently unchecked]
- Task Description:** [Text area]
- Member of Business Services:** [Text input field with lock icon]
- Email Connection:** [Text input field with search icon]
- Reply-To:** [Text input field]
- To:** [Text area]
- Cc:** [Text area]
- Bcc:** [Text area]
- Subject:** [Text input field]
- Body:** [Text area]
- Late Start:** [Checkbox, currently unchecked]
- Late Finish:** [Checkbox, currently unchecked]
- Early Finish:** [Checkbox, currently unchecked]
- Virtual Resource Priority:** [Dropdown menu with value '10']
- Hold Resources on Failure:** [Checkbox, currently unchecked]

At the bottom left, there is a 'Submit' button.

Step 3 Using the field descriptions provided below as a guide, complete the fields as needed.

Step 4	Click the Submit button to save the record and return to the menu, or right-click on the title bar and select Save to save the record and remain on the current display.
Step 5	If appropriate, repeat these steps for any additional tasks you want to add.

Task run statistics show at the bottom appear after the first time this task has been launched.

Email Task Field Descriptions

The table below describes the fields, buttons, and tabs on the task definition and task instance screens. Color coding is provided that differentiates the following three types of fields:

- Fields that display on the task definition and task instance screens are shown in black.
- Fields that display only on the task definition screen are shown in green.
- Fields that display only on the task instance screen are shown in maroon.

Field Name	Description
Task/Instance Name	Required. Name used within Opswise to identify this task. Up to 40 alphanumeric; variables supported. It is the user's responsibility to develop a workable naming scheme for tasks.
Email Template	Optional. The name of the Email template defined using the Email template screen. The Email template allows you to specify standard recipients and text for outgoing emails. Type in a name, or click the magnifying glass to browse to an existing Email template or create a new one. You must specify either an Email template or Email connection, or both. If you specify both, the Email server specified in the Email Connection record overrides the server in the template.
Version	Task definition only; system-supplied. The version number of the current record, which is incremented by Opswise every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning.
Invoked by	Task instance only; system-supplied. How the task instance was launched. One of the following: <ul style="list-style-type: none"> • Trigger: (Trigger Name) - The instance was launched by the named trigger. • Workflow: (Workflow Name) - The instance was launched by the named workflow. • Manually Launched - The instance was launched by a user. To determine the name of the user: <ol style="list-style-type: none"> 1. From the Activity or Task Instances screen, click the task instance name to open the record. 2. The Execution User field identifies the user who launched the task instance.
Task	Task instance only; system-supplied. Hover over the paper icon to display more information about the task instance. Click the paper icon to display the task definition record.
Instance Reference Id	Task instance only; system-supplied. Opswise increments this number each time the task is run.
Description	User-supplied description of this record.
Status	Task instance only; system-supplied. See Task Instance Statuses.
Exit Code	Task instance only; system-supplied. The exit code captured by the agent when executing the task (for example, a command or script).
Status Description	Task instance only; system-supplied. Provides additional information, if any, about the status of the task.
Start Time	Task instance only; system-supplied. The date and time the task started.

Duration	Task instance only; system-supplied. The amount of time the task took to run.
End Time	Task instance only; system-supplied. The date and time the task instance completed.
Hold on Start	If enabled, when the task is launched it appears in the Activity display with a status of Held. The task runs when the user releases it.
Hold Reason	Information about why the task will be put on hold when it starts.
Member of Business Services	User-defined. Allows you to select one or more Business Services that this record definition belongs to. Click the lock icon to unlock the field and select Business Services.
Email Connection	Required. Name of the Email connection defined using the Email connection screen. The email connection specifies information about the email server. You can also specify the Email connection in the Email template (see above). You must specify an Email template and/or an Email connection. If you specify an Email template and an Email connection, the server selected in the Email connection overrides the server selected in the Email template. Type in a name, click the magnifying glass to browse for an existing Email server definition, or create a new one.
Reply-To	Required. Specifies the email address of the sender. Use commas to separate multiple recipients. Variables supported.
To	Required. Specifies the email address of the recipient. Use commas to separate multiple recipients. Variables supported.
CC	Optional. Specifies the email address of the party being sent a copy of the email, if any. Use commas to separate multiple recipients. Variables supported.
BCC	Optional. Specifies the email address of the party being sent a blind (hidden) copy of the email, if any. Use commas to separate multiple recipients. Variables supported.
Subject	Optional. Specifies the subject line of the email. Variables supported.
Body	Optional. Contains the text of the email message. Variables supported. If both the email template and the email task contain text in the body, the text is appended.
Late Start	If enabled, and if the task instance starts after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late start (see Late Start Type). To determine whether a task instance started late, open the task instance and locate the Started Late field; the field is checked if the instance started after the specified time. This field only appears on the task instance if the user specified a Late Start in the task definition.
Late Start Type	Required if Late Start is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it starts after the specified time. • Duration - Flag the task if it starts a certain amount of time after the programmed start time. The task must have a specific start time.
Late Start Time	Time after which the task start time is considered late. Use hh:mm, 24-hour time

Late Start Duration	<p>Duration (amount of relative time) after which the task is considered to have started late. For a task within a workflow, the duration is the period between the time the workflow starts and the time the task itself starts. For example, a task might have a Late Start Duration of 60 minutes. If the workflow starts at 9:00 a.m. but the task itself does not start until 10:30, the task has started late.</p> <p>For a task that is not within a workflow, Late Start Duration has meaning only if the task has been held upon starting. For example, if a task has a Late Start Duration of 60 minutes and the Hold on Start field is enabled, if the task is not released from hold within the amount of time specified in the Late Start Duration field, the task has started late.</p>
Started Late	Task instance only; system-supplied. This field is flagged if the task started later than the time specified in the Late Start fields.
Late Finish	<p>If enabled, and if the task instance finishes after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late finish (see Late Finish Type). To determine whether a task instance finished late, open the task instance and locate the Finished Late field; the field is checked if the instance finished after the specified time or lasted longer than expected. This field only appears on the task instance if the user specified a Late Finish in the task definition.</p>
Late Finish Type	<p>Required if Late Finish is enabled. Options are:</p> <ul style="list-style-type: none"> • Time - Flag the task if it finishes after the specified time (see Late Finish Time). • Duration - Flag the task if it finishes a certain amount of time after the programmed finish time (see Late Finish Duration). The task must have a specific finish time.
Late Finish Time	If Late Finish Type is Time, use this to specify the time after which the task finish time is considered late. Use hh:mm, 24-hour time.
Late Finish Duration	If Late Finish Type is Duration, use this to specify the longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Finished Late	Task instance only; system-supplied. This field is flagged if the task finished later than the time or duration specified in the Late Finish fields.
Early Finish	<p>If enabled, and if the task instance finishes before the time or period specified, the task instance is flagged as early. You can specify a time or duration to determine an early finish (see Early Finish Type). To determine whether a task instance finished early, open the task instance and locate the Finished Early field; the field is checked if the instance finished before the specified time or did not last as long as expected. This field only appears on the task instance if the user added Early Finish specifications to the task definition.</p>
Early Finish Type	<p>Required if Early Finish is enabled. Options are:</p> <ul style="list-style-type: none"> • Time - Flag the task if it finishes before the specified time (see Early Finish Time). • Duration - Flag the task if it finishes a certain amount of time before the programmed finish time (see Early Finish Duration). The task must have a specific finish time.
Early Finish Time	If Early Finish Type is Time, use this to specify the time before which the task finish time is considered early. That is, enter a time at which the task should still be running. Use hh:mm, 24-hour time.
Early Finish Duration	If Early Finish Type is Duration, use this to specify the shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Finished Early	Task instance only; system-supplied. This field is flagged if the task finished earlier than the time specified in the Early Finish fields.

First Time Ran	Task definition only; system-supplied. Displays after the first time the task runs. The date and time this task first ran.
Lowest Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. The shortest amount of time this task has taken to run.
Last Time Ran	Task definition only; system-supplied. Displays after the first time the task runs. The most recent date and time the task ran.
Average Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. Shows the average amount of time this task takes to run.
Number of Instances	Task definition only; system-supplied. Displays after the first time the task runs. Shows the number of times this task has run.
Highest Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. The longest amount of time this task has taken to run.
Last Instance Duration	Task definition only; system-supplied. Displays after the first time the task runs. The amount of time the task took to run the last time it ran.
User Estimated End Time	Task instance only; system-supplied. If the user entered information into the User Estimated Duration field in the task definition, Opwise uses this information to calculate an end time for the task instance, based on the date/time the task instance started.
Shortest Estimated End Time	Task instance only; system-supplied.
Average Estimated End Time	Task instance only; system-supplied.
Longest Estimated End Time	Task instance only; system-supplied.
Virtual Resource Priority	Priority for acquiring a resource when two or more tasks are waiting for the resource. This priority applies to all resources required by the task. Options: 1 (high) - 20 (low). Default is 10.
Hold Resources on Failure	If enabled, the task instance will continue to hold Renewable resources if the task instance fails. Renewable resources will be returned only if the task instance status is either Complete, Finished, or Skipped.
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Launch Task button	Manually launches the task.

View Instances button	Displays a list of task instances for which there has been a status change or a modification to the task instance record within the last 30 days (an Updated on Last 30 Days filter has been pre-selected for the list).
Delete button	Deletes the current record.
View Parent button	Task instance only; Displays this task's parent task (workflow), if any.
Show Details button	Task instance only; displays detailed information about this task instance.
Retrieve Output button	Task instance only; see Retrieving Output .
Skip button	Task instance only; for tasks loaded into the schedule that haven't run yet. Allows you to tell Opwise to skip this task. See Skipping a Task .
Hold button	Task instance only; see Putting a Task on Hold .
Force Finish button	Task instance only; see Force Finishing a Task .
Re-run button	Task instance only; see Re-running a Task .
Output tab	Task instance only. Displays output generated from the process, if any, based on specifications provided by the user in the Automatic Output Retrieval fields in the task definition.
Variables tab	Displays all variables associated with this record.
Actions tab	<p>Allows you to specify actions that Opwise will take automatically based on events that occur during the execution of this task. Events are task instance status, exit codes, late start, late finish, and early finish. Actions are:</p> <ul style="list-style-type: none"> • Abort Action - Abort the task if certain events occur. For details, see Setting Abort Actions. • Email Notification - Send an email if certain events occur. For details, see Creating Email Notifications. • Set Variable - Used in workflows to set a variable based on the occurrence of certain events. For details, see Setting Variables within a Workflow. • SNMP Notification - Send an email if certain events occur. For details, see Creating SNMP Notifications. • System Operation - Run an Opwise system operation based on specified conditions. For details, see Setting System Operations.
Task Virtual Resources tab	Lists Virtual Resources to which this task is assigned.
Mutually Exclusive Tasks tab	Displays all tasks that have been set to be mutually exclusive of this task.
Triggers tab	Displays a list of all triggers that have been defined to launch this task. Also allows you to add new triggers. If you add a new trigger from this location, Opwise automatically constructs a default trigger name as follows: <current task name>#TRIGGER#. You can change the default name if desired. For instructions on creating triggers, see Creating Triggers .

Notes tab	Displays all notes associated with this task.
Versions Tab	Stores copies of all previous versions of the current record. See Record Versioning .

Specifying When a Task Runs

You can run the task as part of a [workflow](#), [specify triggers](#) that run the task automatically based on times or events, or [run the task manually](#).

Monitoring Task Execution

You can monitor all system activity from the [Activity screen](#) and can view activity history from the [Activity History screen](#).

Task Monitor Task

- [Overview](#)
- [Built-In Variables](#)
- [Processing Flow for Task Monitors](#)
 - [Launching a Task Monitor Task Within a Workflow](#)
 - [Launching a Task Monitor Task Using a Task Monitor Trigger](#)
 - [Launching a Task Monitor Task Manually or Via Other Trigger](#)
- [Creating a New Task Monitor Task](#)
- [Task Monitor Field Descriptions](#)
- [Monitoring Task Execution](#)

Overview

The Task Monitor task monitors another task or tasks for one or more specific statuses. When setting up a Task Monitor task, you can monitor all tasks; a specific task; a task type, such as Windows tasks; or a group of tasks based on the name, such as all tasks whose name contains the string "DEV". You can also monitor for any combination of [task statuses](#). For example, you can monitor for all tasks with a status of RESOURCE WAIT or UNDELIVERABLE, all Windows tasks in a FAILED status, or all tasks whose name contains "REPORT" that have a status of SUCCESS. For Task Monitors within a workflow, you can also specify a Time Scope, or window of time, during which the event being monitored for must be satisfied.

Built-In Variables

The built-in variables outlined below can be used to pass data where appropriate:

- [Task and Task Instance Variables](#)
- [Task Monitor Variables](#)

Processing Flow for Task Monitors

The processing on a Task Monitor may differ depending on which of the following methods was used to launch it:

- Launched by a workflow
- Launched by a Task Monitor trigger
- Launched manually or by another trigger

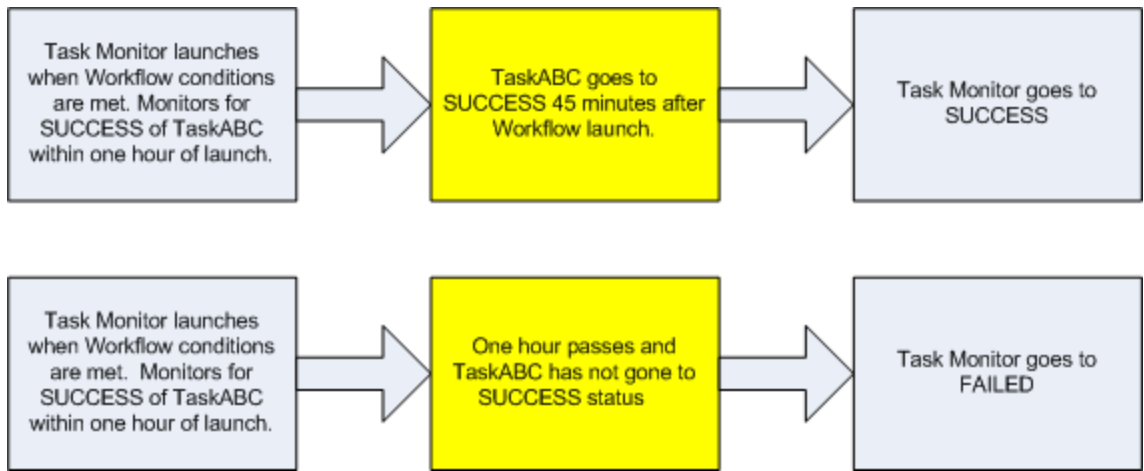
Each method is described in detail below.

Launching a Task Monitor Task Within a Workflow

Within a workflow, the Task Monitor task launches like any other task in the workflow, that is, whenever the workflow conditions warrant it. The Task Monitor runs until one of the conditions described below occurs:

- When the conditions specified in the Task Monitor are met, the Task Monitor goes to a status of SUCCESS.
- When the time window specified in the Task Monitor passes and the conditions in the Task Monitor are not met, the Task Monitor goes to a status of FAILED. If the time window is entirely in the past and Opwise does not locate the required event in the Activity table when the Task Monitor launches, the Task Monitor goes immediately to a FAILED status.
- If no time window is specified in the Task Monitor and the Task Monitor conditions are not met, the Task Monitor task continues running.
- A user can manually force finish the Task Monitor task.

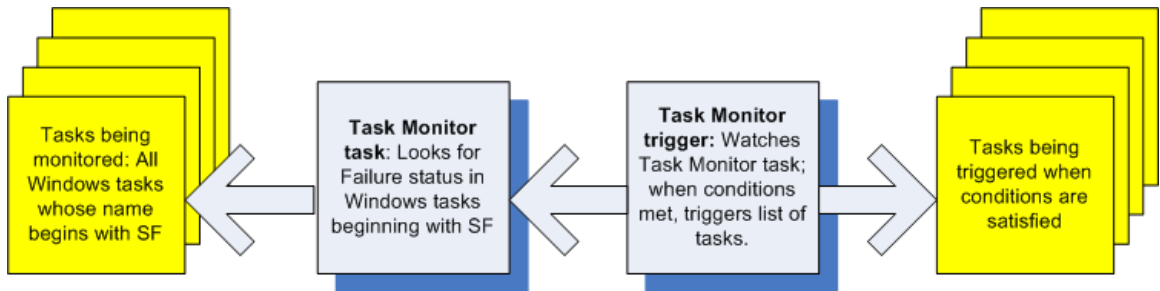
The following diagram illustrates how a Task Monitor might go to SUCCESS and FAILED status within a workflow.



Launching a Task Monitor Task Using a Task Monitor Trigger

The Task Monitor task launches when the user enables the Task Monitor trigger. Each time the conditions in the Task Monitor task are satisfied, the tasks specified in the trigger are launched. This process continues until a user disables the associated Task Monitor trigger.

The following diagram shows an example of how you might set up a task monitoring scheme using the Task Monitor task and Task Monitor trigger.



Launching a Task Monitor Task Manually or Via Other Trigger

If you manually launch a Task Monitor task or launch it using a trigger other than a Task Monitor trigger, such as a Time trigger, the task continues running until its specified conditions are met. When that occurs, the Task Monitor goes to SUCCESS. No other processing occurs unless you have configured notifications with the task or set up some other task(s) to launch based on the status of this task.

The Task Monitor runs until one of the conditions described below occurs:

- When the time window specified in the Task Monitor passes and the conditions in the Task Monitor are not met, the Task Monitor goes to a status of FAILED. If the time window is entirely in the past and Opswise does not locate the required event in the Activity table when the Task Monitor launches, the Task Monitor goes immediately to a FAILED status.
- If no time window is specified in the Task Monitor and the Task Monitor conditions are not met, the Task Monitor task continues running.

Creating a New Task Monitor Task

Step 1 From the navigation pane, select **Automation Center > Tasks > Task Monitors**. The Task Monitors List screen displays.

Step 2 Click **New**. The Task Monitor Definition screen displays.

Step 3 Using the field descriptions provided below as a guide, complete the fields as needed.

Step 4 Click the **Submit** button to save the record and return to the menu, or right-click on the title bar and select **Save** to save the record and remain on the current display.

Step 5 If appropriate, repeat these steps for any additional tasks you want to add.

Task run statistics appear after the first time this task has been launched.

Task Monitor Field Descriptions

The table below describes the fields, buttons, and tabs on the task definition and task instance screens. Color coding is provided that differentiates the following three types of fields:

- Fields that display on the task definition and task instance screens are shown in black.
- Fields that display only on the task definition screen are shown in green.
- Fields that display only on the task instance screen are shown in maroon.

Field Name	Description
Task/Instance Name	Required. Name used within Opwise to identify this task. Up to 40 alphanumeric; variables supported. It is the user's responsibility to develop a workable naming scheme for tasks.
Invoked by	<p>Task instance only; system-supplied. How the task instance was launched. One of the following:</p> <ul style="list-style-type: none"> • Trigger: (Trigger Name) - The instance was launched by the named trigger. • Workflow: (Workflow Name) - The instance was launched by the named workflow. • Manually Launched - The instance was launched by a user. To determine the name of the user: <ol style="list-style-type: none"> 1. From the Activity or Task Instances screen, click the task instance name to open the record. 2. The Execution User field identifies the user who launched the task instance.

Task	Task instance only; system-supplied. Hover over the paper icon to display more information about the task instance. Click the paper icon to display the task definition record.
Instance Reference Id	Task instance only; system-supplied. Opwise increments this number each time the task is run.
Member of Business Services	User-defined. Allows you to select one or more Business Services that this record definition belongs to. Click the lock icon to unlock the field and select Business Services .
Version	Task definition only; system-supplied. The version number of the current record, which is incremented by Opwise every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Description	User-supplied description of this record.
Status	Status being monitored for. When the task being monitored goes to a status specified in this field, the associated trigger is satisfied and the tasks specified in the trigger launch. You can specify status only, or status and exit code. You can specify as many statuses as needed (see Task Statuses).
Monitoring Type	Required. Specifies which task or tasks are being monitored. Options: <ul style="list-style-type: none"> • Specific Task - One or more specific tasks are being monitored. Use the Task to Monitor field to specify the task names. • General Tasks - Allows you to specify selection parameters that determine which task or tasks to be monitored. Use the Task Name To Monitor Condition and Task Type To Monitor field to create your selection parameters.
Task to Monitor	If Monitoring Type = Specific Task, this field specifies one or more tasks to monitor. Type in a task name or click the magnifying glass to browse for an existing task or to add a new task. To display details about the task specified in this field, hover over the paper icon.
Task Name to Monitor Condition	If Monitoring Type = General Task(s), this field allows you to specify selection parameters for which tasks to monitor. Options: <ul style="list-style-type: none"> • ALL TASKS - Specifies that the Task Monitor should monitor all tasks. • Starts With - Allows you to specify a string that one or more task names start with. Use the Task Name To Monitor Value field to enter the string. • Contains - Allows you to specify a string that one or more task names contain. Use the Task Name To Monitor Value field to enter the string. • Ends With - Allows you to specify a string that one or more task names end with. Use the Task Name To Monitor Value field to enter the string.
Task Name to Monitor Value	If the Task Name To Monitor Condition field = Starts With, Contains, or Ends With, use this field to specify the selection string. Up to 40 alphanumerics.
Task Type to Monitor	If you selected a Monitoring Type of General Tasks, this field allows you to define specific task types to monitor for. For example, to monitor all SQL tasks, you would select Monitoring Type = General Tasks, then select Task Type to Monitor = SQL Tasks.
Workflow Condition	With Workflow Condition Value, below, allows you to identify a workflow or workflows that contain the task being monitored for. If you specify these parameters, the task monitor conditions will only be considered met if the task appears within the specified workflow. Use this field to specify the condition. Options: <ul style="list-style-type: none"> • None • Equals • Starts With • Contains • Ends With

Workflow Condition Value	Required if a Workflow Condition is selected. With Workflow Condition, it allows you to identify a workflow or workflows that contain the task being monitored for. If you specify these parameters, the task monitor conditions will only be considered met if the task appears within the specified workflow. Use this field to specify the name or partial name of the workflow.
Time Scope	Used for Task Monitor tasks not associated with a Trigger. The Time Scope fields are used to create a window during which the Task Monitor conditions must be met in order for the Task Monitor to be satisfied. The Time Scope window is always relative to the time that the Task Monitor launched. For example, if you put -01:00 in the From time field and 02:00 in the To time field, the window's begin time is one hour before the Task Monitor is launched and its end time is two hours after it is launched. Note that the task being monitored must still be in the Activity screen in order for you to monitor for events that occurred in the past. If you specify a window that begins in the past, when the Task Monitor launches, it searches through the Activity table for the specified event. If it locates the event, the Task Monitor is satisfied immediately.
From [+/-]hh:mm	Required Time Scope = Relative. Used for Task Monitor tasks not associated with a Trigger. Together with the To [+/-]hh:mm : field, it allows you to specify a window of time, relative to the time the Task Monitor task launched, during which the conditions of the Task Monitor must be met. If the conditions are not met within the specified window, the Task Monitor task goes to a FAILED status. If you specify a past time in the this field, as soon as the Task Monitor task launches, Opwise searches the Activity table for past events that match the specified conditions. If the conditions are satisfied already, the Task Monitor task goes immediately to SUCCESS status. Otherwise, Opwise continues monitoring until the conditions are met or until the To [+/-]hh:mm : time has passed.
To [+/-]hh:mm	Used for Task Monitor tasks not associated with a Trigger. This field, together with the Time Scope From field, allows you to specify a window of time, relative to the time the Task Monitor task launched, during which the conditions of the Task Monitor must be met. If the conditions are not met within the specified window, the Task Monitor task goes to a FAILED status. If the conditions in the Task Monitor task are met before the Time Scope To time arrives, the Task Monitor task goes to SUCCESS. If the conditions are not met by the Time Scope To time, the Task Monitor task goes to FAILED status.
Status	Task instance only; system-supplied. Status of this task instance (see Task Statuses).
Status Description	Task instance only; system-supplied. Provides additional information, if any, about the status of the task.
Start Time	Task instance only; system-supplied. The date and time the task started.
End Time	Task instance only; system-supplied. The date and time the task instance completed.
Hold on Start	If enabled, when the task is launched it appears in the Activity display with a status of Held. The task runs when the user releases it.
Hold Reason	Information about why the task will be put on hold when it starts.
Late Start	If enabled, and if the task instance starts after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late start (see Late Start Type). To determine whether a task instance started late, open the task instance and locate the Started Late field; the field is checked if the instance started after the specified time. This field only appears on the task instance if the user specified a Late Start in the task definition.
Late Start Type	Required if Late Start is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it starts after the specified time. • Duration - Flag the task if it starts a certain amount of time after the programmed start time. The task must have a specific start time.

Late Start Time	Time after which the task start time is considered late. Use hh:mm, 24-hour time
Late Start Duration	<p>Duration (amount of relative time) after which the task is considered to have started late. For a task within a workflow, the duration is the period between the time the workflow starts and the time the task itself starts. For example, a task might have a Late Start Duration of 60 minutes. If the workflow starts at 9:00 a.m. but the task itself does not start until 10:30, the task has started late.</p> <p>For a task that is not within a workflow, Late Start Duration has meaning only if the task has been held upon starting. For example, if a task has a Late Start Duration of 60 minutes and the Hold on Start field is enabled, if the task is not released from hold within the amount of time specified in the Late Start Duration field, the task has started late.</p>
Started Late	Task instance only; system-supplied. This field is flagged if the task started later than the time specified in the Late Start fields.
Late Finish	If enabled, and if the task instance finishes after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late finish (see Late Finish Type). To determine whether a task instance finished late, open the task instance and locate the Finished Late field; the field is checked if the instance finished after the specified time or lasted longer than expected. This field only appears on the task instance if the user specified a Late Finish in the task definition.
Late Finish Type	<p>Required if Late Finish is enabled. Options are:</p> <ul style="list-style-type: none"> • Time - Flag the task if it finishes after the specified time (see Late Finish Time). • Duration - Flag the task if it finishes a certain amount of time after the programmed finish time (see Late Finish Duration). The task must have a specific finish time.
Late Finish Time	If Late Finish Type is Time, use this to specify the time after which the task finish time is considered late. Use hh:mm, 24-hour time.
Late Finish Duration	If Late Finish Type is Duration, use this to specify the longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Finished Late	Task instance only; system-supplied. This field is flagged if the task finished later than the time or duration specified in the Late Finish fields.
Early Finish	If enabled, and if the task instance finishes before the time or period specified, the task instance is flagged as early. You can specify a time or duration to determine an early finish (see Early Finish Type). To determine whether a task instance finished early, open the task instance and locate the Finished Early field; the field is checked if the instance finished before the specified time or did not last as long as expected. This field only appears on the task instance if the user added Early Finish specifications to the task definition.
Early Finish Type	<p>Required if Early Finish is enabled. Options are:</p> <ul style="list-style-type: none"> • Time - Flag the task if it finishes before the specified time (see Early Finish Time). • Duration - Flag the task if it finishes a certain amount of time before the programmed finish time (see Early Finish Duration). The task must have a specific finish time.
Early Finish Time	If Early Finish Type is Time, use this to specify the time before which the task finish time is considered early. That is, enter a time at which the task should still be running. Use hh:mm, 24-hour time.
Early Finish Duration	If Early Finish Type is Duration, use this to specify the shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.

Finished Early	Task instance only; system-supplied. This field is flagged if the task finished earlier than the time specified in the Early Finish fields.
First Time Ran	Task definition only; system-supplied. Displays after the first time the task runs. The date and time this task first ran.
Last Time Ran	Task definition only; system-supplied. Displays after the first time the task runs. The most recent date and time the task ran.
Number of Instances	Task definition only; system-supplied. Displays after the first time the task runs. Shows the number of times this task has run.
Virtual Resource Priority	<p>Priority for acquiring a resource when two or more tasks are waiting for the resource. This priority applies to all resources required by the task.</p> <p>Options: 1 (high) - 20 (low).</p> <p>Default is 10.</p>
Hold Resources on Failure	If enabled, the task instance will continue to hold Renewable resources if the task instance fails. Renewable resources will be returned only if the task instance status is either Complete, Finished, or Skipped.
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Launch Task button	Manually launches the task.
View Instances button	Displays a list of task instances for which there has been a status change or a modification to the task instance record within the last 30 days (an Updated on Last 30 Days filter has been pre-selected for the list).
Delete button	Deletes the current record.
View Parent button	Task instance only; Displays this task's parent task (workflow), if any.
Show Details button	Task instance only; displays detailed information about this task instance.
Retrieve Output button	Task instance only; see Retrieving Output .
Skip button	Task instance only; for tasks loaded into the schedule that haven't run yet. Allows you to tell Opwise to skip this task. See Skipping a Task .
Hold button	Task instance only; see Putting a Task on Hold .
Force Finish button	Task instance only; see Force Finishing a Task .

Re-run button	Task instance only; see Re-running a Task .
Output tab	Task instance only. Displays output generated from the process, if any, based on specifications provided by the user in the Automatic Output Retrieval fields in the task definition.
Variables tab	Displays all variables associated with this record.
Actions tab	<p>Allows you to specify actions that Opwise will take automatically based on events that occur during the execution of this task. Events are task instance status, exit codes, late start, late finish, and early finish. Actions are:</p> <ul style="list-style-type: none"> • Abort Action - Abort the task if certain events occur. For details, see Setting Abort Actions. • Email Notification - Send an email if certain events occur. For details, see Creating Email Notifications. • Set Variable - Used in workflows to set a variable based on the occurrence of certain events. For details, see Setting Variables within a Workflow. • SNMP Notification - Send an email if certain events occur. For details, see Creating SNMP Notifications. • System Operation - Run an Opwise system operation based on specified conditions. For details, see Setting System Operations.
Task Virtual Resources tab	Lists Virtual Resources to which this task is assigned.
Mutually Exclusive Tasks tab	Displays all tasks that have been set to be mutually exclusive of this task.
Triggers tab	Displays a list of all triggers that have been defined to launch this task. Also allows you to add new triggers. If you add a new trigger from this location, Opwise automatically constructs a default trigger name as follows: <current task name>#TRIGGER#. You can change the default name if desired. For instructions on creating triggers, see Creating Triggers .
Notes tab	Displays all notes associated with this task.
Versions Tab	Stores copies of all previous versions of the current record. See Record Versioning .

Monitoring Task Execution

You can monitor all system activity from the [Activity screen](#) and can view activity history from the [History screen](#).

File Monitor Task

- [Overview](#)
 - [Launching a File Monitor Task Within a Workflow](#)
 - [Launching a File Monitor Task Using a File \(Monitor\) Trigger](#)
 - [Launching a File Monitor Task Manually or Via Other Trigger](#)
- [Built-In Variables](#)
- [Creating a New File Monitor Task](#)
- [File Monitor Field Descriptions](#)
- [Monitoring Task Execution](#)

Overview

The File Monitor task allows you to monitor a specific remote machine for the creation, deletion, change, existence, or non-existence of one or more files at a specific location. In order to run a File Monitor task, you need an Opwise agent (Windows, Linux, Unix, or z/OS) running on the machine where you are monitoring for the file.

File Monitor tasks are meant to be launched using a File (Monitor) trigger or within a workflow. However, there are no technical restrictions on how a File Monitor task can be launched. The processing may differ depending on which of the following methods was used to launch it:

- Launched by a workflow
- Launched by a File Monitor trigger
- Launched manually or by another trigger type

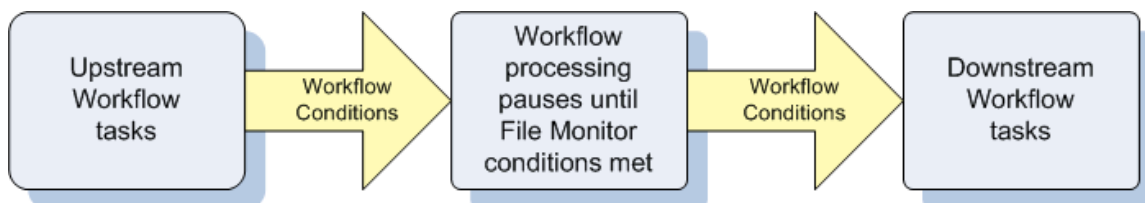
The processing on a File Monitor task for each launching method is described below.

Launching a File Monitor Task Within a Workflow

The File Monitor task can be launched within a [Workflow](#).

In this scenario, the task launches when the upstream workflow conditions are satisfied. Workflow processing then pauses until the conditions in the File Monitor task are satisfied. If the File Monitor is watching for the creation, change, or deletion of a file, the task goes to SUCCESS when the event occurs. If the File Monitor is watching for the existence or non-existence of a file, the task immediately goes to SUCCESS or FAILURE. Subsequent processing depends on the conditions built into the workflow.

The following diagram illustrates the processing for this scenario.



Launching a File Monitor Task Using a File (Monitor) Trigger

A common use for the File Monitor task is to launch it using a [File \(Monitor\) trigger](#), which specifies one or more tasks that are launched when the condition(s) is satisfied.

In this scenario, the File Monitor task launches when its associated File (Monitor) trigger is enabled.

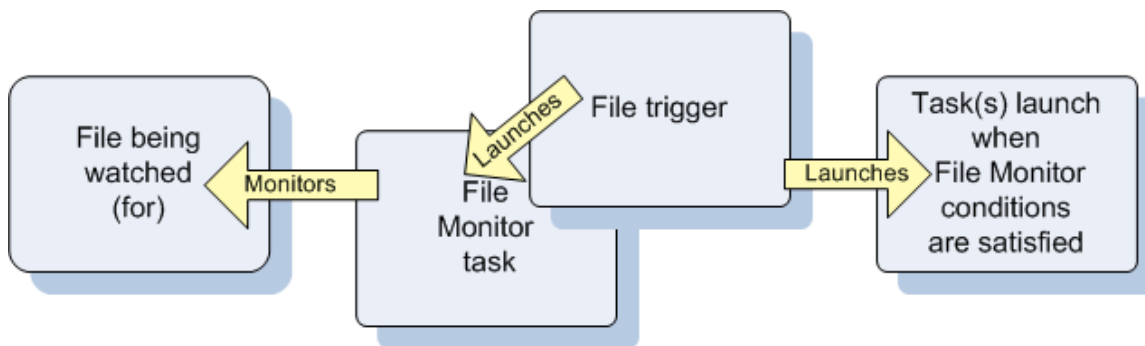
This method is best geared toward watching for the creation, deletion, or change in files. When the conditions in the File Monitor are satisfied, the File Monitor task goes to SUCCESS and the tasks listed in the associated trigger are launched. The File Monitor task continues running until its conditions are satisfied or until the user disables the trigger.

If you use this method to check for the existence or non-existence of a file, as soon as the task is launched it goes to SUCCESS or FINISHED status. If it goes to SUCCESS, the tasks specified in the trigger are launched. A FINISHED status indicates that it found a file that shouldn't be there or didn't find a file that should be there. Both of these cases constitute a "failure" of the conditions and therefore the tasks in the trigger are not launched.

When the File Monitor task goes to FINISHED or SUCCESS, the associated File (Monitor) trigger is automatically disabled.

When you launch a File Monitor task from a File trigger, you cannot manually cancel or force finish the task. You can only stop the task by disabling the trigger. If you manually disable the trigger while the task is still running, the task goes to FINISHED status.

The diagram below illustrates the processing flow for this scenario.



Launching a File Monitor Task Manually or Via Other Trigger

If you manually launch a File Monitor task or launch it using a non-File trigger, such as a Time trigger, the task continues running until its specified conditions are met, at which time the task goes to SUCCESS. No other processing occurs unless you have configured notifications with the task or set up some other task(s) to launch based on the status of this task.

If the conditions are not met, the task runs perpetually or until a user issues a Cancel or Force Finish command against it.

Built-In Variables

The built-in variables outlined below can be used to pass data where appropriate:

- [Task and Task Instance Variables](#)
- [File Monitor Variables](#).

Creating a New File Monitor Task

Step 1 From the navigation pane, select **Automation Center > Tasks > File Monitors**. The File Monitors List screen displays.

Step 2 Click **New**. The File Monitor Definition screen displays.

Step 3 Using the field descriptions provided below as a guide, complete the fields as needed.

Step 4	Click the Submit button to save the record and return to the menu, or right-click on the title bar and select Save to save the record and remain on the current display.
Step 5	If appropriate, repeat these steps for any additional tasks you want to add.

Task run statistics appear after the first time this task has been launched.

File Monitor Field Descriptions

The table below describes the fields, buttons, and tabs on the task definition and task instance screens. Color coding is provided that differentiates the following three types of fields:

- Fields that display on the task definition and task instance screens are shown in black.
- Fields that display only on the task definition screen are shown in green.
- Fields that display only on the task instance screen are shown in maroon.

Field Name	Description
Task/Instance Name	Required. Name used within Opwise to identify this task. Up to 40 alphanumeric; variables supported. It is the user's responsibility to develop a workable naming scheme for tasks.
Invoked by	Task instance only; system-supplied. How the task instance was launched. One of the following: <ul style="list-style-type: none"> • Trigger: (Trigger Name) - The instance was launched by the named trigger. • Workflow: (Workflow Name) - The instance was launched by the named workflow. • Manually Launched - The instance was launched by a user. To determine the name of the user: <ol style="list-style-type: none"> 1. From the Activity or Task Instances screen, click the task instance name to open the record. 2. The Execution User field identifies the user who launched the task instance.
Task	Task instance only; system-supplied. Hover over the paper icon to display more information about the task instance. Click the paper icon to display the task definition record.
Execution User	Task instance only; system-supplied. If the task was launched manually, the ID of the user who launched it.
Instance Reference Id	Task instance only; system-supplied. Opwise increments this number each time the task is run.
Credentials	Credentials under which an Agent runs this task. These credentials override any credentials provided in the Agent resource definition for any Agent running this task.
Version	Task definition only; system-supplied. The version number of the current record, which is incremented by Opwise every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning.
Credentials Variable	Optional. If enabled, the Credentials field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: <code>#{variable name}</code> . The variable must be a supported type as described in Variables and Functions .
Agent	Optional. The name of the agent resource definition that identifies the machine where the operation will run. If you do not specify an agent, you must specify an agent cluster (see below).
Agent Cluster	Optional. You can specify an agent cluster in addition to or in place of a specific agent. An agent cluster is a group of agents, one of which Opwise will choose to run this task. If you specify an agent and an agent cluster, Opwise Automation Center first tries to run the task on the specific agent. If the agent is not available, Opwise reverts to the agent cluster. See Agent Clusters for more information.

Agent Variable	Optional. If enabled, the Agent field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: \${variable name}. The variable must be a supported type as described in Variables and Functions .
Agent Cluster Variable	Optional. If enabled, the Agent Cluster field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: \${variable name}. The variable must be a supported type as described in Variables and Functions .
Cluster Broadcast	Task definition only; optional. You can specify a Cluster Broadcast in place of a specific agent and/or Agent Cluster. When you specify an agent cluster in the Cluster Broadcast field, Opwise Automation Center runs the task on all the agents in the cluster. Each instance of the task running on its own agent becomes a separate task instance record in the database and displays separately in the Activity monitor. See Agent Clusters for more information about defining agent clusters.
Task Description	User-supplied description of this record.
Status	Task instance only; system-supplied. See Task Instance Statuses .
Exit Code	Task instance only; system-supplied. The exit code captured by the agent when executing the task (for example, a command or script).
Status Description	Task instance only; system-supplied. Provides additional information, if any, about the status of the task.
Start Time	Task instance only; system-supplied. The date and time the task started.
Duration	Task instance only; system-supplied. The amount of time the task took to run.
End Time	Task instance only; system-supplied. The date and time the task instance completed.
Member of Business Services	User-defined. Allows you to select one or more Business Services that this record definition belongs to. Click the lock icon to unlock the field and select Business Services .
User Estimated Duration	Task definition only; optional. The estimated amount of time it should normally take to run this task. Opwise uses this information to calculate the User Estimated End Time on a task instance record.
Monitor Type	Type of file event being monitored for. Options: <ul style="list-style-type: none"> • Create - Wait for the creation of one or more files. • Delete - Wait for the deletion of one or more files. • Change - Monitor for a change in one or more files. [NOTE: not supported for z/OS.] • Exists - Check to see if one or more files already exists. • Missing - Check to see if one or more files is missing.
Monitor File(s)	Required. Location and name of a specific file or file pattern (for example, ACT001*) being monitored. Variables supported. Wildcards supported.
Recursive	If enabled, the monitor searches the specified directory and all subdirectories.

File Owner	User ID of owner of the file on the operating system. Specifying a file owner limits the search to files with that owner.
Maximum Files	For searches that use wildcards, limits the number of files to be searched.
Stable (seconds)	If Monitor Type = Change or Create: Period of time, in seconds, during which the file has not changed.
By percentage (+/-)	If Type = Change, the amount that the file size has changed, expressed as a percentage of the original file size. For example, enter 10 to monitor for a change in file size of 10 percent (larger or smaller).
By Size (+/-)	If Type = Change, used in conjunction with the By scale field, specifies an actual change in file size. For example, to monitor for a change in file size of 10 MB, enter 10 in this field and select MB in the By scale field.
By scale	If Type = Change, used in conjunction with the By Size field, specifies Bytes, KB (kilobytes), or MB (megabytes). For example, to monitor for a change in file size of 10 MB, enter 10 in the By Size field and select MB in this field.
To Size	If Type = Change, used in conjunction with the To scale field, specifies an actual file size that you want to monitor for. For example, to monitor for a file size of 5 KB, enter 5 in this field and select KB in the To scale field.
To scale	If Type = Change, used in conjunction with the To Size field, specifies an actual file size that you want to monitor for. For example, to monitor for a file size of 5 KB, enter 5 in the To Size field and select KB in this field.
Scan Text	Optional. If Type = Create, Change or Exists, this field specifies a string that the monitor will search for in the file or files. Specifying a string means that only files containing the string constitute a match. Opwise will process this field as a regular expression.
Scan Forward	Optional. If Type = Change. If enabled, this field specifies that once the File Monitor has been satisfied, it should continue from where it left off. If it is scanning within a file, it should resume from the point in the file that it last scanned. If it is monitoring for files, it should resume monitoring for the next file. If you are scanning a file that is being overwritten each time and you want to start from the beginning each time, you should disable Scan Forward.
User Estimated End Time	Task instance only; system-supplied. If the user entered information into the User Estimated Duration field in the task definition, Opwise uses this information to calculate an end time for the task instance, based on the date/time the task instance started.
Shortest Estimated End Time	Task instance only; system-supplied.
Average Estimated End Time	Task instance only; system-supplied.
Longest Estimated End Time	Task instance only; system-supplied.
First Time Ran	Task definition only; system-supplied. Displays after the first time the task runs. The date and time this task first ran.

Lowest Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. The shortest amount of time this task has taken to run.
Last Time Ran	Task definition only; system-supplied. Displays after the first time the task runs. The most recent date and time the task ran.
Average Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. Shows the average amount of time this task takes to run.
Number of Instances	Task definition only; system-supplied. Displays after the first time the task runs. Shows the number of times this task has run.
Highest Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. The longest amount of time this task has taken to run.
Last Instance Duration	Task definition only; system-supplied. Displays after the first time the task runs. The amount of time the task took to run the last time it ran.
Virtual Resource Priority	<p>Priority for acquiring a resource when two or more tasks are waiting for the resource. This priority applies to all resources required by the task.</p> <p>Options: 1 (high) - 20 (low).</p> <p>Default is 10.</p>
Hold Resources on Failure	If enabled, the task instance will continue to hold Renewable resources if the task instance fails. Renewable resources will be returned only if the task instance status is either Complete, Finished, or Skipped.
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Launch Task button	Manually launches the task.
View Instances button	Displays a list of task instances for which there has been a status change or a modification to the task instance record within the last 30 days (an Updated on Last 30 Days filter has been pre-selected for the list).
Delete button	Deletes the current record.
View Parent button	Task instance only; Displays this task's parent task (workflow), if any.
Show Details button	Task instance only; displays detailed information about this task instance.
Retrieve Output button	Task instance only; see Retrieving Output .

Skip button	Task instance only; for tasks loaded into the schedule that haven't run yet. Allows you to tell Opwise to skip this task. See Skipping a Task .
Hold button	Task instance only; see Putting a Task on Hold .
Force Finish button	Task instance only; see Force Finishing a Task .
Re-run button	Task instance only; see Re-running a Task .
Output tab	Task instance only. Displays output generated from the process, if any, based on specifications provided by the user in the Automatic Output Retrieval fields in the task definition.
Variables tab	Displays all variables associated with this record.
Actions tab	Allows you to specify actions that Opwise will take automatically based on events that occur during the execution of this task. Events are task instance status, exit codes, late start, late finish, and early finish. Actions are: <ul style="list-style-type: none"> • Abort Action - Abort the task if certain events occur. For details, see Setting Abort Actions. • Email Notification - Send an email if certain events occur. For details, see Creating Email Notifications. • Set Variable - Used in workflows to set a variable based on the occurrence of certain events. For details, see Setting Variables within a Workflow. • SNMP Notification - Send an email if certain events occur. For details, see Creating SNMP Notifications. • System Operation - Run an Opwise system operation based on specified conditions. For details, see Setting System Operations.
Task Virtual Resources tab	Lists Virtual Resources to which this task is assigned.
Mutually Exclusive Tasks tab	Displays all tasks that have been set to be mutually exclusive of this task.
File Monitor Triggers tab	Displays a list of all File Monitor triggers associated with this task. Enabling any of the triggers will launch this task. When the conditions in the task are satisfied, the tasks specified in the trigger will launch. For details, see Launching a File Monitor Task Using a File (Monitor) Trigger .
Notes tab	Displays all notes associated with this task.
Versions Tab	Stores copies of all previous versions of the current record. See Record Versioning .

Monitoring Task Execution

You can monitor all system activity from the [Activity screen](#) and can view activity history from the [Activity History screen](#).

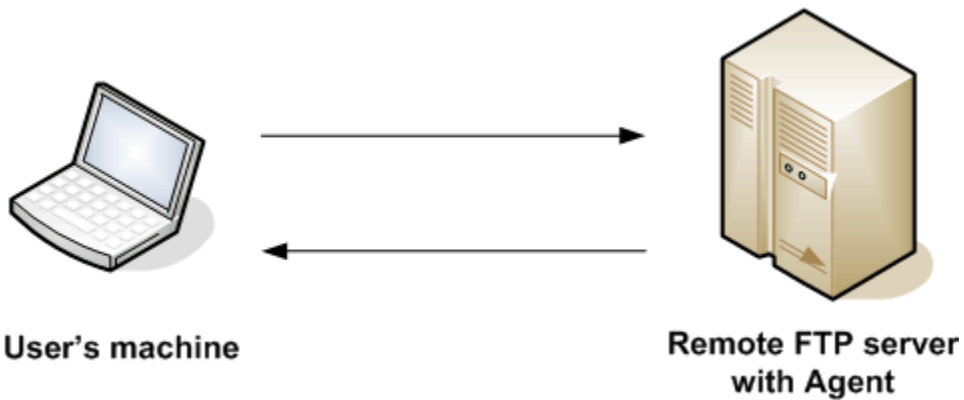
FTP File Monitor Task

- [Overview](#)
- [Built-In Variables](#)
- [Creating a New FTP File Monitor Task](#)
- [FTP File Monitor Task Field Descriptions](#)
- [Specifying When a Task Runs](#)
- [Monitoring Task Execution](#)

Overview

The FTP File Monitor task allows you to monitor for a file on a remote machine where an FTP server is running. The FTP File Monitor connects to the FTP server rather than the machine's file system to monitor for files. The FTP File Monitor can be used only within a workflow; you cannot run a FTP File Monitor task based on a trigger. To run an FTP File Monitor task, you need an Agent to communicate with the FTP server. The Agent can, but does not have to be, running on the same machine as the FTP server.

In the following example, the user wants to monitor for a file on a remote FTP Server that has an Agent running on it. In this case, the login credentials for the Agent machine and the FTP server machine are the same.



In the following example, the user wants to monitor for a file on a remote FTP Server that does not have an Agent running on it. In this case, the FTP File Monitor task definition provides an address and login credentials for the machine where the Agent is running as well as address and login credentials for the FTP server.



Built-In Variables

The built-in variables outlined below can be used in an FTP File Monitor task to pass data where appropriate:

- [Task and Task Instance Variables](#)
- [FTP File Monitor Variables.](#)

Creating a New FTP File Monitor Task

Step 1 From the navigation pane, select **Automation Center > Tasks > FTP File Monitors**. The FTP File Monitors List screen displays.

Step 2 Click **New**. The FTP File Monitor Task Definition screen displays.

Step 3 Using the field descriptions provided below as a guide, complete the fields as needed.

Step 4 Click the **Submit** button to save the record and return to the menu, or right-click on the title bar and select **Save** to save the record and remain on the current display.

Step 5 If appropriate, repeat these steps for any additional tasks you want to add.

Task run statistics appear after the first time this task has been launched.

FTP File Monitor Task Field Descriptions

The table below describes the fields, buttons, and tabs on the task definition and task instance screens. Color coding is provided that differentiates the following three types of fields:

- Fields that display on the task definition and task instance screens are shown in black.
- Fields that display only on the task definition screen are shown in green.
- Fields that display only on the task instance screen are shown in maroon.

Field Name	Description
Task/Instance Name	Required. Name used within Opswise to identify this task. Up to 40 alphanumeric; variables supported. It is the user's responsibility to develop a workable naming scheme for tasks.
Invoked by	Task instance only; system-supplied. How the task instance was launched. One of the following: <ul style="list-style-type: none"> • Trigger: (Trigger Name) - The instance was launched by the named trigger. • Workflow: (Workflow Name) - The instance was launched by the named workflow. • Manually Launched - The instance was launched by a user. To determine the name of the user: <ol style="list-style-type: none"> 1. From the Activity or Task Instances screen, click the task instance name to open the record. 2. The Execution User field identifies the user who launched the task instance.

Task	Task instance only; system-supplied. Hover over the paper icon to display more information about the task instance. Click the paper icon to display the task definition record.
Execution User	Task instance only; system-supplied. If the task was launched manually, the ID of the user who launched it.
Instance Reference ID	Task instance only; system-supplied. Opwise increments this number each time the task is run.
Credentials	Credentials under which an Agent runs this task. These credentials override any credentials provided in the Agent resource definition for any Agent running this task.
Version	Task definition only; system-supplied. The version number of the current record, which is incremented by Opwise every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Credentials Variable	Optional. If enabled, the Credentials field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: <code>\${variable name}</code> . The variable must be a supported type as described in Variables and Functions .
Agent	Optional. The name of the agent resource definition that identifies the machine where the operation will run. If you do not specify an agent, you must specify an agent cluster (see below).
Agent Cluster	Optional. You can specify an agent cluster in addition to or in place of a specific agent. An agent cluster is a group of agents, one of which Opwise will choose to run this task. If you specify an agent and an agent cluster, Opwise Automation Center first tries to run the task on the specific agent. If the agent is not available, Opwise reverts to the agent cluster. See Agent Clusters for more information.
Agent Variable	Optional. If enabled, the Agent field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: <code>\${variable name}</code> . The variable must be a supported type as described in Variables and Functions .
Agent Cluster Variable	Optional. If enabled, the Agent Cluster field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: <code>\${variable name}</code> . The variable must be a supported type as described in Variables and Functions .
Cluster Broadcast	Task definition only; optional. You can specify a Cluster Broadcast in place of a specific agent and/or Agent Cluster. When you specify an agent cluster in the Cluster Broadcast field, Opwise Automation Center runs the task on all the agents in the cluster. Each instance of the task running on its own agent becomes a separate task instance record in the database and displays separately in the Activity monitor. See Agent Clusters for more information about defining agent clusters.
Task Description	User-supplied description of this record.
Status	Task instance only; system-supplied. See Task Instance Statuses .
Status Description	Task instance only; system-supplied. Provides additional information, if any, about the status of the task.
Exit Code	Task instance only; system-supplied. The exit code captured by the agent when executing the task (for example, a command or script).

Start Time	Task instance only; system-supplied. The date and time the task started.
Duration	Task instance only; system-supplied. The amount of time the task took to run.
End Time	Task instance only; system-supplied. The date and time the task instance completed.
Hold on Start	If enabled, when the task is launched it appears in the Activity display with a status of Held. The task runs when the user releases it.
Member of Business Services	User-defined. Allows you to select one or more Business Services that this record definition belongs to. Click the lock icon to unlock the field and select Business Services .
User Estimated Duration	Task definition only; optional. The estimated amount of time it should normally take to run this task. Opwise uses this information to calculate the User Estimated End Time on a task instance record.
Monitor Type	Type of file event being monitored for. Options: <ul style="list-style-type: none"> • Exists - Checks to see if the file exists. • Missing - Checks to see if the file does not exist.
Wait until Satisfied	If enabled, the task instance starts and continues to run until one of the following events occurs: <ul style="list-style-type: none"> • If Monitor Type = Exists and the specified file exists or appears, the task instance completes with a status of SUCCESS. • If Monitor Type = Missing and the specified file does not exist or exists then disappears, the task instance completes with a status of SUCCESS. <p>If not enabled, the task instance:</p> <ol style="list-style-type: none"> 1. Starts. 2. Checks for the existence of the file. 3. Takes one of the following actions: <ul style="list-style-type: none"> • If Monitor Type = Exists and if the file exists, the task instance completes with a status of SUCCESS. • If Monitor Type = Exists and if the file does not exist, the task instance completes with a status of FAILURE. • If Monitor Type = Missing and if the file exists, the task instance completes with a status of FAILURE. • If Monitor Type = Missing and if the file does not exist, the task instance completes with a status of SUCCESS.
Poll Interval (Seconds)	If Wait until Satisfied is enabled: Frequency, in seconds, in which the FTP File Monitor will check to see if the file exists or is missing.
Maximum Polls	If Wait until Satisfied is enabled: Maximum number of times that the FTP File Monitor will check to see if the file exists or is missing.

Stable (Seconds)	<p>If Wait until Satisfied is enabled: Period of time, in seconds, during which the file has not changed.</p> <p>For an FTP/SFTP File Monitor task, a file's stability depends on its size. If the file size displayed in the FTP/SFTP output does not change during the specified number of seconds, the file is considered stable. In order for the task to reliably monitor the file's stability, the task must display a file's size in a well-known location. This means that the file list returned in the output must be in Unix long-listing format, as follows:</p> <pre data-bbox="360 331 1442 415">-rwxr-xr-x 1 owner group 12345 Jan 1 2012 somefile.txt</pre> <p>The task will only find the size if it is in the 5th column (for example, 12345 in the example above).</p> <p>The default file list format varies across different FTP client/server implementations, but most support additional commands that can force the output to the required format. The List Format Options field, below, is provided to insert those statements into the FTP script that the file monitor task executes.</p> <p>By default, if a value for Stable (Seconds) is specified, an FTP File Monitor task instance will verify that the Agent version is 5.1.0.16 or higher. If the Agent version is 5.1.0.15 or lower, the task instance will not run, the status will be set to Start Failure, and the following message will be logged: Stable (Seconds) option only supported on agent 5.1.0.16 or higher.</p>
Server Type	<p>Type of FTP server. Options:</p> <ul data-bbox="370 806 456 856" style="list-style-type: none">• FTP• SFTP

List Format Options	<p>If Server Type is FTP: Allows you to add statements to the FTP script that control the format of the file list returned by the FTP task. The Agent depends on the file list being in Unix "long" format (that is, what you would see if you entered "ls -l" from the command shell) in order to correctly and reliably parse out file name and size (when a Stable period is specified). If the FTP Server is configured to return a different format, the Server may support commands that alter the format.</p> <p>For example, the following statements may be used for an FTP File Monitor task executing against an IBM iSeries (AS/400) FTP Server to ensure a correctly-formatted file list:</p> <pre style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;">SITE LISTFMT 1 SITE NAMEFMT 1</pre> <p>If the FTP File Monitor task is executing against a Microsoft FTP Server and that Server is configured to return a file list in DOS format, the following statement will toggle the format to a Unix-style listing.</p> <pre style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;">SITE DIRSTYLE</pre> <p>Not all FTP client/server implementations provide statements that can alter the format of the LS command, which the FTP File Monitor task issues to generate the file listing. However, those implementations may support the DIR command, which can return the file list in the correct format. If the DIR command is specified in the List Format Options field, the FTP File Monitor task will use the results from that command to obtain the file sizes. In such cases, the FTP script will contain the DIR and LS commands, but since statements in the List Format Options field are inserted into the script prior the LS command, the results from the DIR command are parsed first.</p> <p>If the DIR command is necessary to obtain the correct file list format, simply specify that command along with the same value specified in the Remote Filename field. For example, if Remote Filename is <code>/uagtests/data/somefile*.txt</code>, enter the following into the List Format Options field:</p> <pre style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;">DIR /uagtests/data/somefile*.txt</pre> <p>This statement also can be used with other commands to get the correct output. For example, if a Windows FTP Server is configured to return file lists in Windows format, use SITE and DIR commands together in the List Format Options field:</p> <pre style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;">SITE DIRSTYLE DIR /uagtests/data/somefile*.txt</pre> <p>Invalid statements or valid statements that do not control the file list format are ignored.</p>
Remote Server	<p>Required. Name or IP address of the File Transfer server. This machine may or may not be the same as the Opwise agent machine.</p> <p>You also can specify a non-standard FTP or SFTP port:</p> <ul style="list-style-type: none"> • For FTP, specify the port number separated from the host name with a space: "some.server.com 2222". • For SFTP, specify the port number separated from the host name with a colon: "some.server.com:2222".
FTP Credentials	<p>Login credentials the Opwise agent will use to access the FTP or SFTP server machine. If the File Transfer server and Opwise agent are running on the same machine, enter the same credentials as those you entered in the Credentials field.</p>

Transfer Mode	<p>Transfer mode. Options:</p> <ul style="list-style-type: none"> • Active • Passive • Extended Passive
FTP Credentials Variable	<p>Optional. If enabled, the FTP Credentials field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: <code>\${variable name}</code>. The variable must be a supported type as described in Variables and Functions.</p>
Remote Filename	<p>Required. Path and file name on the remote server.</p>
Job Card (z/OS only)	<p>For z/OS, the job card information for the JCL statement. Example:</p> <pre>//File TransferJOB01 JOB (File Transfer,001),FANNY,MSGCLASS=X,MSGLEVEL=(1,1),NOTIFY=&SYSUID,CLASS=A</pre>
User Estimated End Time	<p>Task instance only; system-supplied. If the user entered information into the User Estimated Duration field in the task definition, Opwise uses this information to calculate an end time for the task instance, based on the date/time the task instance started.</p>
Shortest Estimated End Time	<p>Task instance only; system-supplied.</p>
Average Estimated End Time	<p>Task instance only; system-supplied.</p>
Longest Estimated End Time	<p>Task instance only; system-supplied.</p>
Late Start	<p>If enabled, and if the task instance starts after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late start (see Late Start Type). To determine whether a task instance started late, open the task instance and locate the Started Late field; the field is checked if the instance started after the specified time. This field only appears on the task instance if the user specified a Late Start in the task definition.</p>
Late Start Type	<p>Required if Late Start is enabled. Options are:</p> <ul style="list-style-type: none"> • Time - Flag the task if it starts after the specified time. • Duration - Flag the task if it starts a certain amount of time after the programmed start time. The task must have a specific start time.
Late Start Time	<p>Time after which the task start time is considered late. Use hh:mm, 24-hour time</p>
Late Start Duration	<p>Duration (amount of relative time) after which the task is considered to have started late. For a task within a workflow, the duration is the period between the time the workflow starts and the time the task itself starts. For example, a task might have a Late Start Duration of 60 minutes. If the workflow starts at 9:00 a.m. but the task itself does not start until 10:30, the task has started late.</p> <p>For a task that is not within a workflow, Late Start Duration has meaning only if the task has been held upon starting. For example, if a task has a Late Start Duration of 60 minutes and the Hold on Start field is enabled, if the task is not released from hold within the amount of time specified in the Late Start Duration field, the task has started late.</p>

Started Late	Task instance only; system-supplied. This field is flagged if the task started later than the time specified in the Late Start fields.
Late Finish	If enabled, and if the task instance finishes after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late finish (see Late Finish Type). To determine whether a task instance finished late, open the task instance and locate the Finished Late field; the field is checked if the instance finished after the specified time or lasted longer than expected. This field only appears on the task instance if the user specified a Late Finish in the task definition.
Late Finish Type	Required if Late Finish is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it finishes after the specified time (see Late Finish Time). • Duration - Flag the task if it finishes a certain amount of time after the programmed finish time (see Late Finish Duration). The task must have a specific finish time.
Late Finish Time	If Late Finish Type is Time, use this to specify the time after which the task finish time is considered late. Use hh:mm, 24-hour time.
Late Finish Duration	If Late Finish Type is Duration, use this to specify the longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Finished Late	Task instance only; system-supplied. This field is flagged if the task finished later than the time or duration specified in the Late Finish fields.
Early Finish	If enabled, and if the task instance finishes before the time or period specified, the task instance is flagged as early. You can specify a time or duration to determine an early finish (see Early Finish Type). To determine whether a task instance finished early, open the task instance and locate the Finished Early field; the field is checked if the instance finished before the specified time or did not last as long as expected. This field only appears on the task instance if the user added Early Finish specifications to the task definition.
Early Finish Type	Required if Early Finish is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it finishes before the specified time (see Early Finish Time). • Duration - Flag the task if it finishes a certain amount of time before the programmed finish time (see Early Finish Duration). The task must have a specific finish time.
Early Finish Time	If Early Finish Type is Time, use this to specify the time before which the task finish time is considered early. That is, enter a time at which the task should still be running. Use hh:mm, 24-hour time.
Early Finish Duration	If Early Finish Type is Duration, use this to specify the shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Finished Early	Task instance only; system-supplied. This field is flagged if the task finished earlier than the time specified in the Early Finish fields.
First Time Ran	Task definition only; system-supplied. Displays after the first time the task runs. The date and time this task first ran.
Lowest Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. The shortest amount of time this task has taken to run.

Last Time Ran	Task definition only; system-supplied. Displays after the first time the task runs. The most recent date and time the task ran.
Average Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. Shows the average amount of time this task takes to run.
Number of Instances	Task definition only; system-supplied. Displays after the first time the task runs. Shows the number of times this task has run.
Highest Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. The longest amount of time this task has taken to run.
Last Instance Duration	Task definition only; system-supplied. Displays after the first time the task runs. The amount of time the task took to run the last time it ran.
Virtual Resource Priority	<p>Priority for acquiring a resource when two or more tasks are waiting for the resource. This priority applies to all resources required by the task.</p> <p>Options: 1 (high) - 20 (low).</p> <p>Default is 10.</p>
Hold Resources on Failure	If enabled, the task instance will continue to hold Renewable resources if the task instance fails. Renewable resources will be returned only if the task instance status is either Complete, Finished, or Skipped.
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Launch Task button	Manually launches the task.
View Instances button	Displays a list of task instances for which there has been a status change or a modification to the task instance record within the last 30 days (an Updated on Last 30 Days filter has been pre-selected for the list).
Delete button	Deletes the current record.
View Parent button	Task instance only; Displays this task's parent task (workflow), if any.
Show Details button	Task instance only; displays detailed information about this task instance.
Retrieve Output button	Task instance only; see Retrieving Output .
Skip button	Task instance only; for tasks loaded into the schedule that haven't run yet. Allows you to tell Opwise to skip this task. See Skipping a Task .

Hold button	Task instance only; see Putting a Task on Hold .
Force Finish button	Task instance only; see Force Finishing a Task .
Output tab	Task instance only. Displays output generated from the process, if any, based on specifications provided by the user in the Automatic Output Retrieval fields in the task definition.
Variables tab	Displays all variables associated with this record.
Actions tab	<p>Allows you to specify actions that Opwise will take automatically based on events that occur during the execution of this task. Events are task instance status, exit codes, late start, late finish, and early finish. Actions are:</p> <ul style="list-style-type: none"> • Abort Action - Abort the task if certain events occur. For details, see Setting Abort Actions. • Email Notification - Send an email if certain events occur. For details, see Creating Email Notifications. • Set Variable - Used in workflows to set a variable based on the occurrence of certain events. For details, see Setting Variables within a Workflow. • SNMP Notification - Send an email if certain events occur. For details, see Creating SNMP Notifications. • System Operation - Run an Opwise system operation based on specified conditions. For details, see Setting System Operations.
Task Virtual Resources tab	Lists Virtual Resources to which this task is assigned.
Mutually Exclusive Tasks tab	Displays all tasks that have been set to be mutually exclusive of this task.
Triggers tab	Displays a list of all triggers that have been defined to launch this task. Also allows you to add new triggers. If you add a new trigger from this location, Opwise automatically constructs a default trigger name as follows: <current task name>#TRIGGER#. You can change the default name if desired. For instructions on creating triggers, see Creating Triggers .
Notes tab	Displays all notes associated with this task.
Versions Tab	Stores copies of all previous versions of the current record. See Record Versioning .

Specifying When a Task Runs

The FTP File Monitor can be used only within a workflow; you cannot run a FTP File Monitor task based on a trigger.

Monitoring Task Execution

You can monitor all system activity from the [Activity screen](#) and can view activity history from the [Activity History screen](#).

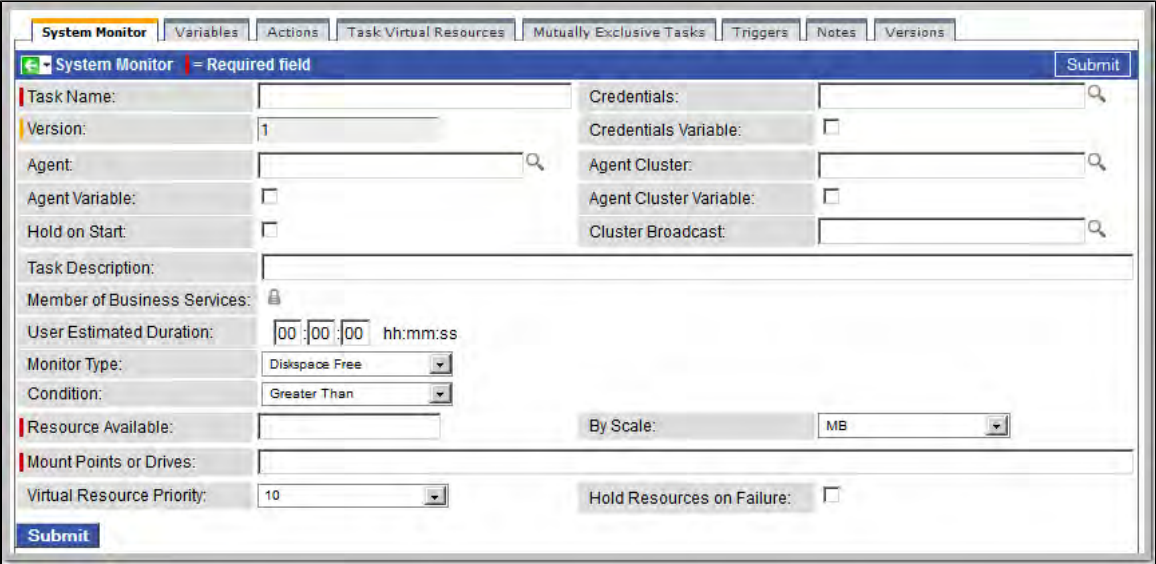
System Monitor Task

- [Overview](#)
- [Creating a New System Monitor Task](#)
- [System Monitor Field Descriptions](#)
- [Specifying When a Task Runs](#)
- [Monitoring Task Execution](#)

Overview

The System Monitor task allows you to monitor a specific remote machine and check for free disk space. You might use this task to check for sufficient disk space before running a task on it that requires a specific amount. In order for this task to execute, the remote machine must have an Agent running on it.

Creating a New System Monitor Task

Step 1	From the navigation pane, select Automation Center > Tasks > System Monitors . The System Monitors List displays.
Step 2	Click New . The System Monitor Definition screen displays.
	
Step 3	Using the field descriptions provided below as a guide, complete the fields as needed.
Step 4	Click the Submit button to save the record and return to the menu, or right-click on the title bar and select Save to save the record and remain on the current display.
Step 5	If appropriate, repeat these steps for any additional tasks you want to add.

Task run statistics appear after the first time this task has been launched.

System Monitor Field Descriptions

The table below describes the fields, buttons, and tabs on the task definition and task instance screens. Color coding is provided that differentiates the following three types of fields:

- Fields that display on the task definition and task instance screens are shown in black.
- Fields that display only on the task definition screen are shown in green.
- Fields that display only on the task instance screen are shown in maroon.

Field Name	Description
------------	-------------

Task/Instance Name	Required. Name used within Opwise to identify this task. Up to 40 alphanumeric; variables supported. It is the user's responsibility to develop a workable naming scheme for tasks.
Invoked by	Task instance only; system-supplied. How the task instance was launched. One of the following: <ul style="list-style-type: none"> • Trigger: (Trigger Name) - The instance was launched by the named trigger. • Workflow: (Workflow Name) - The instance was launched by the named workflow. • Manually Launched - The instance was launched by a user. To determine the name of the user: <ol style="list-style-type: none"> 1. From the Activity or Task Instances screen, click the task instance name to open the record. 2. The Execution User field identifies the user who launched the task instance.
Task	Task instance only; system-supplied. Hover over the paper icon to display more information about the task instance. Click the paper icon to display the task definition record.
Execution User	Task instance only; system-supplied. If the task was launched manually, the ID of the user who launched it.
Instance Reference Id	Task instance only; system-supplied. Opwise increments this number each time the task is run.
Credentials	Credentials under which an Agent runs this task. These credentials override any credentials provided in the Agent resource definition for any Agent running this task.
Version	Task definition only; system-supplied. The version number of the current record, which is incremented by Opwise every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning.
Credentials Variable	Optional. If enabled, the Credentials field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: <code>#{variable name}</code> . The variable must be a supported type as described in Variables and Functions .
Agent	Optional. The name of the agent resource definition that identifies the machine where the operation will run. If you do not specify an agent, you must specify an agent cluster (see below).
Agent Cluster	Optional. You can specify an agent cluster in addition to or in place of a specific agent. An agent cluster is a group of agents, one of which Opwise will choose to run this task. If you specify an agent and an agent cluster, Opwise Automation Center first tries to run the task on the specific agent. If the agent is not available, Opwise reverts to the agent cluster. See Agent Clusters for more information.
Agent Variable	Optional. If enabled, the Agent field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: <code>#{variable name}</code> . The variable must be a supported type as described in Variables and Functions .
Agent Cluster Variable	Optional. If enabled, the Agent Cluster field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: <code>#{variable name}</code> . The variable must be a supported type as described in Variables and Functions .
Hold on Start	If enabled, when the task is launched it appears in the Activity display with a status of Held. The task runs when the user releases it.

Cluster Broadcast	Task definition only; optional. You can specify a Cluster Broadcast in place of a specific agent and/or Agent Cluster. When you specify an agent cluster in the Cluster Broadcast field, Opwise Automation Center runs the task on all the agents in the cluster. Each instance of the task running on its own agent becomes a separate task instance record in the database and displays separately in the Activity monitor. See Agent Clusters for more information about defining agent clusters.
Task Description	User-supplied description of this record.
Status	Task instance only; system-supplied. See Task Instance Statuses .
Exit Code	Task instance only; system-supplied. The exit code captured by the agent when executing the task (for example, a command or script).
Status Description	Task instance only; system-supplied. Provides additional information, if any, about the status of the task.
Start Time	Task instance only; system-supplied. The date and time the task started.
Duration	Task instance only; system-supplied. The amount of time the task took to run.
End Time	Task instance only; system-supplied. The date and time the task instance completed.
Member of Business Services	User-defined. Allows you to select one or more Business Services that this record definition belongs to. Click the lock icon to unlock the field and select Business Services .
User Estimated Duration	Task definition only; optional. The estimated amount of time it should normally take to run this task. Opwise uses this information to calculate the User Estimated End Time on a task instance record.
Monitor Type	Type of system status being monitored for. Options: <ul style="list-style-type: none"> • Diskspace Free - Monitors for the amount of free diskspace.
Condition	Specifies whether you want to check for free diskspace greater than or less than the amount specified in the Resource Available field.
Resource Available	Required. Used in conjunction with the By Scale field. Enter a number indicating the amount of the resource you are checking for. For example, to check to see if the machine has at least one gigabyte of free diskspace, select Greater Than in the Condition field, type 1 in the Resource Available field, and select GB in the By Scale field.
By Scale	Scale of the number you entered in the Resource Available field. Options: KB (kilobyte), MB (megabyte), GB (gigabyte).
Mount Points or Drives	Required. Use this field to limit the check to a specific mount point or drive, such as drive C: for Windows.
User Estimated End Time	Task instance only; system-supplied. If the user entered information into the User Estimated Duration field in the task definition, Opwise uses this information to calculate an end time for the task instance, based on the date/time the task instance started.

Shortest Estimated End Time	Task instance only; system-supplied.
Average Estimated End Time	Task instance only; system-supplied.
Longest Estimated End Time	Task instance only; system-supplied.
First Time Ran	Task definition only; system-supplied. Displays after the first time the task runs. The date and time this task first ran.
Lowest Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. The shortest amount of time this task has taken to run.
Last Time Ran	Task definition only; system-supplied. Displays after the first time the task runs. The most recent date and time the task ran.
Average Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. Shows the average amount of time this task takes to run.
Number of Instances	Task definition only; system-supplied. Displays after the first time the task runs. Shows the number of times this task has run.
Highest Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. The longest amount of time this task has taken to run.
Last Instance Duration	Task definition only; system-supplied. Displays after the first time the task runs. The amount of time the task took to run the last time it ran.
Virtual Resource Priority	<p>Priority for acquiring a resource when two or more tasks are waiting for the resource. This priority applies to all resources required by the task.</p> <p>Options: 1 (high) - 20 (low).</p> <p>Default is 10.</p>
Hold Resources on Failure	If enabled, the task instance will continue to hold Renewable resources if the task instance fails. Renewable resources will be returned only if the task instance status is either Complete, Finished, or Skipped.
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Launch Task button	Manually launches the task.
View Instances button	Displays a list of task instances for which there has been a status change or a modification to the task instance record within the last 30 days (an Updated on Last 30 Days filter has been pre-selected for the list).

Delete button	Deletes the current record.
View Parent button	Task instance only; Displays this task's parent task (workflow), if any.
Show Details button	Task instance only; displays detailed information about this task instance.
Retrieve Output button	Task instance only; see Retrieving Output .
Skip button	Task instance only; for tasks loaded into the schedule that haven't run yet. Allows you to tell Opwise to skip this task. See Skipping a Task .
Hold button	Task instance only; see Putting a Task on Hold .
Force Finish button	Task instance only; see Force Finishing a Task .
Re-run button	Task instance only; see Re-running a Task .
Output tab	Task instance only. Displays output generated from the process, if any, based on specifications provided by the user in the Automatic Output Retrieval fields in the task definition.
Variables tab	Displays all variables associated with this record.
Actions tab	<p>Allows you to specify actions that Opwise will take automatically based on events that occur during the execution of this task. Events are task instance status, exit codes, late start, late finish, and early finish. Actions are:</p> <ul style="list-style-type: none"> • Abort Action - Abort the task if certain events occur. For details, see Setting Abort Actions. • Email Notification - Send an email if certain events occur. For details, see Creating Email Notifications. • Set Variable - Used in workflows to set a variable based on the occurrence of certain events. For details, see Setting Variables within a Workflow. • SNMP Notification - Send an email if certain events occur. For details, see Creating SNMP Notifications. • System Operation - Run an Opwise system operation based on specified conditions. For details, see Setting System Operations.
Task Virtual Resources tab	Lists Virtual Resources to which this task is assigned.
Mutually Exclusive Tasks tab	Displays all tasks that have been set to be mutually exclusive of this task.
Triggers tab	Displays a list of all triggers that have been defined to launch this task. Also allows you to add new triggers. If you add a new trigger from this location, Opwise automatically constructs a default trigger name as follows: <current task name>#TRIGGER#. You can change the default name if desired. For instructions on creating triggers, see Creating Triggers .
Notes tab	Displays all notes associated with this task.
Versions Tab	Stores copies of all previous versions of the current record. See Record Versioning .

Specifying When a Task Runs

You can run the task as part of a [workflow](#), [specify triggers](#) that run the task automatically based on times or events, or [run the task manually](#).

Monitoring Task Execution

You can monitor all system activity from the [Activity screen](#) and can view activity history from the [Activity History screen](#).

Copying Tasks

- [Overview](#)
- [Copying One or More Tasks from the Tasks List Screen](#)
- [Copying a Task from the Task Definition Screen](#)

Overview

You can make copies of all Opswise records, including tasks, using the standard methods for copying: Insert, Insert and Stay (see [Saving, Updating, Deleting, and Copying Records](#)).

However, these methods do not make copies of other records that may be attached to the task, such as Notifications, Actions, Variables, and so on.

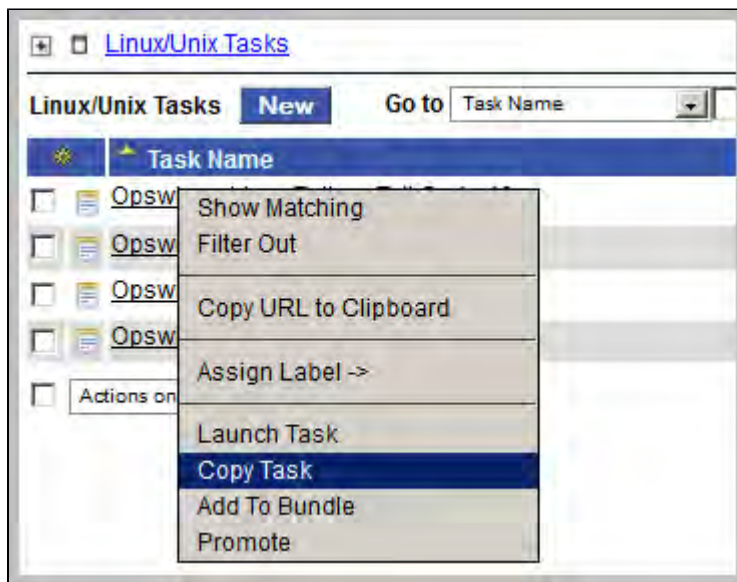
The Copy Task option allows you to make a complete copy of a task, including all of its associated records, such as variables and notes. It does not copy referenced records, such as virtual resources, but retains the relationship to these records for the copied task.

Copying One or More Tasks from the Tasks List Screen

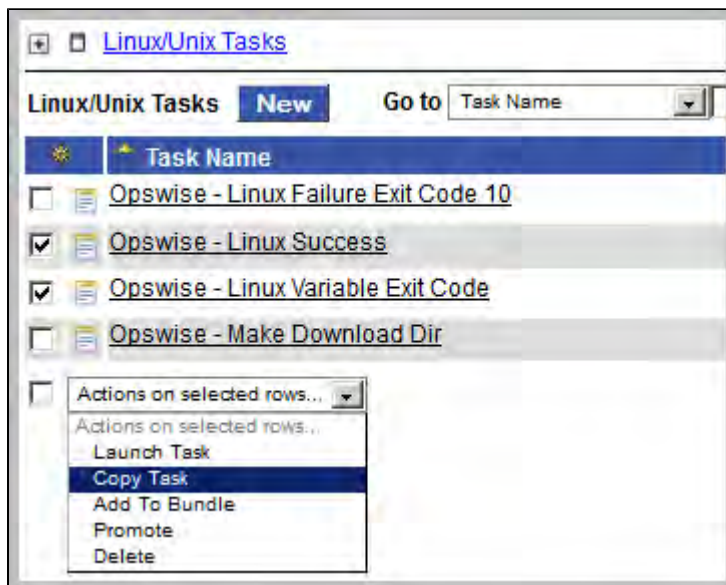
Step 1	From the navigation pane, select a task type from Automation Center > Tasks . The Tasks List screen for that task type displays.
Step 2	Locate the task(s) you want to copy (see Searching for Records).

Step 3 Copy the task(s) using either of two methods:

1. To copy a single task, hover over the task name and right-click. From the pop-up menu, select **Copy Task**.



2. To copy one or more tasks, click the box to the left of each task name. From the **Action on selected rows...** drop-down list at the bottom of the page, select **Copy Task**.



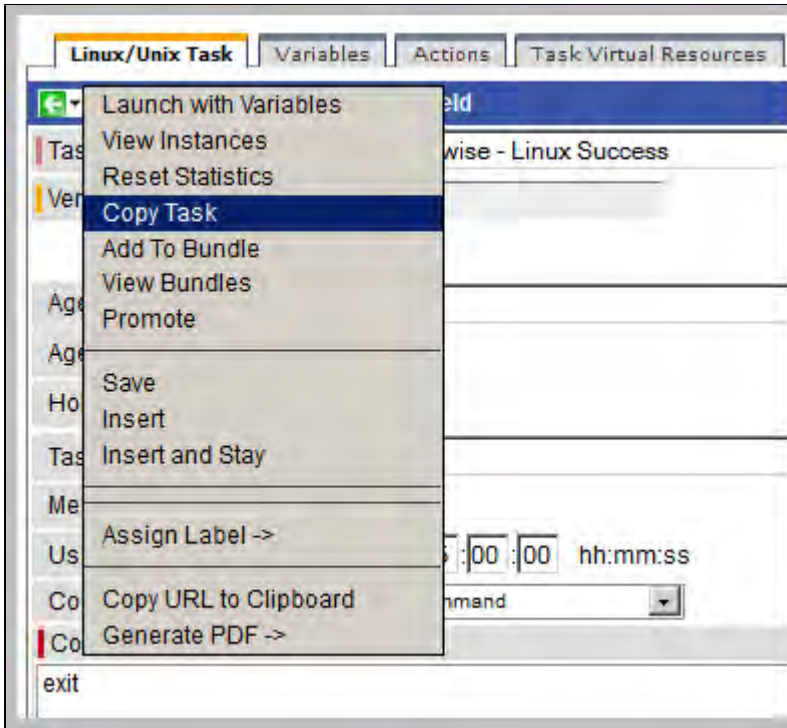
Step 4 Opwise copies the task(s), automatically creating the new name by prepending the original name with "Copy of" (for example, "Copy of Task XYZ"), and adds it to the list. If the new name already exists, Opwise appends a counter to the name, such as "Copy of Task XYZ 1", "Copy of Task XYZ 2", and so on, until it finds a name that is available.

Step 5 To modify the name or any other information in the task, open the new task, make your changes, and click **Update**.

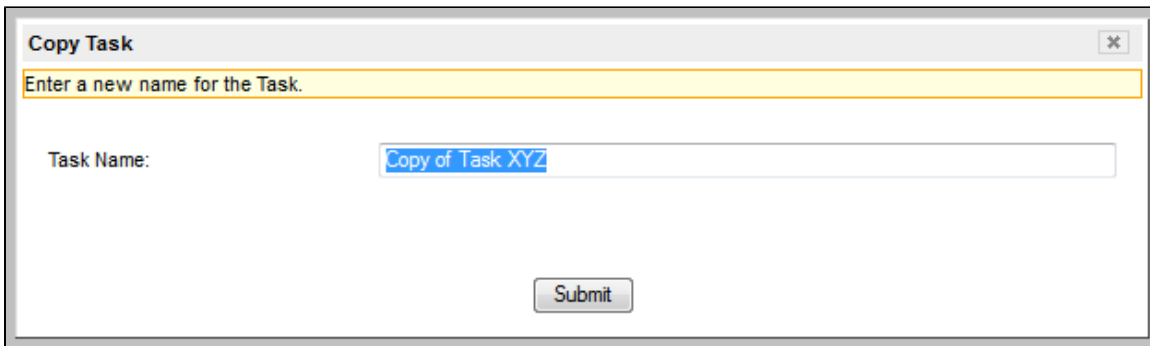
Copying a Task from the Task Definition Screen

Step 1 Select a task from a Tasks List screen. The Task Definition screen for that task displays.

Step 2 Hover over the down arrow above **Task Name**: A pop-up menu displays:



Step 3 Click **Copy Task**. Another window appears, prompting for a name for the new task. The default is the original task name, prepended with "Copy of," as shown in the following example:



Step 4 Enter a new name for the task and click **Submit**. Opwise copies the task and all its attachments and saves it under the new name. (If the new name already exists, the copy will fail.)

Creating Email Notifications

- [Overview](#)
- [Attaching an Email Notification to a Task](#)
- [Email Notifications Field Descriptions](#)

Overview

You can attach one or more Email notifications to any Opwise task. For Workflow tasks, you can also specify whether you want the email to be triggered by the workflow itself, the workflow and/or its tasks, or by the tasks only.

You can trigger the notification based on one or more of the following events associated with the task instance of the task to which you attach the notification:

- Status or statuses of the task instance
- Exit code(s) generated by the program (along with at least one status)
- Late start
- Early or late finish

In order to generate Email notifications, there must be an [Email connection](#) defined, which provides the Email server name and other pertinent information.

You can also generate notifications based on the status of agents, servers (cluster nodes), and connectors. See [Sending Notifications on Opwise Component Status](#).

Attaching an Email Notification to a Task

Step 1	Display the task that will generate the Email notification.
Step 2	Click the Actions tab. The Actions List screen displays a list of all defined actions for that task.
Step 3	Click New . The Actions Wizard screen displays: <div data-bbox="224 1108 1373 1480" data-label="Image"> </div>

Step 4 Click **Email Notification**. The Email Notifications screen displays.

Step 5 Using the field descriptions provided below as a guide, complete the fields as needed.

Step 6 Click the **Submit** button to save the record and return to the Email notifications list, or right-click and select **Save** to save the record and remain on the current display.

Step 7 If appropriate, repeat these steps for any additional Email notifications you want to add.

Email Notifications Field Descriptions

The table below describes the fields and buttons on the Email notifications screen.

Field Name	Description
Action Inheritance	<p>Workflow tasks only. Specifies what records these instructions apply to. Options:</p> <ul style="list-style-type: none">• SELF - These instructions apply only to the workflow and are not inherited by its children tasks.• SELF/CHILDREN - These instructions apply to the workflow and its contained tasks (children).• CHILDREN - These instructions apply only to the tasks within the workflow (children).

Status	<p>The status of this task. To trigger an abort action, you can specify status only, or status and exit code. You can specify as many statuses as needed. Options:</p> <table border="1" data-bbox="269 210 1481 1797"> <thead> <tr> <th data-bbox="269 210 423 258">Status</th> <th data-bbox="431 210 1481 258">Description</th> </tr> </thead> <tbody> <tr> <td data-bbox="269 268 423 296">Defined</td> <td data-bbox="431 268 1481 296">All task types. The new task instance has been created (the task has been launched). Not yet implemented.</td> </tr> <tr> <td data-bbox="269 306 423 333">Waiting</td> <td data-bbox="431 306 1481 333">All task types. The task has been loaded by a workflow and is waiting to run.</td> </tr> <tr> <td data-bbox="269 344 423 371">Held</td> <td data-bbox="431 344 1481 371">All task types. The task has been put on hold by a user.</td> </tr> <tr> <td data-bbox="269 382 423 457">Resource Requested</td> <td data-bbox="431 382 1481 457">All tasks with a virtual resource defined go immediately to a status of Resource Requested. If the resource is available, the task then moves to the next appropriate processing status.</td> </tr> <tr> <td data-bbox="269 468 423 564">Resource Wait</td> <td data-bbox="431 468 1481 564">All tasks with a virtual resource defined go immediately to a status of Resource Requested. If the resource is not available, the task goes to a status of Resource Wait. When the resource becomes available, the task moves to the next appropriate processing status</td> </tr> <tr> <td data-bbox="269 575 423 630">Execution Wait</td> <td data-bbox="431 575 1481 630">Agent-based tasks. The task must wait to be completed; either the Agent/Agent Cluster running the task has reached its Task Execution Limit, or the ability of the Agent/Agent Cluster to run tasks has been suspended.</td> </tr> <tr> <td data-bbox="269 640 423 667">Undeliverable</td> <td data-bbox="431 640 1481 667">Agent-based tasks. The agent is unavailable.</td> </tr> <tr> <td data-bbox="269 678 423 705">Queued</td> <td data-bbox="431 678 1481 705">Agent-based tasks only. The task has been queued on a resource.</td> </tr> <tr> <td data-bbox="269 716 423 791">Submitted</td> <td data-bbox="431 716 1481 791">z/OS only. The task has been submitted to the z/OS Job Entry subsystem and scheduled by the z/OS Job Scheduler.</td> </tr> <tr> <td data-bbox="269 802 423 856">Action Required</td> <td data-bbox="431 802 1481 856">Manual tasks only. When a Manual task launches, it goes into Action Required status, meaning a user must perform some manual activity.</td> </tr> <tr> <td data-bbox="269 867 423 924">Started</td> <td data-bbox="431 867 1481 924">Agent-based and Manual tasks only. The task has started. For agent-based tasks, this means the agent has received the task.</td> </tr> <tr> <td data-bbox="269 934 423 961">Running</td> <td data-bbox="431 934 1481 961">All task types. The task is running. For agent-based tasks, the agent has started running the program.</td> </tr> <tr> <td data-bbox="269 972 423 1257">Running Problems</td> <td data-bbox="431 972 1481 1257"> Workflows only. One or more tasks within the workflow has one of the following statuses: <ul style="list-style-type: none"> • Held • Undeliverable • Running Problems (for sub-workflows) • Cancel Pending • In Doubt • Start Failure • Cancelled </td> </tr> <tr> <td data-bbox="269 1268 423 1365">In Doubt</td> <td data-bbox="431 1268 1481 1365">Agent-based tasks only. The agent is "in doubt" about the current status of the task instance. This may occur if an agent or agent connection goes down. In this case, the agent restarts and reviews its data about tasks in progress. If the agent finds a task still running, it resumes normal monitoring. If the agent cannot find the task, this usually indicates that the task completed, but the agent considers the task status to be "in doubt."</td> </tr> <tr> <td data-bbox="269 1375 423 1402">Start Failure</td> <td data-bbox="431 1375 1481 1402">All task types. The task was unable to start.</td> </tr> <tr> <td data-bbox="269 1413 423 1488">Confirmation Required</td> <td data-bbox="431 1413 1481 1488">z/OS only. If you make JCL changes and restart a z/OS task, Opwise will put the task into Confirmation Required status and prompt you for a confirmation. For detailed processing steps, see Rerunning a z/OS Task.</td> </tr> <tr> <td data-bbox="269 1499 423 1526">Cancelled</td> <td data-bbox="431 1499 1481 1526">All task types. The task was cancelled by a user.</td> </tr> <tr> <td data-bbox="269 1537 423 1564">Failed</td> <td data-bbox="431 1537 1481 1564">All task types. The task ran to a failure status.</td> </tr> <tr> <td data-bbox="269 1575 423 1602">Skipped</td> <td data-bbox="431 1575 1481 1602">All task types. The task was skipped by a user.</td> </tr> <tr> <td data-bbox="269 1612 423 1688">Finished</td> <td data-bbox="431 1612 1481 1688">All task types. The task was forced by the user to finish. The user may do this in cases where the task had "Cancelled" or "Failed" status, and the user needed to release other task instances depending on the successful completion of this task instance in a workflow. For more information, see Force Finishing a Task.</td> </tr> <tr> <td data-bbox="269 1698 423 1726">Success</td> <td data-bbox="431 1698 1481 1726">All task types. The task has completed successfully.</td> </tr> </tbody> </table>	Status	Description	Defined	All task types. The new task instance has been created (the task has been launched). Not yet implemented.	Waiting	All task types. The task has been loaded by a workflow and is waiting to run.	Held	All task types. The task has been put on hold by a user.	Resource Requested	All tasks with a virtual resource defined go immediately to a status of Resource Requested. If the resource is available, the task then moves to the next appropriate processing status.	Resource Wait	All tasks with a virtual resource defined go immediately to a status of Resource Requested. If the resource is not available, the task goes to a status of Resource Wait. When the resource becomes available, the task moves to the next appropriate processing status	Execution Wait	Agent-based tasks. The task must wait to be completed; either the Agent/Agent Cluster running the task has reached its Task Execution Limit, or the ability of the Agent/Agent Cluster to run tasks has been suspended.	Undeliverable	Agent-based tasks. The agent is unavailable.	Queued	Agent-based tasks only. The task has been queued on a resource.	Submitted	z/OS only. The task has been submitted to the z/OS Job Entry subsystem and scheduled by the z/OS Job Scheduler.	Action Required	Manual tasks only. When a Manual task launches, it goes into Action Required status, meaning a user must perform some manual activity.	Started	Agent-based and Manual tasks only. The task has started. For agent-based tasks, this means the agent has received the task.	Running	All task types. The task is running. For agent-based tasks, the agent has started running the program.	Running Problems	Workflows only. One or more tasks within the workflow has one of the following statuses: <ul style="list-style-type: none"> • Held • Undeliverable • Running Problems (for sub-workflows) • Cancel Pending • In Doubt • Start Failure • Cancelled 	In Doubt	Agent-based tasks only. The agent is "in doubt" about the current status of the task instance. This may occur if an agent or agent connection goes down. In this case, the agent restarts and reviews its data about tasks in progress. If the agent finds a task still running, it resumes normal monitoring. If the agent cannot find the task, this usually indicates that the task completed, but the agent considers the task status to be "in doubt."	Start Failure	All task types. The task was unable to start.	Confirmation Required	z/OS only. If you make JCL changes and restart a z/OS task, Opwise will put the task into Confirmation Required status and prompt you for a confirmation. For detailed processing steps, see Rerunning a z/OS Task .	Cancelled	All task types. The task was cancelled by a user.	Failed	All task types. The task ran to a failure status.	Skipped	All task types. The task was skipped by a user.	Finished	All task types. The task was forced by the user to finish. The user may do this in cases where the task had "Cancelled" or "Failed" status, and the user needed to release other task instances depending on the successful completion of this task instance in a workflow. For more information, see Force Finishing a Task .	Success	All task types. The task has completed successfully.
Status	Description																																												
Defined	All task types. The new task instance has been created (the task has been launched). Not yet implemented.																																												
Waiting	All task types. The task has been loaded by a workflow and is waiting to run.																																												
Held	All task types. The task has been put on hold by a user.																																												
Resource Requested	All tasks with a virtual resource defined go immediately to a status of Resource Requested. If the resource is available, the task then moves to the next appropriate processing status.																																												
Resource Wait	All tasks with a virtual resource defined go immediately to a status of Resource Requested. If the resource is not available, the task goes to a status of Resource Wait. When the resource becomes available, the task moves to the next appropriate processing status																																												
Execution Wait	Agent-based tasks. The task must wait to be completed; either the Agent/Agent Cluster running the task has reached its Task Execution Limit, or the ability of the Agent/Agent Cluster to run tasks has been suspended.																																												
Undeliverable	Agent-based tasks. The agent is unavailable.																																												
Queued	Agent-based tasks only. The task has been queued on a resource.																																												
Submitted	z/OS only. The task has been submitted to the z/OS Job Entry subsystem and scheduled by the z/OS Job Scheduler.																																												
Action Required	Manual tasks only. When a Manual task launches, it goes into Action Required status, meaning a user must perform some manual activity.																																												
Started	Agent-based and Manual tasks only. The task has started. For agent-based tasks, this means the agent has received the task.																																												
Running	All task types. The task is running. For agent-based tasks, the agent has started running the program.																																												
Running Problems	Workflows only. One or more tasks within the workflow has one of the following statuses: <ul style="list-style-type: none"> • Held • Undeliverable • Running Problems (for sub-workflows) • Cancel Pending • In Doubt • Start Failure • Cancelled 																																												
In Doubt	Agent-based tasks only. The agent is "in doubt" about the current status of the task instance. This may occur if an agent or agent connection goes down. In this case, the agent restarts and reviews its data about tasks in progress. If the agent finds a task still running, it resumes normal monitoring. If the agent cannot find the task, this usually indicates that the task completed, but the agent considers the task status to be "in doubt."																																												
Start Failure	All task types. The task was unable to start.																																												
Confirmation Required	z/OS only. If you make JCL changes and restart a z/OS task, Opwise will put the task into Confirmation Required status and prompt you for a confirmation. For detailed processing steps, see Rerunning a z/OS Task .																																												
Cancelled	All task types. The task was cancelled by a user.																																												
Failed	All task types. The task ran to a failure status.																																												
Skipped	All task types. The task was skipped by a user.																																												
Finished	All task types. The task was forced by the user to finish. The user may do this in cases where the task had "Cancelled" or "Failed" status, and the user needed to release other task instances depending on the successful completion of this task instance in a workflow. For more information, see Force Finishing a Task .																																												
Success	All task types. The task has completed successfully.																																												
Exit Codes	<p>Specifies one or more exit codes that will trigger the event. If you specify an exit code, you must also specify at least one status. Use commas to separate multiple exit codes; use a hyphen to specify a range. Example: 1, 5, 22-30.</p>																																												

On Late Start	Generate the action or notification if the task started late, based on the Late Start Time specified in the task.
On Late Finish	Generate the action or notification if the task finishes late, based on the Late Finish time specified in the task.
On Early Finish	Generate the action or notification if the task finishes early, based on the Early Finish Time specified in the task.
Description	Optional. Description of this email notification.
Email Template	Optional. The name of the Email template defined using the Email template screen. The Email template allows you to specify standard recipients and text for outgoing emails. Type in a name, or click the magnifying glass to browse to an existing Email template or create a new one. You must specify either an Email template or Email connection, or both. If you specify both, the Email server specified in the Email Connection record overrides the server in the template.
Email Connection	Required. Name of the Email connection defined using the Email connection screen. The email connection specifies information about the email server. You can also specify the Email connection in the Email template (see above). You must specify an Email template and/or an Email connection. If you specify an Email template and an Email connection, the server selected in the Email connection overrides the server selected in the Email template. Type in a name, click the magnifying glass to browse for an existing Email server definition, or create a new one.
Reply-To	Required. Specifies the email address of the sender. Use commas to separate multiple recipients. Variables supported.
To	Required. Specifies the email address of the recipient. Use commas to separate multiple recipients. Variables supported.
CC	Optional. Specifies the email address of the party being sent a copy of the email, if any. Use commas to separate multiple recipients. Variables supported.
BCC	Optional. Specifies the email address of the party being sent a blind (hidden) copy of the email, if any. Use commas to separate multiple recipients. Variables supported.
Subject	Optional. Specifies the subject line of the email. Variables supported.
Body	Optional. Contains the text of the email message. Variables supported. If both the email template and the email task contain text in the body, the text is appended.
Attach Standard Output	Attach any standard output generated by the associated task.
Attach Standard Error	Attach standard error data generated by the associated task.
Attach File	For agent-based tasks only; attach any single text file that is accessible by the agent. Full path name is required. Wildcards are NOT supported. The Controller will request the file from the agent. If the file does not exist, the agent will return a file output type with the content: OPSWISE WARNING - File is not available.
Start Line	Attach data beginning at the line indicated.

Number of Lines	Optional. Allows you to limit the retrieved data to the number of lines specified. If a Number of Lines value is not specified, the default is the value of the Retrieve Output Default Maximum Lines Opswise system property.
Scan Text:	Optional. Regex pattern that Opswise will search for a match for in STDOUT/STDERR or a specified file. Opswise will include "Number of Lines" above and below the first line matched.
File Name	For Attach File only, the path and filename of the file you want to attach to the email notification.
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Delete button	Deletes the current record.

Creating SNMP Notifications

- [Overview](#)
- [Attaching an SNMP Notification to a Task](#)
- [SNMP Notifications Field Descriptions](#)

Overview

You can attach one or more SNMP notifications to any Opwise task. For Workflow tasks, you can also specify whether you want the SNMP notification to be triggered by the workflow itself, the workflow and/or its tasks, or by the tasks only.

You can trigger the notification based on one or more of the following events associated with the task instance to which you attach the notification:

- Status or statuses of the task instance
- Exit code(s) generated by the program (along with at least one status)
- Late start
- Early or late finish

In order to generate SNMP notifications, there must be an [SNMP Manager](#) defined, which provides the server name and other pertinent information of the SNMP Manager that will receive the notification.

You can also generate notifications based on the status of agents, agent clusters, and connectors. See [Sending Notifications on Opwise Component Status](#).

Attaching an SNMP Notification to a Task

Step 1	Display the task that will generate the SNMP notification.
Step 2	Click the Actions tab. The Actions List screen displays a list of all defined actions for that task.
Step 3	Click New . The Actions Wizard screen displays. <div data-bbox="224 1087 1373 1457" data-label="Image"> </div>

Step 4 Click **SNMP Notification**. The SNMP Notifications screen displays.

Step 5 Using the field descriptions provided below as a guide, complete the fields as needed.

Step 6 Click the **Submit** button to save the record and return to the SNMP notifications list, or right-click and select **Save** to save the record and remain on the current display.

Step 7 If appropriate, repeat these steps for any additional SNMP notifications you want to add.

SNMP Notifications Field Descriptions

The table below describes the fields and buttons on the SNMP notifications screen.

Field Name	Description
Action Inheritance	<p>Workflow tasks only. Specifies what records these instructions apply to. Options:</p> <ul style="list-style-type: none"> • SELF - These instructions apply only to the workflow and are not inherited by its children tasks. • SELF/CHILDREN - These instructions apply to the workflow and its contained tasks (children). • CHILDREN - These instructions apply only to the tasks within the workflow (children).

Status	<p>The status of this task. To trigger a notification you can specify status only, or status and exit code. You can specify as many statuses as needed. Options:</p> <table border="1" data-bbox="269 210 1479 1797"> <thead> <tr> <th data-bbox="269 210 423 258">Status</th> <th data-bbox="431 210 1479 258">Description</th> </tr> </thead> <tbody> <tr> <td data-bbox="269 268 423 296">Defined</td> <td data-bbox="431 268 1479 296">All task types. The new task instance has been created (the task has been launched). Not yet implemented.</td> </tr> <tr> <td data-bbox="269 306 423 333">Waiting</td> <td data-bbox="431 306 1479 333">All task types. The task has been loaded by a workflow and is waiting to run.</td> </tr> <tr> <td data-bbox="269 344 423 371">Held</td> <td data-bbox="431 344 1479 371">All task types. The task has been put on hold by a user.</td> </tr> <tr> <td data-bbox="269 382 423 457">Resource Requested</td> <td data-bbox="431 382 1479 457">All tasks with a virtual resource defined go immediately to a status of Resource Requested. If the resource is available, the task then moves to the next appropriate processing status.</td> </tr> <tr> <td data-bbox="269 468 423 564">Resource Wait</td> <td data-bbox="431 468 1479 564">All tasks with a virtual resource defined go immediately to a status of Resource Requested. If the resource is not available, the task goes to a status of Resource Wait. When the resource becomes available, the task moves to the next appropriate processing status</td> </tr> <tr> <td data-bbox="269 575 423 630">Execution Wait</td> <td data-bbox="431 575 1479 630">Agent-based tasks. The task must wait to be completed; either the Agent/Agent Cluster running the task has reached its Task Execution Limit, or the ability of the Agent/Agent Cluster to run tasks has been suspended.</td> </tr> <tr> <td data-bbox="269 640 423 667">Undeliverable</td> <td data-bbox="431 640 1479 667">Agent-based tasks. The agent is unavailable.</td> </tr> <tr> <td data-bbox="269 678 423 705">Queued</td> <td data-bbox="431 678 1479 705">Agent-based tasks only. The task has been queued on a resource.</td> </tr> <tr> <td data-bbox="269 716 423 791">Submitted</td> <td data-bbox="431 716 1479 791">z/OS only. The task has been submitted to the z/OS Job Entry subsystem and scheduled by the z/OS Job Scheduler.</td> </tr> <tr> <td data-bbox="269 802 423 856">Action Required</td> <td data-bbox="431 802 1479 856">Manual tasks only. When a Manual task launches, it goes into Action Required status, meaning a user must perform some manual activity.</td> </tr> <tr> <td data-bbox="269 867 423 924">Started</td> <td data-bbox="431 867 1479 924">Agent-based and Manual tasks only. The task has started. For agent-based tasks, this means the agent has received the task.</td> </tr> <tr> <td data-bbox="269 934 423 961">Running</td> <td data-bbox="431 934 1479 961">All task types. The task is running. For agent-based tasks, the agent has started running the program.</td> </tr> <tr> <td data-bbox="269 972 423 1257">Running Problems</td> <td data-bbox="431 972 1479 1257"> Workflows only. One or more tasks within the workflow has one of the following statuses: <ul style="list-style-type: none"> • Held • Undeliverable • Running Problems (for sub-workflows) • Cancel Pending • In Doubt • Start Failure • Cancelled </td> </tr> <tr> <td data-bbox="269 1268 423 1365">In Doubt</td> <td data-bbox="431 1268 1479 1365">Agent-based tasks only. The agent is "in doubt" about the current status of the task instance. This may occur if an agent or agent connection goes down. In this case, the agent restarts and reviews its data about tasks in progress. If the agent finds a task still running, it resumes normal monitoring. If the agent cannot find the task, this usually indicates that the task completed, but the agent considers the task status to be "in doubt."</td> </tr> <tr> <td data-bbox="269 1375 423 1402">Start Failure</td> <td data-bbox="431 1375 1479 1402">All task types. The task was unable to start.</td> </tr> <tr> <td data-bbox="269 1413 423 1488">Confirmation Required</td> <td data-bbox="431 1413 1479 1488">z/OS only. If you make JCL changes and restart a z/OS task, Opwise will put the task into Confirmation Required status and prompt you for a confirmation. For detailed processing steps, see Rerunning a z/OS Task.</td> </tr> <tr> <td data-bbox="269 1499 423 1526">Cancelled</td> <td data-bbox="431 1499 1479 1526">All task types. The task was cancelled by a user.</td> </tr> <tr> <td data-bbox="269 1537 423 1564">Failed</td> <td data-bbox="431 1537 1479 1564">All task types. The task ran to a failure status.</td> </tr> <tr> <td data-bbox="269 1575 423 1602">Skipped</td> <td data-bbox="431 1575 1479 1602">All task types. The task was skipped by a user.</td> </tr> <tr> <td data-bbox="269 1612 423 1751">Finished</td> <td data-bbox="431 1612 1479 1751">All task types. The task was forced by the user to finish. The user may do this in cases where the task had "Cancelled" or "Failed" status, and the user needed to release other task instances depending on the successful completion of this task instance in a workflow. For more information, see Force Finishing a Task.</td> </tr> <tr> <td data-bbox="269 1761 423 1789">Success</td> <td data-bbox="431 1761 1479 1789">All task types. The task has completed successfully.</td> </tr> </tbody> </table>	Status	Description	Defined	All task types. The new task instance has been created (the task has been launched). Not yet implemented.	Waiting	All task types. The task has been loaded by a workflow and is waiting to run.	Held	All task types. The task has been put on hold by a user.	Resource Requested	All tasks with a virtual resource defined go immediately to a status of Resource Requested. If the resource is available, the task then moves to the next appropriate processing status.	Resource Wait	All tasks with a virtual resource defined go immediately to a status of Resource Requested. If the resource is not available, the task goes to a status of Resource Wait. When the resource becomes available, the task moves to the next appropriate processing status	Execution Wait	Agent-based tasks. The task must wait to be completed; either the Agent/Agent Cluster running the task has reached its Task Execution Limit, or the ability of the Agent/Agent Cluster to run tasks has been suspended.	Undeliverable	Agent-based tasks. The agent is unavailable.	Queued	Agent-based tasks only. The task has been queued on a resource.	Submitted	z/OS only. The task has been submitted to the z/OS Job Entry subsystem and scheduled by the z/OS Job Scheduler.	Action Required	Manual tasks only. When a Manual task launches, it goes into Action Required status, meaning a user must perform some manual activity.	Started	Agent-based and Manual tasks only. The task has started. For agent-based tasks, this means the agent has received the task.	Running	All task types. The task is running. For agent-based tasks, the agent has started running the program.	Running Problems	Workflows only. One or more tasks within the workflow has one of the following statuses: <ul style="list-style-type: none"> • Held • Undeliverable • Running Problems (for sub-workflows) • Cancel Pending • In Doubt • Start Failure • Cancelled 	In Doubt	Agent-based tasks only. The agent is "in doubt" about the current status of the task instance. This may occur if an agent or agent connection goes down. In this case, the agent restarts and reviews its data about tasks in progress. If the agent finds a task still running, it resumes normal monitoring. If the agent cannot find the task, this usually indicates that the task completed, but the agent considers the task status to be "in doubt."	Start Failure	All task types. The task was unable to start.	Confirmation Required	z/OS only. If you make JCL changes and restart a z/OS task, Opwise will put the task into Confirmation Required status and prompt you for a confirmation. For detailed processing steps, see Rerunning a z/OS Task .	Cancelled	All task types. The task was cancelled by a user.	Failed	All task types. The task ran to a failure status.	Skipped	All task types. The task was skipped by a user.	Finished	All task types. The task was forced by the user to finish. The user may do this in cases where the task had "Cancelled" or "Failed" status, and the user needed to release other task instances depending on the successful completion of this task instance in a workflow. For more information, see Force Finishing a Task .	Success	All task types. The task has completed successfully.
Status	Description																																												
Defined	All task types. The new task instance has been created (the task has been launched). Not yet implemented.																																												
Waiting	All task types. The task has been loaded by a workflow and is waiting to run.																																												
Held	All task types. The task has been put on hold by a user.																																												
Resource Requested	All tasks with a virtual resource defined go immediately to a status of Resource Requested. If the resource is available, the task then moves to the next appropriate processing status.																																												
Resource Wait	All tasks with a virtual resource defined go immediately to a status of Resource Requested. If the resource is not available, the task goes to a status of Resource Wait. When the resource becomes available, the task moves to the next appropriate processing status																																												
Execution Wait	Agent-based tasks. The task must wait to be completed; either the Agent/Agent Cluster running the task has reached its Task Execution Limit, or the ability of the Agent/Agent Cluster to run tasks has been suspended.																																												
Undeliverable	Agent-based tasks. The agent is unavailable.																																												
Queued	Agent-based tasks only. The task has been queued on a resource.																																												
Submitted	z/OS only. The task has been submitted to the z/OS Job Entry subsystem and scheduled by the z/OS Job Scheduler.																																												
Action Required	Manual tasks only. When a Manual task launches, it goes into Action Required status, meaning a user must perform some manual activity.																																												
Started	Agent-based and Manual tasks only. The task has started. For agent-based tasks, this means the agent has received the task.																																												
Running	All task types. The task is running. For agent-based tasks, the agent has started running the program.																																												
Running Problems	Workflows only. One or more tasks within the workflow has one of the following statuses: <ul style="list-style-type: none"> • Held • Undeliverable • Running Problems (for sub-workflows) • Cancel Pending • In Doubt • Start Failure • Cancelled 																																												
In Doubt	Agent-based tasks only. The agent is "in doubt" about the current status of the task instance. This may occur if an agent or agent connection goes down. In this case, the agent restarts and reviews its data about tasks in progress. If the agent finds a task still running, it resumes normal monitoring. If the agent cannot find the task, this usually indicates that the task completed, but the agent considers the task status to be "in doubt."																																												
Start Failure	All task types. The task was unable to start.																																												
Confirmation Required	z/OS only. If you make JCL changes and restart a z/OS task, Opwise will put the task into Confirmation Required status and prompt you for a confirmation. For detailed processing steps, see Rerunning a z/OS Task .																																												
Cancelled	All task types. The task was cancelled by a user.																																												
Failed	All task types. The task ran to a failure status.																																												
Skipped	All task types. The task was skipped by a user.																																												
Finished	All task types. The task was forced by the user to finish. The user may do this in cases where the task had "Cancelled" or "Failed" status, and the user needed to release other task instances depending on the successful completion of this task instance in a workflow. For more information, see Force Finishing a Task .																																												
Success	All task types. The task has completed successfully.																																												
Exit Codes	<p>Specifies one or more exit codes that will trigger the event. If you specify an exit code, you must also specify at least one status. Use commas to separate multiple exit codes; use a hyphen to specify a range. Example: 1, 5, 22-30.</p>																																												

On Late Start	Generate the action or notification if the task started late, based on the Late Start Time specified in the task.
On Late Finish	Generate the action or notification if the task finishes late, based on the Late Finish time specified in the task.
On Early Finish	Generate the action or notification if the task finishes early, based on the Early Finish Time specified in the task.
Description	Optional. Description of this SNMP notification.
SNMP Manager	The SNMP Manager that will receive the SNMP notification.
Notification Severity	Optional. Informational only. Indicates the severity of this notification. Options: Normal, Warning, Minor, Major Critical.
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Delete button	Deletes the current record.

Setting Abort Actions

- [Overview](#)
- [Attaching an Abort Action to a Task](#)
- [Abort Actions Field Descriptions](#)

Overview

The Abort Action allows you to abort a waiting or running task instance by issuing a Skip command on a waiting task instance or issuing a Force Finish command on a running task instance.

Additionally, for running task instances, the Abort Action provides the ability to Force Finish and Cancel by using the [Cancel Process if Active](#) option and/or override the exit code of the Force Finished task instance by using the [Override Exit Code](#) option.

You can trigger this action based on one or more of the following events associated with the task instance:

- Status or statuses of the task instance
- Exit code(s) generated by the program (along with at least one status)
- Late start
- Early or late finish

You can attach one or more Abort Actions to any Opwise task. For Workflow tasks, you can also specify whether you want the Abort Action instructions to apply to the workflow itself, the workflow and/or its tasks, or to the tasks only.

Attaching an Abort Action to a Task

Step 1	Display the task to which you are attaching the Abort Action.
Step 2	Click the Actions tab. The Actions List screen displays a list of defined Actions for that task.
Step 3	Click New . The Action Wizard screen displays. <div data-bbox="224 1087 1373 1457" data-label="Image"> </div>

Step 4 Click **Abort Action**. The Abort Action screen displays.

Step 5 Using the field descriptions provided below as a guide, complete the fields as needed.

Step 6 Click the **Submit** button to save the record and return to the Actions list, or, right-click and select **Save** to save the record and remain on the current display.


Step 7 If appropriate, repeat these steps for any additional Actions you want to add.

Abort Actions Field Descriptions

The following table describes the fields and buttons on the Abort Actions screen.

Field Name	Description
Action Inheritance	<p>Workflow tasks only. Specifies what records these instructions apply to. Options:</p> <ul style="list-style-type: none"> • SELF - These instructions apply only to the workflow and are not inherited by its children tasks. • SELF/CHILDREN - These instructions apply to the workflow and its contained tasks (children). • CHILDREN - These instructions apply only to the tasks within the workflow (children).

Status	<p>The status of this task. To trigger an abort action, you can specify status only, or status and exit code. You can specify as many statuses as needed. Options:</p> <table border="1" data-bbox="269 210 1479 1801"> <thead> <tr> <th data-bbox="269 210 423 258">Status</th> <th data-bbox="431 210 1479 258">Description</th> </tr> </thead> <tbody> <tr> <td data-bbox="269 268 423 296">Defined</td> <td data-bbox="431 268 1479 296">All task types. The new task instance has been created (the task has been launched). Not yet implemented.</td> </tr> <tr> <td data-bbox="269 306 423 333">Waiting</td> <td data-bbox="431 306 1479 333">All task types. The task has been loaded by a workflow and is waiting to run.</td> </tr> <tr> <td data-bbox="269 344 423 371">Held</td> <td data-bbox="431 344 1479 371">All task types. The task has been put on hold by a user.</td> </tr> <tr> <td data-bbox="269 382 423 457">Resource Requested</td> <td data-bbox="431 382 1479 457">All tasks with a virtual resource defined go immediately to a status of Resource Requested. If the resource is available, the task then moves to the next appropriate processing status.</td> </tr> <tr> <td data-bbox="269 468 423 564">Resource Wait</td> <td data-bbox="431 468 1479 564">All tasks with a virtual resource defined go immediately to a status of Resource Requested. If the resource is not available, the task goes to a status of Resource Wait. When the resource becomes available, the task moves to the next appropriate processing status</td> </tr> <tr> <td data-bbox="269 575 423 623">Execution Wait</td> <td data-bbox="431 575 1479 623">Agent-based tasks. The task must wait to be completed; either the Agent/Agent Cluster running the task has reached its Task Execution Limit, or the ability of the Agent/Agent Cluster to run tasks has been suspended.</td> </tr> <tr> <td data-bbox="269 634 423 661">Undeliverable</td> <td data-bbox="431 634 1479 661">Agent-based tasks. The agent is unavailable.</td> </tr> <tr> <td data-bbox="269 672 423 699">Queued</td> <td data-bbox="431 672 1479 699">Agent-based tasks only. The task has been queued on a resource.</td> </tr> <tr> <td data-bbox="269 709 423 785">Submitted</td> <td data-bbox="431 709 1479 785">z/OS only. The task has been submitted to the z/OS Job Entry subsystem and scheduled by the z/OS Job Scheduler.</td> </tr> <tr> <td data-bbox="269 795 423 844">Action Required</td> <td data-bbox="431 795 1479 844">Manual tasks only. When a Manual task launches, it goes into Action Required status, meaning a user must perform some manual activity.</td> </tr> <tr> <td data-bbox="269 854 423 882">Started</td> <td data-bbox="431 854 1479 882">Agent-based and Manual tasks only. The task has started. For agent-based tasks, this means the agent has received the task.</td> </tr> <tr> <td data-bbox="269 892 423 919">Running</td> <td data-bbox="431 892 1479 919">All task types. The task is running. For agent-based tasks, the agent has started running the program.</td> </tr> <tr> <td data-bbox="269 930 423 1257">Running Problems</td> <td data-bbox="431 930 1479 1257"> Workflows only. One or more tasks within the workflow has one of the following statuses: <ul style="list-style-type: none"> • Held • Undeliverable • Running Problems (for sub-workflows) • Cancel Pending • In Doubt • Start Failure • Cancelled </td> </tr> <tr> <td data-bbox="269 1268 423 1365">In Doubt</td> <td data-bbox="431 1268 1479 1365">Agent-based tasks only. The agent is "in doubt" about the current status of the task instance. This may occur if an agent or agent connection goes down. In this case, the agent restarts and reviews its data about tasks in progress. If the agent finds a task still running, it resumes normal monitoring. If the agent cannot find the task, this usually indicates that the task completed, but the agent considers the task status to be "in doubt."</td> </tr> <tr> <td data-bbox="269 1375 423 1402">Start Failure</td> <td data-bbox="431 1375 1479 1402">All task types. The task was unable to start.</td> </tr> <tr> <td data-bbox="269 1413 423 1488">Confirmation Required</td> <td data-bbox="431 1413 1479 1488">z/OS only. If you make JCL changes and restart a z/OS task, Opwise will put the task into Confirmation Required status and prompt you for a confirmation. For detailed processing steps, see Rerunning a z/OS Task.</td> </tr> <tr> <td data-bbox="269 1499 423 1526">Cancelled</td> <td data-bbox="431 1499 1479 1526">All task types. The task was cancelled by a user.</td> </tr> <tr> <td data-bbox="269 1537 423 1564">Failed</td> <td data-bbox="431 1537 1479 1564">All task types. The task ran to a failure status.</td> </tr> <tr> <td data-bbox="269 1575 423 1602">Skipped</td> <td data-bbox="431 1575 1479 1602">All task types. The task was skipped by a user.</td> </tr> <tr> <td data-bbox="269 1612 423 1751">Finished</td> <td data-bbox="431 1612 1479 1751">All task types. The task was forced by the user to finish. The user may do this in cases where the task had "Cancelled" or "Failed" status, and the user needed to release other task instances depending on the successful completion of this task instance in a workflow. For more information, see Force Finishing a Task.</td> </tr> <tr> <td data-bbox="269 1761 423 1789">Success</td> <td data-bbox="431 1761 1479 1789">All task types. The task has completed successfully.</td> </tr> </tbody> </table>	Status	Description	Defined	All task types. The new task instance has been created (the task has been launched). Not yet implemented.	Waiting	All task types. The task has been loaded by a workflow and is waiting to run.	Held	All task types. The task has been put on hold by a user.	Resource Requested	All tasks with a virtual resource defined go immediately to a status of Resource Requested. If the resource is available, the task then moves to the next appropriate processing status.	Resource Wait	All tasks with a virtual resource defined go immediately to a status of Resource Requested. If the resource is not available, the task goes to a status of Resource Wait. When the resource becomes available, the task moves to the next appropriate processing status	Execution Wait	Agent-based tasks. The task must wait to be completed; either the Agent/Agent Cluster running the task has reached its Task Execution Limit, or the ability of the Agent/Agent Cluster to run tasks has been suspended.	Undeliverable	Agent-based tasks. The agent is unavailable.	Queued	Agent-based tasks only. The task has been queued on a resource.	Submitted	z/OS only. The task has been submitted to the z/OS Job Entry subsystem and scheduled by the z/OS Job Scheduler.	Action Required	Manual tasks only. When a Manual task launches, it goes into Action Required status, meaning a user must perform some manual activity.	Started	Agent-based and Manual tasks only. The task has started. For agent-based tasks, this means the agent has received the task.	Running	All task types. The task is running. For agent-based tasks, the agent has started running the program.	Running Problems	Workflows only. One or more tasks within the workflow has one of the following statuses: <ul style="list-style-type: none"> • Held • Undeliverable • Running Problems (for sub-workflows) • Cancel Pending • In Doubt • Start Failure • Cancelled 	In Doubt	Agent-based tasks only. The agent is "in doubt" about the current status of the task instance. This may occur if an agent or agent connection goes down. In this case, the agent restarts and reviews its data about tasks in progress. If the agent finds a task still running, it resumes normal monitoring. If the agent cannot find the task, this usually indicates that the task completed, but the agent considers the task status to be "in doubt."	Start Failure	All task types. The task was unable to start.	Confirmation Required	z/OS only. If you make JCL changes and restart a z/OS task, Opwise will put the task into Confirmation Required status and prompt you for a confirmation. For detailed processing steps, see Rerunning a z/OS Task .	Cancelled	All task types. The task was cancelled by a user.	Failed	All task types. The task ran to a failure status.	Skipped	All task types. The task was skipped by a user.	Finished	All task types. The task was forced by the user to finish. The user may do this in cases where the task had "Cancelled" or "Failed" status, and the user needed to release other task instances depending on the successful completion of this task instance in a workflow. For more information, see Force Finishing a Task .	Success	All task types. The task has completed successfully.
Status	Description																																												
Defined	All task types. The new task instance has been created (the task has been launched). Not yet implemented.																																												
Waiting	All task types. The task has been loaded by a workflow and is waiting to run.																																												
Held	All task types. The task has been put on hold by a user.																																												
Resource Requested	All tasks with a virtual resource defined go immediately to a status of Resource Requested. If the resource is available, the task then moves to the next appropriate processing status.																																												
Resource Wait	All tasks with a virtual resource defined go immediately to a status of Resource Requested. If the resource is not available, the task goes to a status of Resource Wait. When the resource becomes available, the task moves to the next appropriate processing status																																												
Execution Wait	Agent-based tasks. The task must wait to be completed; either the Agent/Agent Cluster running the task has reached its Task Execution Limit, or the ability of the Agent/Agent Cluster to run tasks has been suspended.																																												
Undeliverable	Agent-based tasks. The agent is unavailable.																																												
Queued	Agent-based tasks only. The task has been queued on a resource.																																												
Submitted	z/OS only. The task has been submitted to the z/OS Job Entry subsystem and scheduled by the z/OS Job Scheduler.																																												
Action Required	Manual tasks only. When a Manual task launches, it goes into Action Required status, meaning a user must perform some manual activity.																																												
Started	Agent-based and Manual tasks only. The task has started. For agent-based tasks, this means the agent has received the task.																																												
Running	All task types. The task is running. For agent-based tasks, the agent has started running the program.																																												
Running Problems	Workflows only. One or more tasks within the workflow has one of the following statuses: <ul style="list-style-type: none"> • Held • Undeliverable • Running Problems (for sub-workflows) • Cancel Pending • In Doubt • Start Failure • Cancelled 																																												
In Doubt	Agent-based tasks only. The agent is "in doubt" about the current status of the task instance. This may occur if an agent or agent connection goes down. In this case, the agent restarts and reviews its data about tasks in progress. If the agent finds a task still running, it resumes normal monitoring. If the agent cannot find the task, this usually indicates that the task completed, but the agent considers the task status to be "in doubt."																																												
Start Failure	All task types. The task was unable to start.																																												
Confirmation Required	z/OS only. If you make JCL changes and restart a z/OS task, Opwise will put the task into Confirmation Required status and prompt you for a confirmation. For detailed processing steps, see Rerunning a z/OS Task .																																												
Cancelled	All task types. The task was cancelled by a user.																																												
Failed	All task types. The task ran to a failure status.																																												
Skipped	All task types. The task was skipped by a user.																																												
Finished	All task types. The task was forced by the user to finish. The user may do this in cases where the task had "Cancelled" or "Failed" status, and the user needed to release other task instances depending on the successful completion of this task instance in a workflow. For more information, see Force Finishing a Task .																																												
Success	All task types. The task has completed successfully.																																												
Exit Codes	<p>Specifies one or more exit codes that will trigger the event. If you specify an exit code, you must also specify at least one status. Use commas to separate multiple exit codes; use a hyphen to specify a range. Example: 1, 5, 22-30.</p>																																												

On Late Start	Generate the action or notification if the task started late, based on the Late Start Time specified in the task.
On Late Finish	Generate the action or notification if the task finishes late, based on the Late Finish time specified in the task.
On Early Finish	Generate the action or notification if the task finishes early, based on the Early Finish Time specified in the task.
Description	Optional. Description of this Abort Action.
Cancel Process if Active	If enabled, instructs Opswise to cancel the process that was launched by this task before Force Finishing the task.
Override Exit Code	<p>Overrides the exit code returned by the process with the exit code specified in this field. This enables you to you Force Finish a task instance with a specific exit code so that you can force the workflow to take a conditional path using Conditions.</p> <div style="background-color: #ffffcc; padding: 10px; margin: 10px 0;"> <p> Note If you run the Abort Action against a task that has not yet started, the task will be skipped, and the Override Exit Code is not applicable.</p> </div>
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Delete button	Deletes the current record.

Setting Variable Actions

For information on how to set Variable Actions (for use within a workflow), see [Setting Variables within a Workflow](#) in the [Variables and Functions](#) section of this documentation.

Setting System Operations

- [Overview](#)
- [Adding a System Operation to a Task](#)
- [System Operation Field Descriptions](#)

Overview

A System Operation allows you to run an Opwise system operation based on specified conditions.

You can trigger the operation based on one or more of the following events associated with the task instance:

- Status or statuses of the task instance
- Exit code(s) generated by the program (along with at least one status)
- Late start
- Early or late finish

You can add one or more System Operations to any Opwise task. For Workflow tasks, you can also specify whether you want the System Operation instructions to apply to the workflow itself, the workflow an/or its tasks, or to the tasks only.

System Operations will run under the security context of the of the task instance Execution User, which must have the appropriate privileges for the specified Operation Type; otherwise, the System Operation will be prohibited.

Adding a System Operation to a Task

Step 1	Display the task to which you are adding the System Operation.
Step 2	Click the Actions tab. The Actions List screen displays a list of defined Actions for that task.
Step 3	Click New . The Action Wizard screen displays. <div data-bbox="224 1062 1373 1430" data-label="Image"> </div>

Step 4 Click **System Operation**. The System Operation screen displays.

Step 5 Using the field descriptions provided below as a guide, complete the fields as needed.

Step 6 Click the **Submit** button to save the record and return to the Actions list, or, right-click and select **Save** to save the record and remain on the current display.


Step 7 If appropriate, repeat these steps for any additional Actions you want to add.

System Operation Field Descriptions

The table below describes the fields and buttons on the System Operation screen.

Field Name	Description
Type Details	<p>Displays - on the Actions List screen - one of the following operation types for this action:</p> <ul style="list-style-type: none"> • Suspend Agent • Resume Agent • Suspend Agent Cluster • Resume Agent Cluster • Suspend Cluster Membership • Resume Cluster Membership • Set Agent Task Execution Limit • Set Cluster Task Execution Limit • Set Virtual Resource Limit
Action Inheritance	<p>Workflow tasks only. Specifies what records these instructions apply to. Options:</p> <ul style="list-style-type: none"> • SELF - These instructions apply only to the workflow and are not inherited by its children tasks. • SELF/CHILDREN - These instructions apply to the workflow and its contained tasks (children). • CHILDREN - These instructions apply only to the tasks within the workflow (children).

Status	<p>The status of this task. To trigger a System Operation, you can specify status only, or status and exit code. You can specify as many statuses as needed. Options:</p> <table border="1" data-bbox="269 210 1481 1801"> <thead> <tr> <th data-bbox="269 210 423 258">Status</th> <th data-bbox="431 210 1481 258">Description</th> </tr> </thead> <tbody> <tr> <td data-bbox="269 268 423 296">Defined</td> <td data-bbox="431 268 1481 296">All task types. The new task instance has been created (the task has been launched). Not yet implemented.</td> </tr> <tr> <td data-bbox="269 306 423 333">Waiting</td> <td data-bbox="431 306 1481 333">All task types. The task has been loaded by a workflow and is waiting to run.</td> </tr> <tr> <td data-bbox="269 344 423 371">Held</td> <td data-bbox="431 344 1481 371">All task types. The task has been put on hold by a user.</td> </tr> <tr> <td data-bbox="269 382 423 457">Resource Requested</td> <td data-bbox="431 382 1481 457">All tasks with a virtual resource defined go immediately to a status of Resource Requested. If the resource is available, the task then moves to the next appropriate processing status.</td> </tr> <tr> <td data-bbox="269 468 423 564">Resource Wait</td> <td data-bbox="431 468 1481 564">All tasks with a virtual resource defined go immediately to a status of Resource Requested. If the resource is not available, the task goes to a status of Resource Wait. When the resource becomes available, the task moves to the next appropriate processing status</td> </tr> <tr> <td data-bbox="269 575 423 623">Execution Wait</td> <td data-bbox="431 575 1481 623">Agent-based tasks. The task must wait to be completed; either the Agent/Agent Cluster running the task has reached its Task Execution Limit, or the ability of the Agent/Agent Cluster to run tasks has been suspended.</td> </tr> <tr> <td data-bbox="269 634 423 661">Undeliverable</td> <td data-bbox="431 634 1481 661">Agent-based tasks. The agent is unavailable.</td> </tr> <tr> <td data-bbox="269 672 423 699">Queued</td> <td data-bbox="431 672 1481 699">Agent-based tasks only. The task has been queued on a resource.</td> </tr> <tr> <td data-bbox="269 709 423 785">Submitted</td> <td data-bbox="431 709 1481 785">z/OS only. The task has been submitted to the z/OS Job Entry subsystem and scheduled by the z/OS Job Scheduler.</td> </tr> <tr> <td data-bbox="269 795 423 844">Action Required</td> <td data-bbox="431 795 1481 844">Manual tasks only. When a Manual task launches, it goes into Action Required status, meaning a user must perform some manual activity.</td> </tr> <tr> <td data-bbox="269 854 423 882">Started</td> <td data-bbox="431 854 1481 882">Agent-based and Manual tasks only. The task has started. For agent-based tasks, this means the agent has received the task.</td> </tr> <tr> <td data-bbox="269 892 423 919">Running</td> <td data-bbox="431 892 1481 919">All task types. The task is running. For agent-based tasks, the agent has started running the program.</td> </tr> <tr> <td data-bbox="269 930 423 1257">Running Problems</td> <td data-bbox="431 930 1481 1257"> Workflows only. One or more tasks within the workflow has one of the following statuses: <ul style="list-style-type: none"> • Held • Undeliverable • Running Problems (for sub-workflows) • Cancel Pending • In Doubt • Start Failure • Cancelled </td> </tr> <tr> <td data-bbox="269 1268 423 1365">In Doubt</td> <td data-bbox="431 1268 1481 1365">Agent-based tasks only. The agent is "in doubt" about the current status of the task instance. This may occur if an agent or agent connection goes down. In this case, the agent restarts and reviews its data about tasks in progress. If the agent finds a task still running, it resumes normal monitoring. If the agent cannot find the task, this usually indicates that the task completed, but the agent considers the task status to be "in doubt."</td> </tr> <tr> <td data-bbox="269 1375 423 1402">Start Failure</td> <td data-bbox="431 1375 1481 1402">All task types. The task was unable to start.</td> </tr> <tr> <td data-bbox="269 1413 423 1509">Confirmation Required</td> <td data-bbox="431 1413 1481 1509">z/OS only. If you make JCL changes and restart a z/OS task, Opwise will put the task into Confirmation Required status and prompt you for a confirmation. For detailed processing steps, see Rerunning a z/OS Task.</td> </tr> <tr> <td data-bbox="269 1520 423 1547">Cancelled</td> <td data-bbox="431 1520 1481 1547">All task types. The task was cancelled by a user.</td> </tr> <tr> <td data-bbox="269 1558 423 1585">Failed</td> <td data-bbox="431 1558 1481 1585">All task types. The task ran to a failure status.</td> </tr> <tr> <td data-bbox="269 1596 423 1623">Skipped</td> <td data-bbox="431 1596 1481 1623">All task types. The task was skipped by a user.</td> </tr> <tr> <td data-bbox="269 1633 423 1751">Finished</td> <td data-bbox="431 1633 1481 1751">All task types. The task was forced by the user to finish. The user may do this in cases where the task had "Cancelled" or "Failed" status, and the user needed to release other task instances depending on the successful completion of this task instance in a workflow. For more information, see Force Finishing a Task.</td> </tr> <tr> <td data-bbox="269 1761 423 1789">Success</td> <td data-bbox="431 1761 1481 1789">All task types. The task has completed successfully.</td> </tr> </tbody> </table>	Status	Description	Defined	All task types. The new task instance has been created (the task has been launched). Not yet implemented.	Waiting	All task types. The task has been loaded by a workflow and is waiting to run.	Held	All task types. The task has been put on hold by a user.	Resource Requested	All tasks with a virtual resource defined go immediately to a status of Resource Requested. If the resource is available, the task then moves to the next appropriate processing status.	Resource Wait	All tasks with a virtual resource defined go immediately to a status of Resource Requested. If the resource is not available, the task goes to a status of Resource Wait. When the resource becomes available, the task moves to the next appropriate processing status	Execution Wait	Agent-based tasks. The task must wait to be completed; either the Agent/Agent Cluster running the task has reached its Task Execution Limit, or the ability of the Agent/Agent Cluster to run tasks has been suspended.	Undeliverable	Agent-based tasks. The agent is unavailable.	Queued	Agent-based tasks only. The task has been queued on a resource.	Submitted	z/OS only. The task has been submitted to the z/OS Job Entry subsystem and scheduled by the z/OS Job Scheduler.	Action Required	Manual tasks only. When a Manual task launches, it goes into Action Required status, meaning a user must perform some manual activity.	Started	Agent-based and Manual tasks only. The task has started. For agent-based tasks, this means the agent has received the task.	Running	All task types. The task is running. For agent-based tasks, the agent has started running the program.	Running Problems	Workflows only. One or more tasks within the workflow has one of the following statuses: <ul style="list-style-type: none"> • Held • Undeliverable • Running Problems (for sub-workflows) • Cancel Pending • In Doubt • Start Failure • Cancelled 	In Doubt	Agent-based tasks only. The agent is "in doubt" about the current status of the task instance. This may occur if an agent or agent connection goes down. In this case, the agent restarts and reviews its data about tasks in progress. If the agent finds a task still running, it resumes normal monitoring. If the agent cannot find the task, this usually indicates that the task completed, but the agent considers the task status to be "in doubt."	Start Failure	All task types. The task was unable to start.	Confirmation Required	z/OS only. If you make JCL changes and restart a z/OS task, Opwise will put the task into Confirmation Required status and prompt you for a confirmation. For detailed processing steps, see Rerunning a z/OS Task .	Cancelled	All task types. The task was cancelled by a user.	Failed	All task types. The task ran to a failure status.	Skipped	All task types. The task was skipped by a user.	Finished	All task types. The task was forced by the user to finish. The user may do this in cases where the task had "Cancelled" or "Failed" status, and the user needed to release other task instances depending on the successful completion of this task instance in a workflow. For more information, see Force Finishing a Task .	Success	All task types. The task has completed successfully.
Status	Description																																												
Defined	All task types. The new task instance has been created (the task has been launched). Not yet implemented.																																												
Waiting	All task types. The task has been loaded by a workflow and is waiting to run.																																												
Held	All task types. The task has been put on hold by a user.																																												
Resource Requested	All tasks with a virtual resource defined go immediately to a status of Resource Requested. If the resource is available, the task then moves to the next appropriate processing status.																																												
Resource Wait	All tasks with a virtual resource defined go immediately to a status of Resource Requested. If the resource is not available, the task goes to a status of Resource Wait. When the resource becomes available, the task moves to the next appropriate processing status																																												
Execution Wait	Agent-based tasks. The task must wait to be completed; either the Agent/Agent Cluster running the task has reached its Task Execution Limit, or the ability of the Agent/Agent Cluster to run tasks has been suspended.																																												
Undeliverable	Agent-based tasks. The agent is unavailable.																																												
Queued	Agent-based tasks only. The task has been queued on a resource.																																												
Submitted	z/OS only. The task has been submitted to the z/OS Job Entry subsystem and scheduled by the z/OS Job Scheduler.																																												
Action Required	Manual tasks only. When a Manual task launches, it goes into Action Required status, meaning a user must perform some manual activity.																																												
Started	Agent-based and Manual tasks only. The task has started. For agent-based tasks, this means the agent has received the task.																																												
Running	All task types. The task is running. For agent-based tasks, the agent has started running the program.																																												
Running Problems	Workflows only. One or more tasks within the workflow has one of the following statuses: <ul style="list-style-type: none"> • Held • Undeliverable • Running Problems (for sub-workflows) • Cancel Pending • In Doubt • Start Failure • Cancelled 																																												
In Doubt	Agent-based tasks only. The agent is "in doubt" about the current status of the task instance. This may occur if an agent or agent connection goes down. In this case, the agent restarts and reviews its data about tasks in progress. If the agent finds a task still running, it resumes normal monitoring. If the agent cannot find the task, this usually indicates that the task completed, but the agent considers the task status to be "in doubt."																																												
Start Failure	All task types. The task was unable to start.																																												
Confirmation Required	z/OS only. If you make JCL changes and restart a z/OS task, Opwise will put the task into Confirmation Required status and prompt you for a confirmation. For detailed processing steps, see Rerunning a z/OS Task .																																												
Cancelled	All task types. The task was cancelled by a user.																																												
Failed	All task types. The task ran to a failure status.																																												
Skipped	All task types. The task was skipped by a user.																																												
Finished	All task types. The task was forced by the user to finish. The user may do this in cases where the task had "Cancelled" or "Failed" status, and the user needed to release other task instances depending on the successful completion of this task instance in a workflow. For more information, see Force Finishing a Task .																																												
Success	All task types. The task has completed successfully.																																												
Exit Codes	<p>Specifies one or more exit codes that will trigger the event. If you specify an exit code, you must also specify at least one status. Use commas to separate multiple exit codes; use a hyphen to specify a range. Example: 1, 5, 22-30.</p>																																												

On Late Start	Generate the action or notification if the task started late, based on the Late Start Time specified in the task.
On Late Finish	Generate the action or notification if the task finishes late, based on the Late Finish time specified in the task.
On Early Finish	Generate the action or notification if the task finishes early, based on the Early Finish Time specified in the task.
Description	Optional. Description of this System Operation.
System Operation	<p>Specific system operation to perform. Options:</p> <ul style="list-style-type: none"> • Suspend Agent • Resume Agent • Suspend Agent Cluster • Resume Agent Cluster • Suspend Cluster Membership • Resume Cluster Membership • Set Agent Task Execution Limit • Set Cluster Task Execution Limit • Set Virtual Resource Limit
System Notification	<p>Status of the specified system operation (see above) that will trigger a system notification. Options:</p> <ul style="list-style-type: none"> • None • Operation Failure (default) • Operation Success/Failure • Operation Success <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note The Opwise Controller must be configured for system notifications in order for system notifications to be triggered.</p> </div>
Agent	If System Operation is Suspend Agent, Resume Agent, Suspend Cluster Membership, Resume Cluster Membership, or Set Agent Task Execution Limit: Agent for which the system operation is to be performed.
Agent Variable	If System Operation is Suspend Agent, Resume Agent, Suspend Cluster Membership, Resume Cluster Membership, or Set Agent Task Execution Limit: Variable specifying an Agent for which the system operation is to be performed.
Agent Cluster	If System Operation is Suspend Agent Cluster, Resume Agent Cluster, Suspend Cluster Membership, Resume Cluster Membership, or Set Cluster Task Execution Limit: Agent Cluster for which the system operation is to be performed.
Agent Cluster Variable	If System Operation is Suspend Agent Cluster, Resume Agent Cluster, Suspend Cluster Membership, Resume Cluster Membership, or Set Cluster Task Execution Limit: Variable specifying an Agent Cluster for which the system operation is to be performed.
Task Execution Limit	If System Operation is Set Agent Task Execution Limit or Set Cluster Task Execution Limit: Specification for whether a Limited or Unlimited number of task instances can be run concurrently on the specified Agent / Agent Cluster. (Default is Unlimited.)
Virtual Resource	If System Operation is Set Virtual Resource Limit: Virtual resource for which a virtual resource limit is to be set.
Virtual Resource Variable	If System Operation is Set Virtual Resource Limit: Variable specifying the virtual resource for which a virtual resource limit is to be set.
Limit	<p>If System Operation is Set Agent Task Execution Limit or Set Cluster Task Execution Limit, and Task Execution Limit is Limited: Number of tasks that can be run concurrently by the specified Agent / Agent Cluster.</p> <p>If System Operation is Set Virtual Resource Limit: Virtual resource limit to be set for the specified virtual resource.</p>
Submit button	Submits the new record to the database.

Update button	Saves updates to the record.
Delete button	Deletes the current record.

Setting Mutually Exclusive Tasks

Setting Mutually Exclusive Tasks

You can set a task to be mutually exclusive with one or more other tasks. Opwise does not permit mutually exclusive tasks to run at the same time; if one is running, the other(s) will wait before running.

To set mutually exclusive tasks:

Step 1 Select a task from a Task List screen and, on the Task Definition screen for that task, click the Mutually Exclusive Tasks tab. The Mutually Exclusive Tasks List screen displays:

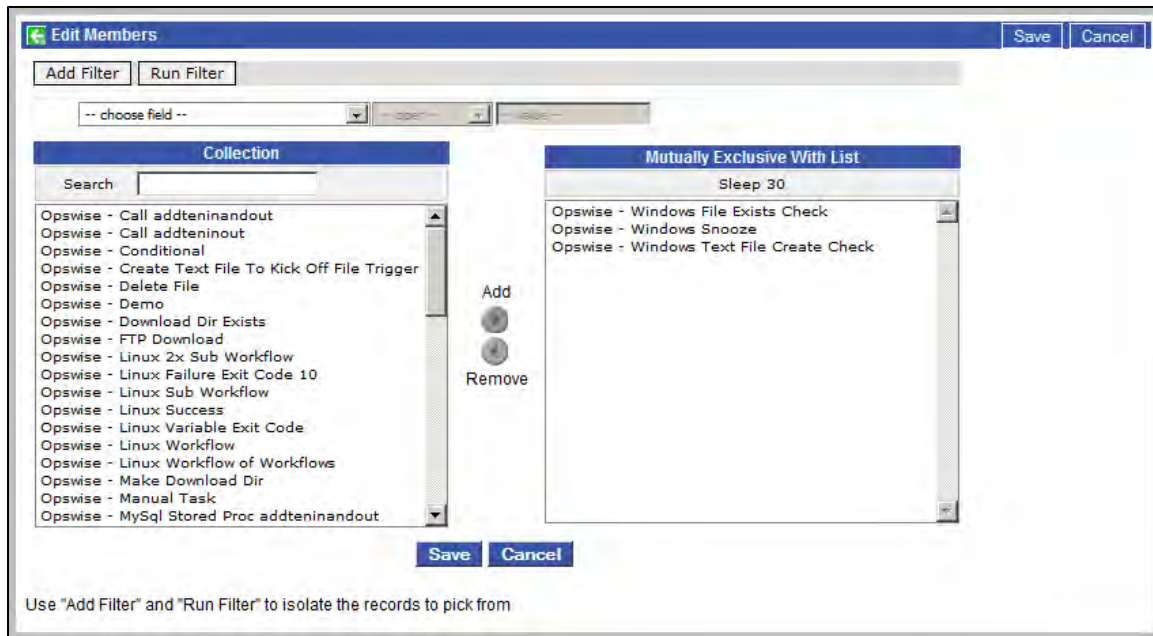
Exclusive Task	Type
Opwise - Windows Text File Create Check	Direct
Opwise - Windows File Exists Check	Direct
Opwise - Windows Snooze	Direct

- The **Exclusive Task** column identifies all tasks that are mutually exclusive with this task.
- The **Type** field indicates which task the mutually exclusive dependency was added to:
 - **Direct** indicates that the mutually exclusive dependency was added to this task manually.
 - **Indirect** indicates that the mutually exclusive dependency was added to this task automatically when this task was directly added as a mutually exclusive dependency to the exclusive task.



You only can delete Direct mutually exclusive tasks.

Step 2 Click **Edit...** to display the Edit Members dialog for this task.



- The Collection window displays all tasks that match any filter criteria you have selected. If you not select any filter criteria, all Opswise Automation Center tasks display.
- The Mutually Exclusive With List window displays all tasks that are to be run mutually exclusive with this task.

Step 3 Select one or more tasks in the Collection window and then click the Add arrow to move them to the Mutually Exclusive With List. These tasks will be mutually exclusive with this task. (Select one or more tasks in the Mutually Exclusive With List window and then click the Remove arrow to move them to the Collection window.)

Step 4 Click Save. All of the tasks in the Mutually Exclusive With List window will list this task (as Type Indirect) on their Mutually Exclusive Tasks List screen.

Creating Step Conditions

- Overview
 - Runtime Monitoring
- Creating a Step Condition
- Step Condition Field Descriptions
- Step Condition Logic
- Example Steps and Condition Codes
 - Example Job and Procedure
 - Opwise User Interface Specifications and Actions

Overview

A z/OS JES batch job consists of one or more steps defined by JCL EXEC statements. The JCL EXEC statement identifies the program that the step is to execute. During job execution, steps are executed sequentially under conditions defined by the JCL statements. When a step completes execution, a step condition code is recorded by JES. The step condition code is either an integer condition code, in the range of 0 - 4095, or an ABEND code. If a step does not execute, which can be for a number of reasons, it is referred to as FLUSH'ed.

A task's status of SUCCESS or FAILED is determined by task exit code processing. The [z/OS Task](#) definition Exit Code Processing field specifies the method used to determine the task status for a z/OS batch job. When the Step Conditions exit code processing method is selected, the task status of the z/OS batch job is controlled by the Step Conditions defined in the z/OS Task and parent workflow.

In addition to determining the z/OS Task status, step conditions provide a means to control the execution of job steps without any changes to the batch job JCL. A step condition definition can specify that job execution is halted, continued, or determined by a console operator. For example, if a multi-step job has a step that ends with a condition code of 8, you could include a step condition check to decide whether or not to run the following steps.

Step conditions can be applied at the z/OS Task level or at the [workflow](#) level that apply to all z/OS tasks in that workflow and sub-workflows.

Runtime Monitoring

You can monitor step conditions at runtime via the [Activity screen](#), which lets you add or change step conditions for a single task instance and then re-run that job.

Creating a Step Condition

Step 1 From the Navigation Pane, select **z/OS Tasks**. The z/OS Tasks List screen displays.

Step 2 Select the task for which you want to create one or more step conditions. The z/OS Task Definition screen displays.

Step 3 In the Exit Code Processing field, select Step Conditions from the drop-down list.

Step 4 Click the **Step Conditions** tab. The Step Conditions List screen displays.

Evaluation Order	Step	Procedure	Program	Condition Codes	Action
0	STEP3			8	Continue/Success
0	STEP2			16	Continue/Success
10				1-4095	Continue/Failed

Step 5 Click **New**. The Step Conditions screen displays.

Step 6 Using the field descriptions provided [below](#) as a guide, complete the fields as needed.

Step 7 Click the **Submit** button to save the record and return to the menu, or right-click on the title bar and select **Save** to save the record and remain on the current display.

Step 8 If appropriate, repeat these steps for any additional step conditions you want to add.

Step Condition Field Descriptions

The table below describes the fields and buttons on the Step Conditions screen.

Field Name	Description
Evaluation Order	The order in which the step conditions are searched. The search order is from the smallest value to the largest. The search order for equal values is not relevant and may be search in any order. Valid values are any negative or positive number.
Step	The job step name to match. A blank value or an asterisk (*) will match any job step name. Generic matching characters asterisk (*) and question mark (?) match zero or more characters and one character, respectively.
Procedure	The procedure step name to match. A blank value or an asterisk (*) will match any procedure step name. Generic matching characters asterisk (*) and question mark (?) match zero or more characters and one character, respectively.
Program	The program name to match. A blank value or an asterisk (*) will match any program name. Generic matching characters asterisk (*) and question mark (?) match zero or more characters and one character, respectively.
Condition Codes	Conditions codes are integer return codes from the program or ABEND codes. Integer return codes are specified as a comma-separated list of integer values or ranges. Ranges are specified with a dash (-) separating the lower and upper bounds of the range. The z/OS job step return code range is 0-4095. ABEND codes are specified directly as either a user ABEND or a system ABEND. The ABEND code must be specified verbatim including leading zeroes.
Action	The action to take and the task status to set if the step condition matches. Refer to step condition logic , below, for an explanation of the actions.

Step Condition Logic

Step Condition exit code processing starts the task with a task status of SUCCESS. As the job executes and steps complete, the task status can change from SUCCESS to FAILED based on step condition definitions and job execution conditions. Once a task status has been changed to FAILED, it cannot be changed back to SUCCESS.

In addition to step condition definitions changing the task status, the following specific job execution conditions will change the task status:

- JCL errors (e.g. IEF452I or IEF453I) change the task status to FAILED.
- A job step ABEND that does not match any step condition definition changes the task status to FAILED.

As job steps complete execution, Opwise searches the list of task-level step condition definitions that matches the current step based on the job step name, procedure step name, program name, and the step condition code. The search stops when the first definition is found. If a matching step condition is found, the step condition action is taken. If no matching task-level step condition is found, the search continues with the parent workflow-level step conditions. If no matching workflow-level step condition is found, the search continues with its parent workflow-level step conditions and so on until a match is found or all step conditions have been search in the hierarchy. If no matching step condition is found, Opwise takes no action and normal JES processing of the job continues.

Note that if a step does not execute, no search is performed for that step in the step condition definitions. For example, if a job step FLUSH'es due to a JCL IF statement, the step conditions will not be search for the step.

Opwise searches step condition definitions based on the step condition evaluation order within the current task or workflow level. For step conditions that have the same evaluation order, the search order is not relevant and may be searched in any order. The default evaluation order is

0. The evaluation order value may be any negative or positive integer. The lower the value the higher the precedence in the search. For example, step condition definition with evaluation order -10 is searched before a definition with a value of 20.

The step condition definition action value specifies two attributes, the action to take and the task status. These two attributes are combined into combinations that form the possible action values. The following step condition actions are supported:

Continue/Success	Job execution continues and task status is set to SUCCESS.
Continue/Failed	Job execution continues and task status is set to FAILED.
Halt/Failed	Job execution is halted at the current step and task status is set to FAILED.
Askoper	Job execution is stopped and Opwise sends a WTOR message to the console operator requesting a reply on how job execution should proceed. The action is dependent upon the operator reply (see Example 4 , below).

During job processing, Opwise issues message UAG1059A to the job log when it matches a step condition definition to a step that has completed execution. Message UAG1059A includes the step condition definition values including the action that is taken. The message provides an audit record of step condition processing that has influenced job execution.

Example Steps and Condition Codes

This section provides a sample job and PROC, followed by example condition code checks for that job.

Example Job and Procedure

Example Job

```
//JOBA JOB ...
//S1 EXEC ACCTBL10
```

Example Procedure (Cataloged Procedure)

```
//ACCTBL10 PROC
//STEP1 EXEC PGM=BALANCE
//STEP2 EXEC PGM=MERGE
//STEP3 EXEC PGM=IEBGENER
// PEND
```

Opwise User Interface Specifications and Actions

The following examples specify condition code checks for the example job above.

Example 1

In this example, if the condition code of any step of the job is greater than 12, the job halts and the task status is set to FAILED.

Example 2

Step Condition = Required field	
Evaluation Order:	1
Step:	S1
Procedure:	
Program:	
Condition Codes:	8
Action:	Continue/Success
Submit	

In this example, if the condition code of any procedure step executed as job step S1 is equal to 8, the job continues and the task status is set to SUCCESS.

Example 3

Step Condition = Required field	
Evaluation Order:	2
Step:	
Procedure:	
Program:	IEBGENER
Condition Codes:	0,12
Action:	Continue/Success
Submit	

In this example, if the condition code of program IEBGENER is 0 or 12, the job continues and the task status is set to SUCCESS.

Example 4

Step Condition = Required field	
Evaluation Order:	3
Step:	S1
Procedure:	STEP2
Program:	
Condition Codes:	U0010
Action:	Askoper
Submit	

In this example, if the condition code from job step S1, procedure step STEP2 is user ABEND U0010, the operator is alerted with a WTOR console message that specifies the job name, the job step, the procedure step, and the actual condition code. Opwise will take the action specified by the operator reply.

Issued WTOR

```
UAG1058A JOBA      ,S1      ,STEP2      ,Code: U0010 Reply 1:CONT/SUCCESS, 2:CONT/FAIL, 3:HALT/FAIL
```

The UAG1058A WTOR message identifies the job name as JOBA, step name as S1, procedure step name as STEP2, and the step condition code as U0010 that matched the step condition definition which resulted in the ASKOPER action.

Operator Reply

The operator must reply with one of the following:

- (1) CONTINUE/SUCCESS
- (2) CONTINUE/FAILED
- (3) HALT/FAILED

(See [Step Condition Logic](#) for an explanation of these replies.)

Example 5

Step Condition = Required field	
Evaluation Order:	4
Step:	S1
Procedure:	STEP3
Program:	
Condition Codes:	0-7
Action:	Continue/Success
Submit	

In this example, if the condition code from job step S1, procedure step STEP3 is within the range of 0-7, the job continues and the task status is set to SUCCESS.

Example 6

Step Condition = Required field	
Evaluation Order:	5
Step:	S1
Procedure:	STEP1
Program:	
Condition Codes:	1-4095
Action:	Continue/Failed
Submit	

In this example, if the condition code from job step S1, procedure step STEP1 is greater than 0, the job continues and the task status is set to FAILED.

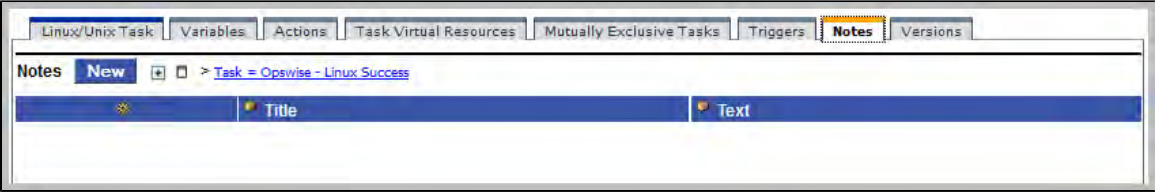
Creating Notes

- Introduction
- Creating a Note
- Note Field Descriptions
- Deleting a Note
- Viewing a Note
- Creating / Viewing a Note from a Task Instance Screen

Introduction

You can create a note for any [task](#) or [script](#) in the Opwise system. The note can consist of information needed by operations personnel or other instructions or tips.

Creating a Note

Step 1	Open the task or script for which you want to create a note.
Step 2	Click on the Notes tab. The Notes list screen displays a list of notes (if any) that have been created for this record.
	
Step 3	Click the New button. The Note definition screen displays.
	
Step 4	Type in the Title and Text.
Step 5	Click the Submit button.

Note Field Descriptions

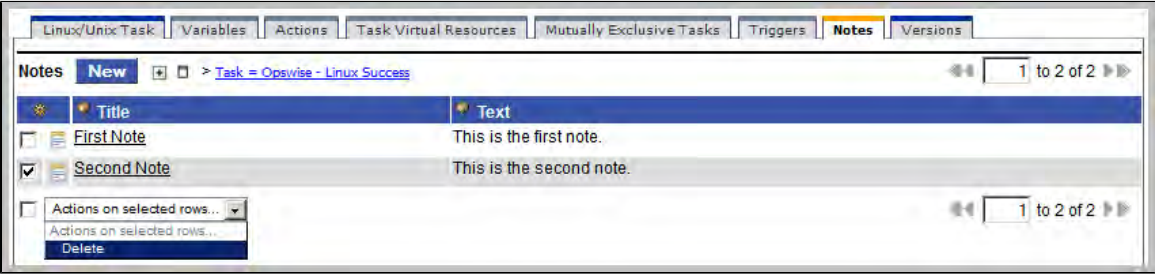
Field Name	Description
Title	Title of this note. Displays in the Title column on the Notes list.

Text	Text of the note.
Updated by	User who last updated this record.
Updated	Date and time this record was last updated.
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Delete button	Deletes the current record.

Deleting a Note

Display the note that you want to delete and click the **Delete** button, or you can delete one or more notes from the Notes list screen:

Step 1 On the Notes list screen, click on the box associated with the note or notes you want to delete.



Step 2 From the **Actions on selected rows** menu, select **Delete**.

Viewing a Note

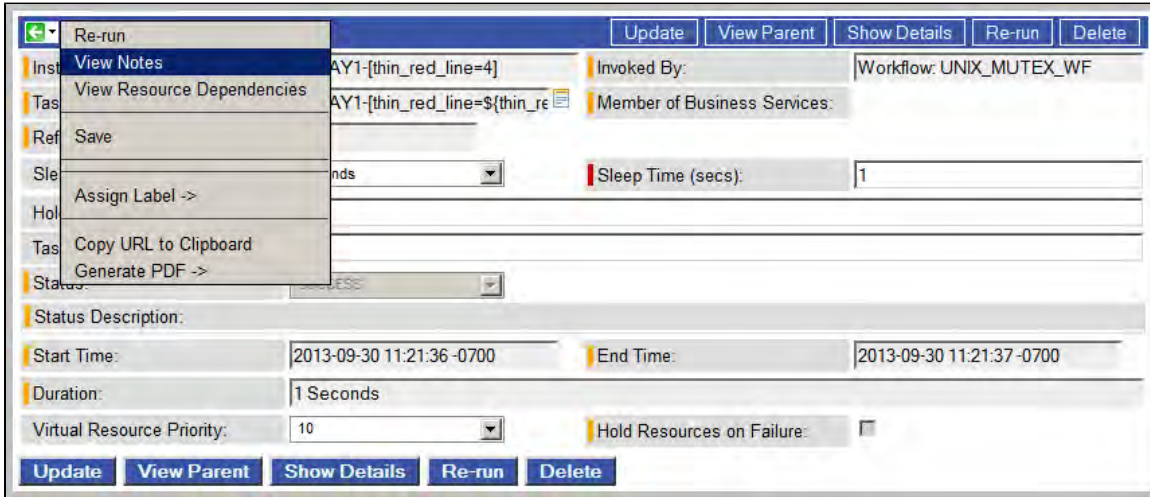
- Step 1** From the Notes list, scroll to the note you want to read.
- Step 2** Click the underlined field displayed in the leftmost column. Opswise displays the contents of the note.

Creating / Viewing a Note from a Task Instance Screen

You also can create and view notes for a task from a task instance screen. Notes created from a task instance screen apply to the task, not to that task instance.

- Step 1** From the **Activity** screen or **Task Instances** screen, select a task instance.

Step 2 On the task instance screen, either hover your cursor over the down arrow on the title bar or right-click the title bar, and then click **View Notes**.



The Notes list screen for this task will display.

Step 3 On Notes list screen, either:

- Click the **New** button to create a note.
- Select a note to view.

Manually Running and Controlling Tasks

- Overview
- Finding Tasks within a Workflow
- Inserting Tasks within a Workflow
- Issuing Commands Against Task Instances
 - Issuing Commands from the Activity Screen
 - Issuing Commands from the Task Instances Screen
 - Issuing Commands from the Workflow Monitor
- Commands Supported for Task Instance Statuses
- Manually Launching a Task
 - Launch One or More Tasks from the Tasks List Screen
 - Launch a Task from the Task Definition Screen
 - Provide Temporary Variable Values and Launch a Task Manually
- Changing the Priority of a Task Instance
 - Set Priority on a Task Instance from the Activity or Task Instances Screen
 - Set Priority on a Task Instance from the Workflow Monitor
- Re-running a Task Instance
 - Re-run a Task Instance from the Activity or Task Instances Screen
 - Re-run a Task Instance from the Task Instance Screen
 - Re-run a Task Instance from the Workflow Monitor
- Cancelling a Task Instance
 - Cancel a Task Instance from the Activity or Task Instances Screen
 - Cancel a Task Instance from the Workflow Monitor
- Force Finishing a Task Instance
 - Force Finish a Task Instance from the Activity or Task Instances Screen
 - Force Finish a Task Instance from the Workflow Monitor
- Force Finish/Cancelling a Task Instance
 - Force Finish/Cancel a Task Instance from the Activity or Task Instances Screen
 - Force Finish/Cancel a Task Instance from the Workflow Monitor
- Putting a Task Instance on Hold
 - Hold a Task Instance from the Activity or Task Instances Screen
 - Hold a Task Instance from the Workflow Monitor
- Releasing a Task Instance from Hold
 - Release a Held Task Instance from the Activity or Task Instances Screen
 - Release a Held Task Instance from the Workflow Monitor
- Skipping a Task Instance
 - Skip a Task Instance from the Activity Screen or Task Instances Screen
 - Skip a Task Instance from the Workflow Monitor
- Showing or Hiding Skipped Task Instances
 - Setting the Show / Hide Skipped Tasks Workflow Definition Option
 - Setting the Show / Hide Skipped Tasks Workflow Instance Option
 - Show / Hide Skipped Tasks from the Workflow Monitor
- Unskipping a Task Instance
 - Unskip a Task Instance from the Activity or Task Instances Screen
 - Unskip a Task Instance from the Workflow Monitor
- Clearing All Predecessor Dependencies for a Task Instance
 - Clear Dependencies on a Task Instance from the Activity or Task Instances Screen
 - Clear Dependencies on a Task Instance from the Workflow Monitor
- Marking a Dependency as Satisfied
- Clearing Resource Dependencies from Tasks
 - Clear Resource Dependencies of a Task Instance from the Activity or Task Instances Screen
 - Clear Resource Dependencies of a Task Instance from the Workflow Monitor
- Clearing Mutually Exclusive Dependencies from Tasks
 - Clear Mutually Exclusive Dependencies from a Task Instance from the Activity or Task Instances Screen
 - Clear Mutually Exclusive Dependencies from a Task Instance from the Workflow Monitor

Overview

A number of options are available on the Activity screen that allow you to intervene in task processing where needed. Some commands are applicable only to certain task types and others are appropriate only when the task is in a particular status. In addition, commands require appropriate permissions.

Finding Tasks within a Workflow

For any workflow task, or any workflow task instance, you can find the location of any task/task instance within the workflow.

Inserting Tasks within a Workflow

After a workflow has been launched, you can insert a new task (except a workflow task) into the active workflow instance. You can insert the task as a predecessor or successor of another task instance within the workflow instance using the [Insert Task as Predecessor](#) and [Insert Task as Successor](#) commands, respectively.

Alternatively, you can use the [Insert Task...](#) command to insert a task with any number of successors and predecessors.

Issuing Commands Against Task Instances

You can issue commands against task instances from the [Activity screen](#), the [Task Instances screen](#) (and the [Task Instances screen for a specific task](#)), and the [Workflow Monitor](#).

See [Commands Supported for Task Instance Statuses](#) for a list of task instances (and their statuses) for which these commands can be issued.

Command	Description
Cancel	Cancels a running task instance (see Cancelling a Task Instance).
Clear Dependencies	Workflow tasks only: Clears predecessor (upstream) dependencies of a task instance (see Clearing All Predecessor Dependencies for a Task Instance).
Clear Exclusive	Clears mutually exclusive dependencies from a task instance (see Clearing Mutually Exclusive Dependencies from Tasks).
Force Finish	Places a task instance into the Finished status (see Force Finishing a Task Instance).
Force Finish/Cancel	Cancels a task and places it into the Finished status (see Force Finish/Cancelling a Task Instance).
Hold	Temporarily prevents a task instance from running (see Putting a Task Instance on Hold).
Release	Removes a task instance from being on Hold (see Releasing a Task Instance from Hold).
Release Recursive	Workflow tasks only: Removes a workflow and its task instance from being on Hold (see Releasing a Task Instance from Hold).
Re-run	Not applicable for Workflow tasks: Re-runs task instance (see Re-running a Task Instance).
Set Completed	Sets a Manual Task instance to the Success status.
Set Started	Resets the Started Time of a Manual Task instance.
Skip	Disregards a task instance (see Skipping a Task Instance).
Skip Path	Disregards a task instance and all of its dependent task instances (see Skipping a Task Instance).
Unskip	Removes the Skip status from a task instance (see Unskipping a Task Instance).

Issuing Commands from the Activity Screen

Today's Task Instances by Created Time						New Report	Edit Report
Instance Name	Status	Start Time	End Time	Duration	Invoked By		
Print DR/CR on Collections	Running	2008-10-19 09:04:21 -0700				Trigger: Run International Tickets	
Check for DR/CR on Collections	Running	2008-10-19 09:04:21 -0700				Trigger: Run International Tickets	
Create a txt file to kick off the fi...	Undeliverable					Trigger: Run International Tickets	
Send File Create/Delete Email	Success	2008-10-19 09:04:21 -0700	2008-10-19 09:04:23 -0700	2 Seconds		Trigger: Run International Tickets	
Standby LCs	Running	2008-10-19 09:04:21 -0700				Trigger: Run International Tickets	
Fees Outgoing Collections	Success	2008-10-19 09:04:21 -0700	2008-10-19 09:04:36 -0700	15 Seconds		Trigger: Run International Tickets	
Welcome Msg Exists	Undeliverable					Trigger: Run International Tickets	
Fees Assessment Collections	Running	2008-10-19 09:04:21 -0700				Trigger: Run International Tickets	
Print DR/CR on Collections	Success	2008-10-19 09:03:21 -0700	2008-10-19 09:04:21 -0700	1 Minute 0 Seconds		Trigger: Run International Tickets	
Check for DR/CR on Collections	Success	2008-10-19 09:03:21 -0700	2008-10-19 09:04:21 -0700	1 Minute 0 Seconds		Trigger: Run International Tickets	
Create a txt file to kick off the fi...	Undeliverable					Trigger: Run International Tickets	
Send File Create/Delete Email	Success	2008-10-19 09:03:21 -0700	2008-10-19 09:03:23 -0700	2 Seconds		Trigger: Run International Tickets	
Standby LCs	Success	2008-10-19 09:03:21 -0700	2008-10-19 09:04:21 -0700	1 Minute 0 Seconds		Trigger: Run International Tickets	
Fees Outgoing Collections	Success	2008-10-19 09:03:21 -0700	2008-10-19 09:03:36 -0700	15 Seconds		Trigger: Run International Tickets	
Welcome Msg Exists	Undeliverable					Trigger: Run International Tickets	
Fees Assessment Collections	Success	2008-10-19 09:03:21 -0700	2008-10-19 09:04:21 -0700	1 Minute 0 Seconds		Trigger: Run International Tickets	
Print DR/CR on Collections	Success	2008-10-19 09:02:21 -0700	2008-10-19 09:03:21 -0700	1 Minute 0 Seconds		Trigger: Run International Tickets	
Check for DR/CR on Collections	Success	2008-10-19 09:02:21 -0700	2008-10-19 09:03:21 -0700	1 Minute 0 Seconds		Trigger: Run International Tickets	
Create a txt file to kick off the fi...	Undeliverable					Trigger: Run International Tickets	
Send File Create/Delete Email	Success	2008-10-19 09:02:21 -0700	2008-10-19 09:02:23 -0700	2 Seconds		Trigger: Run International Tickets	
Standby LCs	Success	2008-10-19 09:02:21 -0700	2008-10-19 09:03:21 -0700	1 Minute 0 Seconds		Trigger: Run International Tickets	
Fees Outgoing Collections	Success	2008-10-19 09:02:21 -0700	2008-10-19 09:02:36 -0700	15 Seconds		Trigger: Run International Tickets	
Welcome Msg Exists	Undeliverable					Trigger: Run International Tickets	
Fees Assessment Collections	Success	2008-10-19 09:02:21 -0700	2008-10-19 09:03:21 -0700	1 Minute 0 Seconds		Trigger: Run International Tickets	
Print DR/CR on Collections	Success	2008-10-19 09:01:21 -0700	2008-10-19 09:02:21 -0700	1 Minute 0 Seconds		Trigger: Run International Tickets	

Page 1 of 14 | 1 - 25 of 337

Grid Refresh Rate: 5 | Set Rate

Issue a Command Against a Single Task Instance

Step 1	Right-click an Instance Name to display a menu of available commands for that instance.
Step 2	Click an Instance Name to display the Task Instance screen for that instance and then either: <ul style="list-style-type: none"> • Click a command button at the bottom of the screen. • Right-click the Task Instance screen title bar to display a menu of available commands. • Move your cursor over the down arrow on the Task Instance screen title bar to display a menu of available commands.

Issue a Command Against Multiple Task Instances

Step 1	Press Ctrl and click on the each task instance that you want to issue a command against. You can click in any column, but do not click on the Instance Name , which is a hyperlink that opens the Task Instance screen for that instance in a new tab.
Step 2	When you have selected all the task instances that you want, right-click to display a menu of commands that are valid for all of the selected task instances.

Issuing Commands from the Task Instances Screen

Task Instances > Updated on Last 7 days 20 per page

Task Instances Go to 17154 to 171559 of 171579

Instance Name	Reference Id	Type	Status	Invoked By	Agent	Start Time	End Time
<input type="checkbox"/> Opswise - Linux 2x Sub Workflow	904	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:42:00 -0700	2013-09-16 13:42:07 -0700
<input type="checkbox"/> Opswise - Linux 2x Sub Workflow	905	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:43:00 -0700	2013-09-16 13:43:07 -0700
<input type="checkbox"/> Opswise - Linux 2x Sub Workflow	906	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:44:00 -0700	2013-09-16 13:44:07 -0700
<input type="checkbox"/> Opswise - Linux 2x Sub Workflow	907	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:45:00 -0700	2013-09-16 13:45:07 -0700
<input type="checkbox"/> Opswise - Linux 2x Sub Workflow	908	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:46:00 -0700	2013-09-16 13:46:07 -0700
<input type="checkbox"/> Opswise - Linux 2x Sub Workflow	909	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:47:00 -0700	2013-09-16 13:47:07 -0700
<input type="checkbox"/> Opswise - Linux 2x Sub Workflow	910	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:48:00 -0700	2013-09-16 13:48:07 -0700
<input type="checkbox"/> Opswise - Linux 2x Sub Workflow	911	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:49:00 -0700	2013-09-16 13:49:07 -0700
<input type="checkbox"/> Opswise - Linux 2x Sub Workflow	912	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:50:00 -0700	2013-09-16 13:50:07 -0700
<input type="checkbox"/> Opswise - Linux 2x Sub Workflow	913	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:51:00 -0700	2013-09-16 13:51:07 -0700
<input type="checkbox"/> Opswise - Linux 2x Sub Workflow	914	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:52:00 -0700	2013-09-16 13:52:07 -0700
<input type="checkbox"/> Opswise - Linux 2x Sub Workflow	915	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:53:00 -0700	2013-09-16 13:53:07 -0700
<input type="checkbox"/> Opswise - Linux 2x Sub Workflow	916	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:54:00 -0700	2013-09-16 13:54:07 -0700
<input type="checkbox"/> Opswise - Linux 2x Sub Workflow	917	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:55:00 -0700	2013-09-16 13:55:07 -0700
<input type="checkbox"/> Opswise - Linux 2x Sub Workflow	918	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:56:00 -0700	2013-09-16 13:56:07 -0700
<input type="checkbox"/> Opswise - Linux 2x Sub Workflow	919	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:57:00 -0700	2013-09-16 13:57:07 -0700
<input type="checkbox"/> Opswise - Linux 2x Sub Workflow	920	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:58:00 -0700	2013-09-16 13:58:07 -0700
<input type="checkbox"/> Opswise - Linux 2x Sub Workflow	921	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:59:00 -0700	2013-09-16 13:59:07 -0700
<input type="checkbox"/> Opswise - Linux 2x Sub Workflow	922	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 14:00:00 -0700	2013-09-16 14:00:07 -0700
<input type="checkbox"/> Opswise - Linux 2x Sub Workflow	923	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 14:01:00 -0700	2013-09-16 14:01:07 -0700

Actions on selected rows... 141 to 160 of 171,603

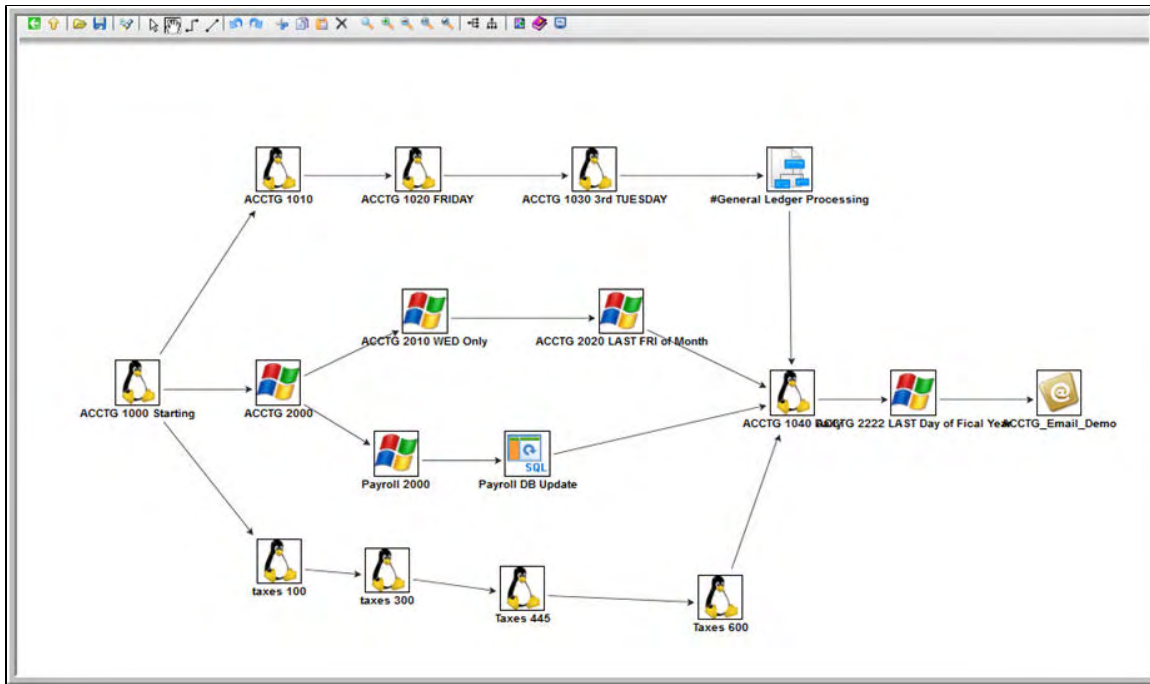
Issue a Command Against a Single Task Instance

Step 1	Click the check box next to an Instance Name and then select a command from the drop-down list at the bottom of the screen.
Step 2	Right-click an Instance Name to display a menu of available commands for that instance.
Step 3	Click an Instance Name to display the Task Instance screen for that instance and then either: <ul style="list-style-type: none"> • Click a command button at the bottom of the screen. • Right-click the Task Instance screen title bar to display a menu of available commands. • Move your cursor over the down arrow on the Task Instance screen title bar to display a menu of available commands.

Issue a Command Against Multiple Task Instances

Step 1	Press Ctrl and click the check box next to Instance Name for each task instance that you want to issue a command against.
Step 2	When you have selected all of the task instances that you want, select a command from the drop-down list at the bottom of the screen.

Issuing Commands from the Workflow Monitor



Issue a Command Against a Single Task Instance

- Step 1** Right-click the task instance to display a pop-up menu of tasks relevant to the selected task instance.
- Step 2** Click **Commands** and then click the command that you want to issue against the selected task instance.

Commands Supported for Task Instance Statuses

The following table identifies all possible task instance statuses, the task types they are valid for, and the commands that you can issue against a task instance in each status.

For a description of each status, see [Task Instance Status Types](#).

For a description of each command, see [Issuing Commands Against Task Instances](#).

For details and instructions on issuing these commands, see the specific section (below) on this page.

Status	Task Type	Supported Commands
Action Required	Manual	<ul style="list-style-type: none"> • Cancel • Force Finish • Force Finish/Cancel • Set Started • Set Completed
Cancel Pending	Agent-based*, SQL, and Stored Procedure	<ul style="list-style-type: none"> • Force Finish
Cancelled	All	<ul style="list-style-type: none"> • Force Finish • Re-run - Not applicable for Workflow tasks.
Confirmation Required	z/OS	<ul style="list-style-type: none"> • Force Finish • Re-run - Not applicable for Workflow tasks.

Defined	All	<ul style="list-style-type: none"> • Clear Dependencies • Force Finish • Hold • Skip • Skip Path • Release Recursive - Workflow tasks only.
Exclusive Requested	All	<ul style="list-style-type: none"> • Force Finish • Hold • Skip • Skip Path
Exclusive Wait	All	<ul style="list-style-type: none"> • Clear Exclusive • Force Finish • Hold • Skip • Skip Path • Release Recursive - Workflow tasks only.
Execution Wait	Agent-based*	<ul style="list-style-type: none"> • Force Finish • Hold • Skip • Skip Path
Failed	All	<ul style="list-style-type: none"> • Force Finish • Re-run - Not applicable for Workflow tasks.
Finished	All	<ul style="list-style-type: none"> • Re-run - Not applicable for Workflow tasks.
Held	All	<ul style="list-style-type: none"> • Clear Dependencies • Force Finish • Release • Skip • Skip Path • Release Recursive - Workflow tasks only.
In Doubt	Agent-based*	<ul style="list-style-type: none"> • Force Finish
Queued	Agent-based*	<ul style="list-style-type: none"> • Cancel • Force Finish • Hold
Resource Requested	All tasks using Virtual Resources	<ul style="list-style-type: none"> • Force Finish • Hold • Skip • Skip Path
Resource Wait	All tasks using Virtual Resources	<ul style="list-style-type: none"> • Force Finish • Hold • Skip • Skip Path • Release Recursive - Workflow tasks only.

Running	All	<ul style="list-style-type: none"> • Cancel • Force Finish • Force Finish/Cancel • Release Recursive - Workflow tasks only.
Running Problems	Workflow	<ul style="list-style-type: none"> • Cancel • Force Finish • Force Finish/Cancel • Hold • Release Recursive - Workflow tasks only.
Skipped	All	<ul style="list-style-type: none"> • Unskip
Start Failure	All	<ul style="list-style-type: none"> • Force Finish • Re-run - Not applicable for Workflow tasks.
Started	Agent-based* and Manual	<ul style="list-style-type: none"> • Cancel • Force Finish • Force Finish/Cancel • Set Completed - Manual tasks only.
Submitted	z/OS	<ul style="list-style-type: none"> • Force Finish
Success	All	<ul style="list-style-type: none"> • Re-run - Not applicable for Workflow tasks.
Undeliverable	Agent-based*	<ul style="list-style-type: none"> • Force Finish • Hold • Skip • Skip Path
Waiting	All	<ul style="list-style-type: none"> • Clear Dependencies • Force Finish • Hold • Skip • Skip Path • Release Recursive - Workflow tasks only

* Agent-based task types are Linux/Unix, Windows, z/OS, Indesca, SAP, File Transfer, File Monitor, FTP File Monitor, and System Monitor.

Manually Launching a Task

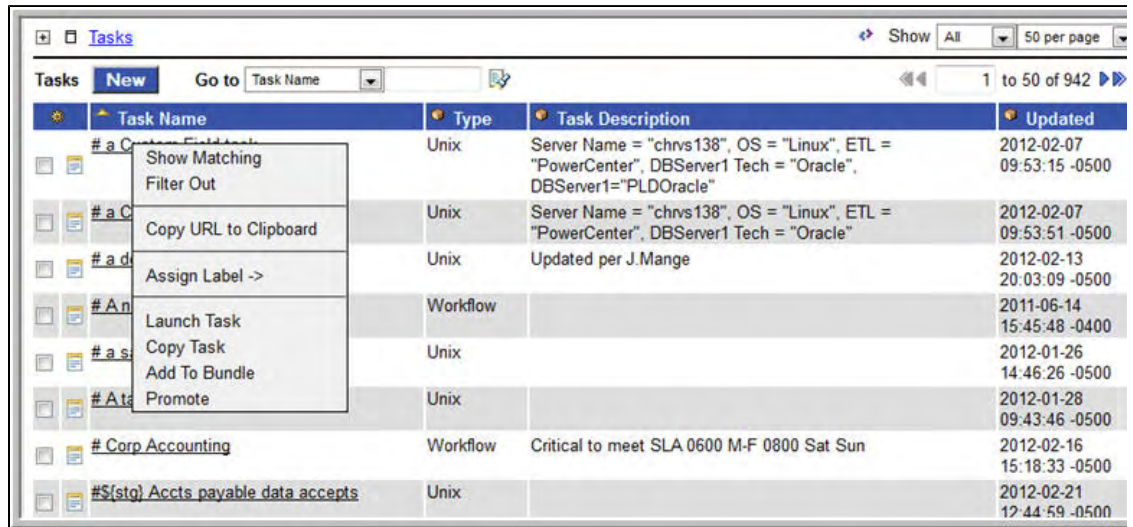
You can manually launch one or more tasks from a tasks list screen or a single task from a task definition screen.

Launch One or More Tasks from the Tasks List Screen

Step 1 Select **Automation Center > Tasks > <type of tasks>** and follow one of the procedures below:

- **To launch a single task:**

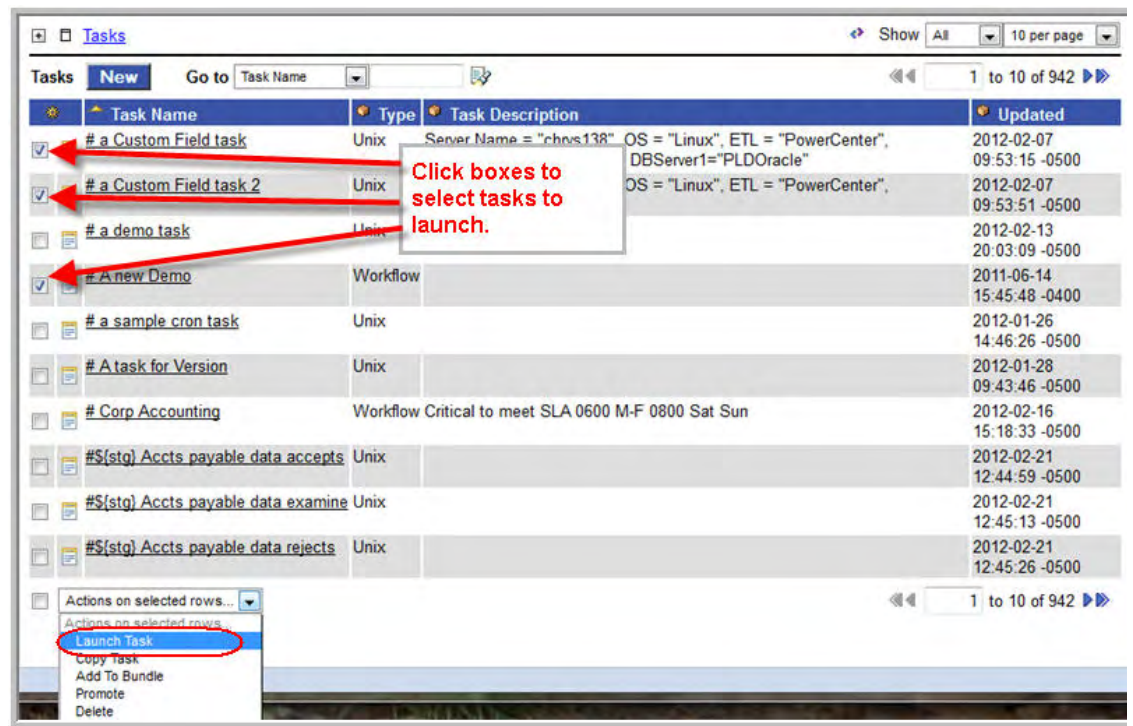
1. Right-click on the task you want to run. The right-click menu displays.



2. Select **Launch Task**. Opwise creates an instance of the task and runs it.

- **To launch one or more tasks:**

1. For each task you want to launch, click the box to the left of the task.
2. From the **Actions on selected rows...** menu at the bottom of the list, select **Launch Task**.



Step 2 To view details about running task instances, select **Automation Center > Activity** and click on the task instance.

Launch a Task from the Task Definition Screen

Step 1 Select the task you want to launch.

Step 2 Click the **Launch Task** button.

The screenshot displays the configuration page for an SQL Task in the Opwise Automation Center. The task is named "Opwise - SQL Create" and is version 1. It is configured to use the "Opwise - Db Connection" database connection. The SQL command to be executed is "CREATE TABLE opwise_demo (name varchar(128), value varchar(128));". The user estimated duration is set to 00:00:00. The task description is empty. The result processing is set to "Skip Result Processing". The auto cleanup is checked. The maximum rows is empty. The late start, late finish, and early finish options are unchecked. The maximum retries is 0, and the retry interval is 60 seconds. The virtual resource priority is 10. The retry indefinitely and hold resources on failure options are unchecked. The interface includes tabs for Variables, Actions, Task Virtual Resources, Mutually Exclusive Tasks, Triggers, Notes, and Versions. Buttons for Update, Launch Task, View Instances, and Delete are visible at the bottom.

Provide Temporary Variable Values and Launch a Task Manually

Opwise supports a Launch with Variables feature that allows you to quickly provide values for the variables specified in the task and launch it. All task types support the Launch with Variables feature.

To launch a task using Launch with Variables:

Step 1 Display the task you want to launch. This procedure assumes the task is already set up with variables where required.

Step 2 Open the task menu by clicking the down arrow or right-clicking in the title bar. The following menu appears.

The screenshot displays the 'Email Task' configuration interface. At the top, there are tabs for 'Email Task', 'Variables', 'Actions', 'Task Virtual Resources', 'Mutually Exclusive Tasks', 'Triggers', 'Notes', and 'Versions'. The 'Email Task' tab is active, showing a title bar with a green icon and the text 'Email Task' followed by a red warning icon and '= Required field'. To the right of the title bar are buttons for 'Update', 'Launch Task', 'View Instances', and 'Delete', along with a search icon. A context menu is open over the title bar, listing actions such as 'Launch with Variables', 'View Instances', 'Reset Statistics', 'Copy Task', 'Add To Bundle', 'View Bundles', 'Promote', 'Save', 'Insert', 'Insert and Stay', 'Assign Label ->', 'Copy URL to Clipboard', and 'Generate PDF ->'. Below the title bar, the configuration form includes fields for 'Task Name' (Opwise - Send), 'Version' (1), 'Task Description' (Send File Create), 'Member of Business Services' (locked), 'Email Connection' (Opwise - Gmail), 'Reply-To', 'To' (Opwise.Test@gmail.com), 'Cc', 'Bcc', 'Subject' (File was created and Processed.), and 'Body' (The File Trigger: \${ops_trigger_name} has fired because file \${ops_trigger_file_name} was created. A Workflow has been launched to process the file. Time: \${_date()} A Manual Task is now waiting to be performed! Sent from \${_hostname()} (\${_ipaddress()})). At the bottom, there are checkboxes for 'Late Start', 'Late Finish', and 'Early Finish', a 'Virtual Resource Priority' dropdown (set to 10), and a 'Hold Resources on Failure' checkbox. A row of buttons at the very bottom includes 'Update', 'Launch Task', 'View Instances', and 'Delete'.

Step 3 Select **Launch with Variables**. The Task Variables pop-up dialog displays. Any variables attached to this task automatically are displayed in alphabetic order (a-z).

Step 4 As needed, set the variable values or add new variables. The window allows you to specify up to six variable and value pairs.

Step 5 When you are finished, click **Submit**. Opwise populates the variables with the values you supplied and launches the task.

Changing the Priority of a Task Instance

For Windows, Linux/Unix, or z/OS tasks in a status of Started, you can change the priority so that they will run sooner or later, as described below. The priority specified here is meaningful only in relation to the priority setting of other tasks sent to that agent from the same Opwise instance.

Set Priority on a Task Instance from the Activity or Task Instances Screen

Step 1 Select the task instance(s) for which you want to set priority. All task instances must be in Started status.

Step 2 Select the priority level you want this task to have.

Today's Task Instances by Created Time						New Report	Edit Report
Instance Name	Type	Status	Start Time	End Time	Dur		
Opwise - Linux Business Day	Linux/Unix	Running	2009-09-14 11:19:49 -0700				
Opwise - Linux Broadcast	Linux/Unix	Started	2009-09-14 11:19:49 -0700				
Opwise - Linux Broadcast	Linux/Unix	Started	2009-09-14 11:19:49 -0700				
Opwise - Linux Broadcast	Linux/Unix	Started	2009-09-14 11:19:49 -0700				
Opwise - Linux Broadcast	Linux/Unix	Started	2009-09-14 11:19:49 -0700				
Opwise - Linux Broadcast	Linux/Unix	Success	2009-09-14 11:19:49 -0700	2009-09-14 11:19:50 -0700	1 Se		
Opwise - Linux Broadcast	Linux/Unix	Running	2009-09-14 11:19:49 -0700				
Opwise - Linux Broadcast	Linux/Unix	Finished	2009-09-14 11:18:53 -0700	2009-09-14 11:18:53 -0700	0 Se		
Opwise - Linux Random Failure	Linux/Unix	Success	2009-09-14 11:18:15 -0700	2009-09-14 11:18:15 -0700	0 Se		
Opwise - Linux Random Success	Linux/Unix	Finished	2009-09-14 11:18:15 -0700	2009-09-14 11:18:15 -0700	0 Se		

Step 3 Once the set priority command has been executed, Opwise displays the following message at the bottom of the screen.

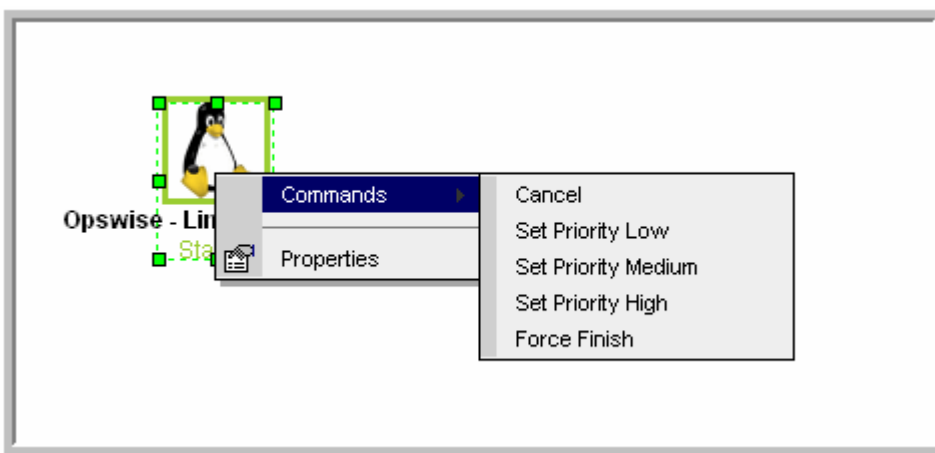
Command Set Priority High executed successfully against task instance Opwise - Linux Broadcast.

Set Priority on a Task Instance from the Workflow Monitor

Step 1 View the workflow that contains the task instance for which you want to set priority.

Step 2 Select the task instance(s) for which you want to set priority. The task(s) must be in Started status.

Step 3 Select **Commands**.



Step 4 Select the priority you want to give it.

Re-running a Task Instance

You can re-run a task instance while it is in any of the following statuses: Success, Start Failure, Failed, Cancelled, Finished. If a task instance is part of a workflow, you can only re-run it as long as the workflow has not completed. If the task instance is not part of a workflow, you can re-run it as long as it has not been manually purged from the Activity screen.

Note
You cannot re-run a workflow task instance.

When you re-run a task instance, Opwise uses the same task instance. That is, the new task instance has the same sys_id. However, you can view the two task instances distinctly on the Activity History screen (one for each time it ran).

You can re-run a task instance from the Activity or Task Instances screen. If the task instance is running as part of a workflow, you also can re-run it from the Workflow Monitor.

Re-run a Task Instance from the Activity or Task Instances Screen

Step 1	Select the task instance(s) you want to re-run.
Step 2	Click Re-run . The task status changes to the next appropriate status as though it had just been launched.

Re-run a Task Instance from the Task Instance Screen

Step 1	From the Activity screen, display the task instance you want to re-run.
Step 2	Click the Re-run button. The task status changes to the next appropriate status as though it had just been launched.

Re-run a Task Instance from the Workflow Monitor

Step 1	View the workflow that contains the task instance you want to re-run.
Step 2	Select the task instance(s) you want to re-run.
Step 3	Select Commands .
Step 4	Select Re-run . The task status changes to the next appropriate status as though it had just been launched, and the Workflow Console opens to display information about the re-run.

Cancelling a Task Instance

The Cancel command cancels a running task instance. For tasks that run on agents, including Windows, Linux, Unix, z/OS, FTP, File Monitor, and Indesca tasks, the Cancel command is sent to the agent.

- If the task instance has not yet been launched, it does not launch.
- If the task instance already has been launched, the agent cancels it, if possible.
- If the task instance is a workflow, any of its task instances in Running status go to Cancelled status; the workflow itself goes to Running/Problems status.
- If the task instance is in a workflow, the workflow goes to Running/Problems status. If the task is re-run, the workflow returns to Running status.

You can cancel a task instance while it is in any of the following statuses: Queued, Action Required, Started, Running.

You can cancel a task instance from the Activity or Task Instances screen. If the task instance is running as part of a workflow, you can also cancel it from the Workflow Monitor.

Cancel a Task Instance from the Activity or Task Instances Screen

Step 1	Select the task instance(s) you want to cancel.
Step 2	Click Cancel . The task status changes to Cancelled.

Cancel a Task Instance from the Workflow Monitor

Step 1	Open the Workflow Monitor for the workflow that contains the task instance you want to cancel.
Step 2	Select the task instance(s).
Step 3	Select Commands .
Step 4	Select Cancel . The task status changes to Cancelled and the Workflow Console opens to display information about the cancellation.

Force Finishing a Task Instance

The Force Finish command puts a task instance into the Finished status, regardless of what the task instance is doing.

One purpose of Force Finish is to allow downstream task instances in a workflow to launch without waiting for the current task instance to complete. You also may want to force finish a stand-alone task instance; for example, you may want to mark a failed job as Finished, rather than

rerunning the job.

If a task instance is running when the user issues a Force Finish, Opwise marks the task instance as Finished even though the actual process continues running. Two exceptions are the File Monitor and FTP File Monitor; for these task types, the monitoring processes are aborted by a Force Finish command. Assuming they have no other dependencies, all downstream task instances waiting for successful completion of this task instance will start.

When you issue a Force Finish against a workflow, the workflow and any of its tasks that are not already in Success, Finished, or Skipped status will go to Finished status.

You can force finish a task instance while it is in any of the following statuses: Defined, Waiting, Held, Resource Wait, Queued, Action Required, Started, Running, Cancel Pending, In Doubt, Failure to Start, Cancelled, Failed.

You can force finish a task instance from the Activity or Task Instances screens. If the task instance is running as part of a workflow, you can also force finish it from the Workflow Monitor.

Force Finish a Task Instance from the Activity or Task Instances Screen

Step 1	Select the task instance(s) you want to force finish.
Step 2	Click Force Finish . The task status changes to Finished.

Force Finish a Task Instance from the Workflow Monitor

Step 1	View the workflow that contains the task instance you want to force finish.
Step 2	Select the task instance(s).
Step 3	Select Commands .
Step 4	Select Force Finish . The task status changes to Finished and the Workflow Console opens to display information about the force finish.

Force Finish/Canceling a Task Instance

The Force Finish/Cancel command cancels a task instance and puts it into Finished status, regardless of what the task instance is doing.

One purpose of Force Finish is to cancel a task instance and allow downstream task instances in a workflow to launch without waiting for that task instance to complete. You also may want to force finish/cancel a stand-alone task instance; for example, you may want to mark a failed job as Finished, rather than rerunning the job.



Note

The Force Finish/Cancel command is not implemented for Sleep tasks, since for this type of task, the Cancel and Force Finish commands essentially perform the same function.

For tasks that run on agents, including Windows, Linux, Unix, z/OS, FTP, File Monitor, and Indesca tasks, the Force Finish/Cancel command is sent to the agent.

- If the task instance has not yet been launched, it does not launch.
- If a task instance is running when the user issues a Force Finish/Cancel command, the agent cancels the task instance, if possible, and then Opwise marks the task instance as Finished; processing does not continue. Assuming they have no other dependencies, all downstream task instances waiting for successful completion of this task instance will start.
- If the task instance is a workflow, any eligible task instances in the workflow are cancelled and set to the Finished status, and then the workflow itself is set to the Finished status.

You can force finish/cancel a task instance while it is in any of the following statuses: Queued, Action Required, Started, Running.

You can force finish/cancel a task instance from the Activity or Task Instances screen. If the task instance is running as part of a workflow, you can also force finish/cancel it from the Workflow Monitor.

Force Finish/Cancel a Task Instance from the Activity or Task Instances Screen

Step 1	Select the task instance(s) you want to force finish/cancel.
Step 2	Click Force Finish/Cancel . The task status changes to Finished.

Force Finish/Cancel a Task Instance from the Workflow Monitor

Step 1	View the workflow that contains the task instance you want to force finish/cancel.
Step 2	Select the task instance(s).
Step 3	Select Commands .
Step 4	Select Force Finish/Cancel . The task status changes to Finished and the Console opens to display information about the force finish/cancel.

Putting a Task Instance on Hold

You can put a task instance on hold while it is in any of the following statuses: Defined, Waiting, Resource Wait, Queued.

If you put a workflow on hold that has not yet started, the workflow and all the task instances in it are put on hold. If you put a workflow on hold when it is in running status, all the task instances within the workflow that have not yet started are put on hold; however, the workflow itself does not go to Hold status because it already has started.

To release the workflow and all of its task instances that are on hold, issue the Release Recursive command against the workflow. To release the workflow but keep the task instances on hold until you release them one by one, use Release on the workflow first, then use Release on each task instance.

Hold a Task Instance from the Activity or Task Instances Screen

Step 1	Select the task instance(s) you want to put on hold.
Step 2	Click Hold . The task status changes to Held.

Hold a Task Instance from the Workflow Monitor

Step 1	View the workflow that contains the task instance you want to put on hold.
Step 2	Select the task instance(s).
Step 3	Select Commands .
Step 4	Select Hold . The task status changes to Held and the Workflow Console opens to display information about the hold.

Releasing a Task Instance from Hold

You can release a non-workflow task instance from hold from the Activity or Task Instances screen while it is in Held status.

For workflows, if the user held a workflow that already was running, only the task instances within the workflow that had not started yet are put into Held status. In this case, the workflow itself does not go to Held status.

To release the workflow, use one of the following methods:

- To release the entire Held workflow and its task instances, use **Release Recursive**.
- To release task instances that are in Held status in a Workflow that itself is not in Held status, use **Release Recursive**. In this case, you can issue a Release Recursive on a workflow in any of the following statuses: Defined, Waiting, Held, Resource Wait, Running.
- To release the workflow but keep the task instances inside on hold so that you can release them one by one, use **Release**. In this case, release the workflow first, then release each task instance manually.

Release a Held Task Instance from the Activity or Task Instances Screen

Step 1	Select the task instance(s) you want to release from hold.
Step 2	Click Release or Release Recursive . The task status changes to the next appropriate status according to where it was in processing at the time it was put on hold.

Release a Held Task Instance from the Workflow Monitor

Step 1	View the workflow that contains the task instance you want to release.
Step 2	Select the task instance(s).
Step 3	Select Commands .
Step 4	Select Release . The task status changes to the next appropriate status according to where it was in processing, and the Workflow Console opens to display information about the release.

Skipping a Task Instance

You can skip any task instance as long as it has not yet started running. This includes task instances in the following statuses: Defined, Waiting, Held, Resource Requested, Resource Wait.

You also can skip a task instance path so that a task instance and all of its dependent task instances automatically are skipped as well.

Two methods are available for specifying that you want to skip a task instance:

1. Instruct Opwise to skip a task instance from the [Activity screen](#) or the [Task Instances screen](#).
2. If a task instance is running as part of a workflow, you also can instruct Opwise to skip the task instance from the Workflow Monitor.



Note

You also can specify that a task instance will be skipped (before the task or its workflow is launched) by:

1. Modifying a trigger definition (using the trigger's Skip Count field) so that Opwise skips the next N number of trigger occurrences for launching the task.
2. Modifying a workflow definition by specifying [conditional paths](#) that may place one or more task instances in the Skipped status when the workflow is run.
3. Modifying a workflow definition by specifying that one or more task instances should be skipped (or run) at specific times (see [Adding Skip/Run Criteria for Specific Tasks](#)).

If you skip a workflow task instance, all the task instances within the workflow also are skipped, along with any nested workflows.

Once a task instance has been skipped, the only command you can run against it is [Unskip](#).

Skip a Task Instance from the Activity Screen or Task Instances Screen

Step 1	Select the task instance(s) you want to skip.
Step 2	Click Skip . The task status changes to Skipped.
Step 3	To skip the task instance and all of its dependent task instances, click Skip Path . The task status of the task instance and all of its dependent task instances changes to Skipped.

Skip a Task Instance from the Workflow Monitor

Step 1	View the workflow that contains the task instance you want to skip.
Step 2	Select the task instance(s).
Step 3	Select Commands .
Step 4	Select Skip . The task status changes to Skipped, and the Console opens to display information about the skip.
Step 5	To skip the task instance and all of its dependent task instances, click Skip Path . The task status of the task instance and all of its dependent task instances changes to Skipped, and the Console opens to display information about the skip.

Showing or Hiding Skipped Task Instances

You can select whether to show or hide skipped task instances from the Workflow Monitor either before the workflow is running or while the workflow is running.

Three methods are available for selecting whether or not to show or hide skipped task instances:

1. From the Workflow Task Definition screen.
2. From the Workflow Task Instance screen.
3. From the Workflow Monitor.

Setting the Show / Hide Skipped Tasks Workflow Definition Option

Step 1	Display the Workflow Task Definition screen for the workflow that you want to configure.
Step 2	Use the Show / Hide Skipped Tasks field to select whether you want to show or hide skipped task instances (default is Show Skipped). When viewing a running workflow in the Workflow Monitor, the skipped task instances will be shown or hidden based on your selection.

Setting the Show / Hide Skipped Tasks Workflow Instance Option

Step 1	Display the Workflow Task Instance screen for the workflow instance that you want to configure.
Step 2	Use the Show / Hide Skipped Tasks field to select whether you want to show or hide skipped task instances (default is Show Skipped). When viewing the workflow instance in the Workflow Monitor, the skipped task instances will be shown or hidden based on your selection.

Show / Hide Skipped Tasks from the Workflow Monitor

Open the workflow instance in the Workflow Monitor. By default, the Workflow Monitor will show or hide skipped task instances based on the workflow instance's Show / Hide Skipped Tasks option.

To temporarily change the behavior, right-click in the Workflow Monitor canvas and select either of the following entries from the pop-up menu:

- Show Skipped / Restore
- Hide Skipped

Unskipping a Task Instance

If a task instance in a workflow has been skipped (perhaps at trigger time due to run criteria or manually by running the skip command), you can unskip that task instance while the workflow is running.



Note

If you unskip a task instance that was skipped by issuing a Skip Path command against it, which automatically skip all of its dependent tasks, those dependent tasks stay in Skipped status. You must manually unskip each task to remove them from Skipped status.

Two methods are available for unskipping a task instance:

1. From the Activity or Task Instances Screen.
2. From the Workflow Monitor.

Unskip a Task Instance from the Activity or Task Instances Screen

Step 1	Select the task instance(s) you want to unskip.
Step 2	Click Unskip . The task instance will run when all of its dependencies have been satisfied.

Unskip a Task Instance from the Workflow Monitor

Step 1	View the workflow that contains the task instance(s) you want to unskip.
Step 2	Select a task instance(s). (You can issue commands only against one task at a time within the Workflow Monitor.)
Step 3	Select Commands .
Step 4	Select Unskip . A confirmation message will appear in the Console, and the task instance will run when all of its dependencies have been satisfied.

Clearing All Predecessor Dependencies for a Task Instance

For task instances running inside of a workflow, you can clear all predecessor (upstream) dependencies to allow the task instance to run. Clearing a dependency has the same result as satisfying a dependency. You can clear dependencies on task instances in the following status: Defined, Waiting, Held.



Note

Clearing predecessor dependencies does not include the clearing of resource and mutually exclusive dependencies. To clear these dependencies, see [Clearing Resource Dependencies from Tasks](#) and [Clearing Mutually Exclusive Dependencies from Tasks](#), below.

Clear Dependencies on a Task Instance from the Activity or Task Instances Screen

Step 1	Select the task instance(s) whose dependencies you want to satisfy.
Step 2	Click Clear Dependencies . The task instance is launched normally.

Clear Dependencies on a Task Instance from the Workflow Monitor

Step 1	View the workflow that contains the task instance whose dependencies you want to satisfy.
Step 2	Select the task instance(s) for which you want to clear upstream dependencies.
Step 3	Select Commands .
Step 4	Select Clear Dependencies . The task instance is launched normally.

Marking a Dependency as Satisfied

For task instances running inside of a workflow, you can clear a single upstream dependency to allow the task instance to run. Clearing a dependency has the same result as satisfying a dependency. You can clear a dependency on task instances in the following status: Defined, Waiting, Held.

Step 1	View the workflow that contains the task instance whose dependencies you want to satisfy.
Step 2	Locate and right-click on the task dependency (the connector line between two tasks).
Step 3	Select Commands .
Step 4	Select Mark as Satisfied . If all other dependencies are satisfied, the task instance is launched normally.

Clearing Resource Dependencies from Tasks

For task instances for which resources have been defined, you can clear those resource dependencies.

You can clear resource dependencies from task instances in the following status: Resource Wait.

Three methods are available for clearing resource dependencies from task instances:

1. From the Activity screen.
2. From the Task Instances screen.
3. From the Workflow Monitor.

Clear Resource Dependencies of a Task Instance from the Activity or Task Instances Screen

Step 1	Select the task instance(s) whose resources you want to clear.
Step 2	Click Clear Resources . Resource dependencies are cleared from the task instance.

Clear Resource Dependencies of a Task Instance from the Workflow Monitor

Step 1	View the workflow that contains the task instance(s) you want to clear of resource dependencies.
Step 2	Select a task instance(s). (You can issue commands only against one task at a time within the Workflow Monitor.)
Step 3	Select Commands .
Step 4	Select Clear Resources . A confirmation message will appear in the Console, and the task instance will run without resources.

Clearing Mutually Exclusive Dependencies from Tasks

For task instances that are mutually exclusive with other task instances, you can clear those mutually exclusive dependencies.

You can clear mutually exclusive dependencies on task instances in the following status: Exclusive Wait.

Three methods are available for clearing mutually exclusive dependencies from task instances:

1. From the Activity screen.
2. From the Task Instances screen.
3. From the Workflow Monitor.

Any task instances that were mutually exclusive with this task instance will no longer be mutually exclusive.

Clear Mutually Exclusive Dependencies from a Task Instance from the Activity or Task Instances Screen

Step 1	Select the task instance(s) whose mutually exclusive dependencies you want to clear.
Step 2	Click Clear Exclusive . Mutually exclusive dependencies are cleared from the task instance and it is launched normally.

Clear Mutually Exclusive Dependencies from a Task Instance from the Workflow Monitor

Step 1	View the workflow that contains the task instance(s) you want to clear of mutually exclusive dependencies.
Step 2	Select a task instance(s). (You can issue commands only against one task at a time within the Workflow Monitor.)
Step 3	Select Commands .
Step 4	Select Clear Exclusive . A confirmation message will appear in the Console, and the task instance will run normally.

Creating and Maintaining Workflows

- Overview
 - Rules for Creating Workflows
 - Workflow Modes
 - Icon Reference
- Defining a Workflow
 - Creating a New Workflow
 - Field Descriptions on Workflow Task Screen
 - Searching For and Adding Tasks
 - Specifying Connections
 - Moving Workflow Elements
 - Deleting Workflow Elements
 - Copying Workflow Elements
 - Undoing and Redoing Workflow Changes
 - Zooming In and Out
 - Panning Around in Large Workflows
 - Automatically Formatting a Workflow
 - Displaying Workflow Documentation
 - Displaying Processing Messages
 - Saving the Workflow
 - Switching Between Workflows
- Adding Skip/Run Criteria for Specific Tasks
 - Creating New Run Criteria
 - Task Run Criteria Field Descriptions
- Specifying When a Workflow Runs
- Monitoring Workflow Execution
- Modifying an Existing Workflow
- Deleting a Workflow
- Finding a Task in a Workflow
- Inserting a Task in a Workflow
 - Insert Task as Predecessor
 - Insert Task as Successor
 - Insert Task with Multiple Predecessors/Successors
- Modifying Tasks in a Workflow

Overview

The Workflow Definition tool is a graphical tool that allows you to select tasks, position them within a workflow, and specify the dependency relationships between them.

The process of creating a workflow involves:





















Step 1	Creating a new workflow task.
Step 2	Accessing the Workflow Editor and: <ul style="list-style-type: none"> • Opening the Add Task window. • Specifying parameters for fetching tasks. • Dragging tasks onto the canvas. • Adding connections and dependencies between the tasks. • Defining the layout of the workflow.
Step 3	Saving the new workflow.










Workflows can be as simple or elaborate as necessary. Zooming and scrolling features are provided that allow you to work on small areas of a very large workflow, or to create simple ad hoc workflows.

The following sample workflow consists of a variety of task types. There are no restrictions on the types of tasks that can be included as part of a single workflow and you can also put workflows within other workflows.

Icon Reference

The following table describes the icons used to define workflows.

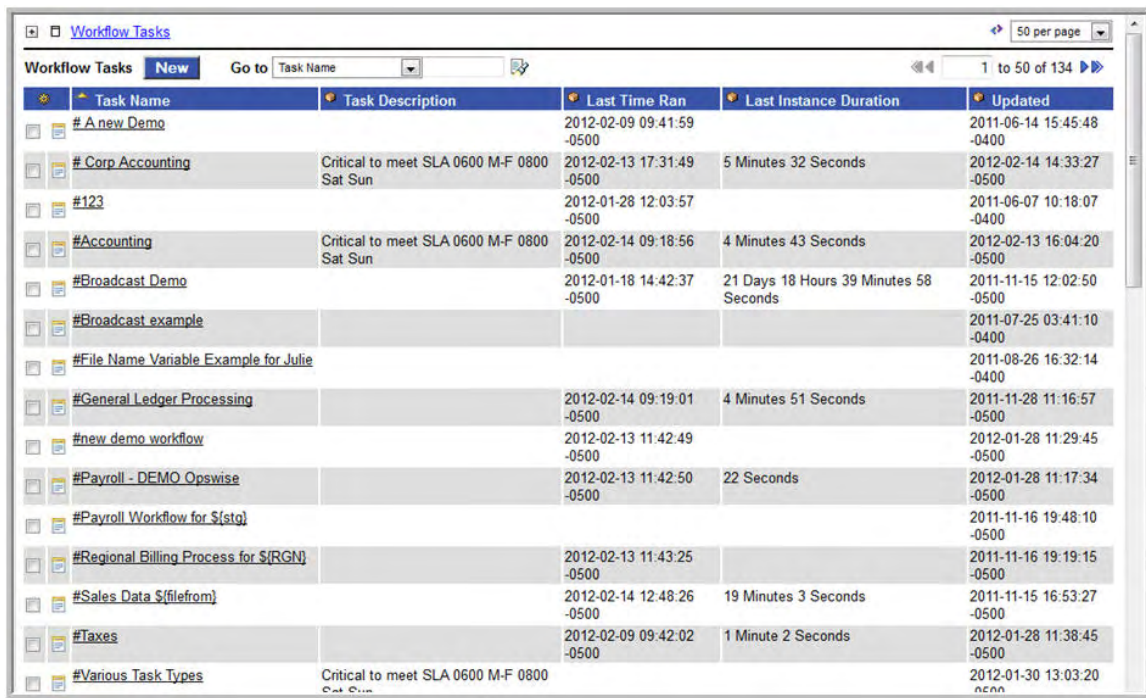
Icon	Description
	Back --- Goes back to the previous screen.
	Go to Parent --- If this workflow is embedded in another workflow, navigates to the parent. If this workflow has more than one parent, clicking this button displays a list of all parent workflows; in this case, to navigate to a parent workflow, double click the workflow's icon. To return to your original workflow, click its name. Note that the current workflow name displays in the lower right corner of the editor.
	Open Workflow --- Allows you to drag another workflow into the current workflow or temporarily switch to another workflow without closing the current workflow. It opens a window and displays a searchable list of defined workflows. (The Workflow Search Result Limit Opwise system property defines how many workflows display on the list.) You can either double-click a workflow from the list to open it and edit it, or you can click and drag a workflow into the current workflow. If you open another workflow, use Open Workflow to return to the previous workflow. Hint: As a best practice, save your workflow before opening another one.
	Save --- Saves the current workflow.
	<p>Add Task --- Displays the Task Find window, which allows you to search for and select tasks for the workflow from a searchable list of defined tasks. (The Workflow Search Result Limit Opwise system property defines how many tasks display on the list.) To search for and select tasks:</p> <ol style="list-style-type: none"> 1. Click the Add Task icon. 2. Optionally, enter a task name, a partial string, optionally using or ? as wildcards , or select the type of task you are searching for. 3. Click Search. Opwise displays the tasks that meet your selection criteria. 4. To bring a task onto the workflow canvas, click and drag the task's icon out of the window and onto the canvas. 5. To close the Task Find window, click the X in the upper right corner.
	<p>Select --- Enters Select mode, which allows you to click tasks or links in order to move or delete them. Use this to perform any of the following:</p> <ul style="list-style-type: none"> • Switch out of Connect mode • Select and open tasks • Select and delete tasks and links • Select and move tasks and links
	Pan --- Enters Pan mode, which allows you to scroll to different areas of the workflow.
	<p>Connect (bent) and Connect (straight) --- Enters Connect mode, which allows you to create links between tasks. To create a link:</p> <ol style="list-style-type: none"> 1. Click either icon to enter Connect mode. 2. Click the "upstream" task. If you are using Firefox, click the center of the task icon. In Internet Explorer, click the task name. 3. Drag the cursor to the "downstream" task. You will see a colored line as you drag. 4. When you reach the "downstream" task, release the cursor. The link appears as a straight or bent line, depending on which icon you selected.
	Undo --- Click to undo the most recent change.
	Redo --- Click to redo the most recent change that you undid by clicking Undo .
	Cut --- Deletes the selected object or objects (tasks and links or both) and keeps a copy in memory. Use CTRL-Click to select and cut multiple objects. Hint: Do not use cut and paste to move workflow elements; use select and drag .
	Copy --- Copies the selected object or objects (tasks, links or both). Use CTRL-Click to select and copy multiple objects.
	Paste --- Pastes the copied or cut object or objects to the currently open workflow.
	Delete --- Permanently deletes the selected object or objects. Delete does not keep a copy of the deleted objects in memory.
	Fit --- Fits the workflow into the display. If necessary, this shrinks the icons and size of the workflow in order to make it fit. You can undo a Fit by selecting Actual Size  .
	Zoom In --- Zooms in (enlarges) the workflow. To return the workflow to its default size, select Actual Size  .
	Zoom Out --- Zooms out (diminishes) the workflow. To return the workflow to its default size, select Actual Size  .

	Actual Size --- Returns the workflow to its default size after a Fit or Zoom .
	Zoom --- Opens a window that allows you to specify a zoom ratio. For example, to double the size of the workflow, enter 200 and click OK . To return the workflow to its default size, select Actual Size  .
	Horizontal Layout --- Reformats the workflow into a horizontal layout.
	Vertical Layout --- Reformats the workflow into a vertical layout.
	Toggle Vertex Style --- For running workflows, switches the icon display between status-related icons and task-related icons. This icon only appears on the icon bar when you are monitoring a running workflow.
	<p>Outline --- For large workflows, the outline provides a way of positioning a specific area of the workflow in the display, without using the Pan icon.</p> <ol style="list-style-type: none"> 1. Click the Outline icon. The Outline window opens. 2. In the Outline window, move and/or resize the blue box to identify the area of the workflow you want to work on. The display repositions to show only the area within the blue box.
	Help --- Displays help documentation for workflows.
	Console --- While a workflow is running, you can click the Console icon to display processing messages. For more information on the Console, see Monitoring Workflows .

Defining a Workflow

Creating a New Workflow

Step 1 From the navigation pane, select **Automation Center > Tasks > Workflow Tasks**. The Workflow Tasks List screen displays.



Task Name	Task Description	Last Time Ran	Last Instance Duration	Updated
# A new Demo		2012-02-09 09:41:59 -0500		2011-06-14 15:45:48 -0400
# Corp Accounting	Critical to meet SLA 0600 M-F 0800 Sat Sun	2012-02-13 17:31:49 -0500	5 Minutes 32 Seconds	2012-02-14 14:33:27 -0500
#123		2012-01-28 12:03:57 -0500		2011-06-07 10:18:07 -0400
#Accounting	Critical to meet SLA 0600 M-F 0800 Sat Sun	2012-02-14 09:18:56 -0500	4 Minutes 43 Seconds	2012-02-13 16:04:20 -0500
#Broadcast Demo		2012-01-18 14:42:37 -0500	21 Days 18 Hours 39 Minutes 58 Seconds	2011-11-15 12:02:50 -0500
#Broadcast example				2011-07-25 03:41:10 -0400
#File Name Variable Example for Julie				2011-08-26 16:32:14 -0400
#General Ledger Processing		2012-02-14 09:19:01 -0500	4 Minutes 51 Seconds	2011-11-28 11:16:57 -0500
#new demo workflow		2012-02-13 11:42:49 -0500		2012-01-28 11:29:45 -0500
#Payroll - DEMO Opwise		2012-02-13 11:42:50 -0500	22 Seconds	2012-01-28 11:17:34 -0500
#Payroll Workflow for \$(stg)				2011-11-16 19:48:10 -0500
#Regional Billing Process for \$(RGN)		2012-02-13 11:43:25 -0500		2011-11-16 19:19:15 -0500
#Sales Data \$(filefrom)		2012-02-14 12:48:26 -0500	19 Minutes 3 Seconds	2011-11-15 16:53:27 -0500
#Taxes		2012-02-09 09:42:02 -0500	1 Minute 2 Seconds	2012-01-28 11:38:45 -0500
#Various Task Types	Critical to meet SLA 0600 M-F 0800 Sat Sun			2012-01-30 13:03:20 -0500

Step 2 Click **New**. The Workflow Task Definition screen displays.

Step 3 Using the field descriptions provided below as a guide, complete the fields as needed.

Step 4 Right-click the title bar and select **Save**. The workflow is added to the database and additional buttons appear at the bottom.

Step 5 Click **Edit Workflow** to proceed into the Workflow Editor. Opwise displays a blank Workflow Editor screen.

Step 6 Follow the instructions provided below for [adding tasks](#), [creating connections](#), [specifying conditions on Connections](#), [organizing](#), and [saving your workflow](#); or see [Icon Reference](#) for a description of each tool icon.

Field Descriptions on Workflow Task Screen

The table below describes the fields, buttons, and tabs on the task definition and task instance screens. Color coding is provided that differentiates the following three types of fields:

- Fields that display on the task definition and task instance screens are shown in black.
- Fields that display only on the task definition screen are shown in green.
- Fields that display only on the task instance screen are shown in maroon.

Field Name	Description
Task/Instance Name	Required. Name used within Opwise to identify this task. Up to 40 alphanumeric; variables supported. It is the user's responsibility to develop a workable naming scheme for tasks.
Invoked by	Task instance only; system-supplied. How the task instance was launched. One of the following: <ul style="list-style-type: none"> • Trigger: (Trigger Name) - The instance was launched by the named trigger. • Workflow: (Workflow Name) - The instance was launched by the named workflow. • Manually Launched - The instance was launched by a user. To determine the name of the user: <ol style="list-style-type: none"> 1. From the Activity or Task Instances screen, click the task instance name to open the record. 2. The Execution User field identifies the user who launched the task instance.
Task	Task instance only; system-supplied. Hover over the paper icon to display more information about the task instance. Click the paper icon to display the task definition record.
Instance Reference Id	Task instance only; system-supplied. Opwise increments this number each time the task is run.
Member of Business Services	User-defined. Allows you to select one or more Business Services that this record definition belongs to. Click the lock icon to unlock the field and select Business Services .

Version	Task definition only; system-supplied. The version number of the current record, which is incremented by Opwise every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Task Description	User-supplied description of this record.
Summary	User-supplied description of this record.
Status	Task instance only; system-supplied. See Task Instance Statuses .
Status Description	Task instance only; system-supplied. Provides additional information, if any, about the status of the task.
Start Time	Task instance only; system-supplied. The date and time the task started.
Duration	Task instance only; system-supplied. The amount of time the task took to run.
End Time	Task instance only; system-supplied. The date and time the task instance completed.
User Estimated End Time	Task instance only; system-supplied. If the user entered information into the User Estimated Duration field in the task definition, Opwise uses this information to calculate an end time for the task instance, based on the date/time the task instance started.
Shortest Estimated End Time	Task instance only; system-supplied.
Average Estimated End Time	Task instance only; system-supplied.
Longest Estimated End Time	Task instance only; system-supplied.
Hold on Start	If enabled, when the task is launched it appears in the Activity display with a status of Held. The task runs when the user releases it.
Hold Reason	Information about why the task will be put on hold when it starts.
User Estimated Duration	Task definition only; optional. The estimated amount of time it should normally take to run this task. Opwise uses this information to calculate the User Estimated End Time on a task instance record.
Late Start	If enabled, and if the task instance starts after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late start (see Late Start Type). To determine whether a task instance started late, open the task instance and locate the Started Late field; the field is checked if the instance started after the specified time. This field only appears on the task instance if the user specified a Late Start in the task definition.

Late Start Type	<p>Required if Late Start is enabled. Options are:</p> <ul style="list-style-type: none"> • Time - Flag the task if it starts after the specified time. • Duration - Flag the task if it starts a certain amount of time after the programmed start time. The task must have a specific start time.
Late Start Time	Time after which the task start time is considered late. Use hh:mm, 24-hour time
Late Start Duration	<p>Duration (amount of relative time) after which the task is considered to have started late. For a task within a workflow, the duration is the period between the time the workflow starts and the time the task itself starts. For example, a task might have a Late Start Duration of 60 minutes. If the workflow starts at 9:00 a.m. but the task itself does not start until 10:30, the task has started late.</p> <p>For a task that is not within a workflow, Late Start Duration has meaning only if the task has been held upon starting. For example, if a task has a Late Start Duration of 60 minutes and the Hold on Start field is enabled, if the task is not released from hold within the amount of time specified in the Late Start Duration field, the task has started late.</p>
Started Late	Task instance only; system-supplied. This field is flagged if the task started later than the time specified in the Late Start fields.
Late Finish	If enabled, and if the task instance finishes after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late finish (see Late Finish Type). To determine whether a task instance finished late, open the task instance and locate the Finished Late field; the field is checked if the instance finished after the specified time or lasted longer than expected. This field only appears on the task instance if the user specified a Late Finish in the task definition.
Late Finish Type	<p>Required if Late Finish is enabled. Options are:</p> <ul style="list-style-type: none"> • Time - Flag the task if it finishes after the specified time (see Late Finish Time). • Duration - Flag the task if it finishes a certain amount of time after the programmed finish time (see Late Finish Duration). The task must have a specific finish time.
Late Finish Time	If Late Finish Type is Time, use this to specify the time after which the task finish time is considered late. Use hh:mm, 24-hour time.
Late Finish Duration	If Late Finish Type is Duration, use this to specify the longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Finished Late	Task instance only; system-supplied. This field is flagged if the task finished later than the time or duration specified in the Late Finish fields.
Early Finish	If enabled, and if the task instance finishes before the time or period specified, the task instance is flagged as early. You can specify a time or duration to determine an early finish (see Early Finish Type). To determine whether a task instance finished early, open the task instance and locate the Finished Early field; the field is checked if the instance finished before the specified time or did not last as long as expected. This field only appears on the task instance if the user added Early Finish specifications to the task definition.
Early Finish Type	<p>Required if Early Finish is enabled. Options are:</p> <ul style="list-style-type: none"> • Time - Flag the task if it finishes before the specified time (see Early Finish Time). • Duration - Flag the task if it finishes a certain amount of time before the programmed finish time (see Early Finish Duration). The task must have a specific finish time.

Early Finish Time	If Early Finish Type is Time, use this to specify the time before which the task finish time is considered early. That is, enter a time at which the task should still be running. Use hh:mm, 24-hour time.
Early Finish Duration	If Early Finish Type is Duration, use this to specify the shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Finished Early	Task instance only; system-supplied. This field is flagged if the task finished earlier than the time specified in the Early Finish fields.
First Time Ran	Task definition only; system-supplied. Displays after the first time the task runs. The date and time this task first ran.
Lowest Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. The shortest amount of time this task has taken to run.
Last Time Ran	Task definition only; system-supplied. Displays after the first time the task runs. The most recent date and time the task ran.
Average Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. Shows the average amount of time this task takes to run.
Number of Instances	Task definition only; system-supplied. Displays after the first time the task runs. Shows the number of times this task has run.
Highest Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. The longest amount of time this task has taken to run.
Last Instance Duration	Task definition only; system-supplied. Displays after the first time the task runs. The amount of time the task took to run the last time it ran.
Show/Hide Skipped Tasks	Specification to either show or hide tasks that have been skipped in a Workflow (see Skipping a Task). Options: <ul style="list-style-type: none">• Show Skipped• Hide Skipped
Virtual Resource Priority	Priority for acquiring a resource when two or more tasks are waiting for the resource. This priority applies to all resources required by the task. Options: 1 (high) - 20 (low). Default is 10.
Hold Resources on Failure	If enabled, the task instance will continue to hold Renewable resources if the task instance fails. Renewable resources will be returned only if the task instance status is either Complete, Finished, or Skipped.
Default Calendar	Default calendar used by the workflow. If the workflow is launched by a trigger, the trigger calendar overrides this default calendar.

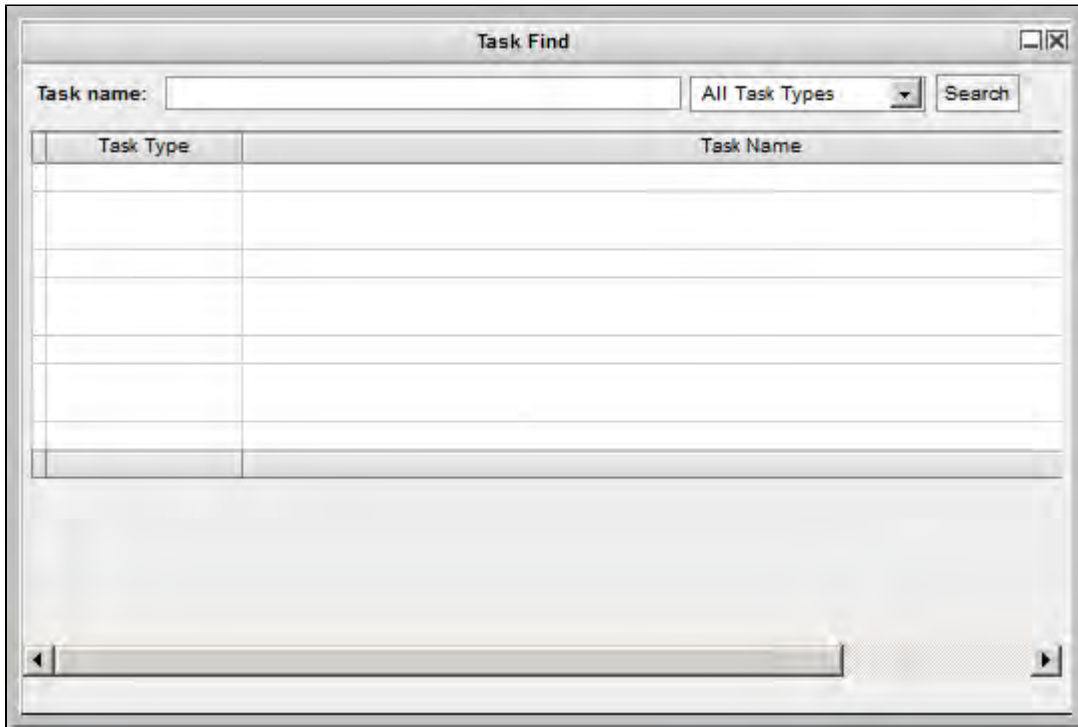
Auto Layout	<p>Task instance only; Specification for the layout of the Workflow view.</p> <p>Options:</p> <ul style="list-style-type: none"> * none * Horizontal Layout * Vertical Layout
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Launch Task button	Manually launches the task.
View Instances button	Displays a list of task instances for which there has been a status change or a modification to the task instance record within the last 30 days (an Updated on Last 30 Days filter has been pre-selected for the list).
Delete button	Deletes the current record.
View Parent button	Task instance only; Displays this task's parent task (workflow), if any.
Show Details button	Task instance only; displays detailed information about this task instance.
View Workflow button	Displays the graphical workflow.
Cancel button	Cancels a running task. See Cancelling a Task Run .
Release Recursive button	Task Instance only. Releases the entire held workflow and its tasks. See Releasing a Task From Hold .
Hold button	Task instance only; see Putting a Task on Hold .
Force Finish button	Task instance only; see Force Finishing a Task .
Output tab	Task instance only. Displays output generated from the process, if any, based on specifications provided by the user in the Automatic Output Retrieval fields in the task definition.
Task Run Criteria tab	Workflows only. Allows you to specify skip and run criteria for specific tasks in the workflow.
Variables tab	Displays all variables associated with this record.

Actions tab	<p>Allows you to specify actions that Opswise will take automatically based on events that occur during the execution of this task. Events are task instance status, exit codes, late start, late finish, and early finish. Actions are:</p> <ul style="list-style-type: none"> • Abort Action - Abort the task if certain events occur. For details, see Setting Abort Actions. • Email Notification - Send an email if certain events occur. For details, see Creating Email Notifications. • Set Variable - Used in workflows to set a variable based on the occurrence of certain events. For details, see Setting Variables within a Workflow. • SNMP Notification - Send an email if certain events occur. For details, see Creating SNMP Notifications. • System Operation - Run an Opswise system operation based on specified conditions. For details, see Setting System Operations.
Task Virtual Resources tab	Lists Virtual Resources to which this task is assigned.
Mutually Exclusive Tasks tab	Displays all tasks that have been set to be mutually exclusive of this task.
Step Conditions tab	Displays a list of all step conditions defined for this task.
Triggers tab	Displays a list of all triggers that have been defined to launch this task. Also allows you to add new triggers. If you add a new trigger from this location, Opswise automatically constructs a default trigger name as follows: <current task name>#TRIGGER#. You can change the default name if desired. For instructions on creating triggers, see Creating Triggers .
Notes tab	Displays all notes associated with this task.
Versions tab	Stores copies of all previous versions of the current record. See Record Versioning .

Searching For and Adding Tasks

Step 1

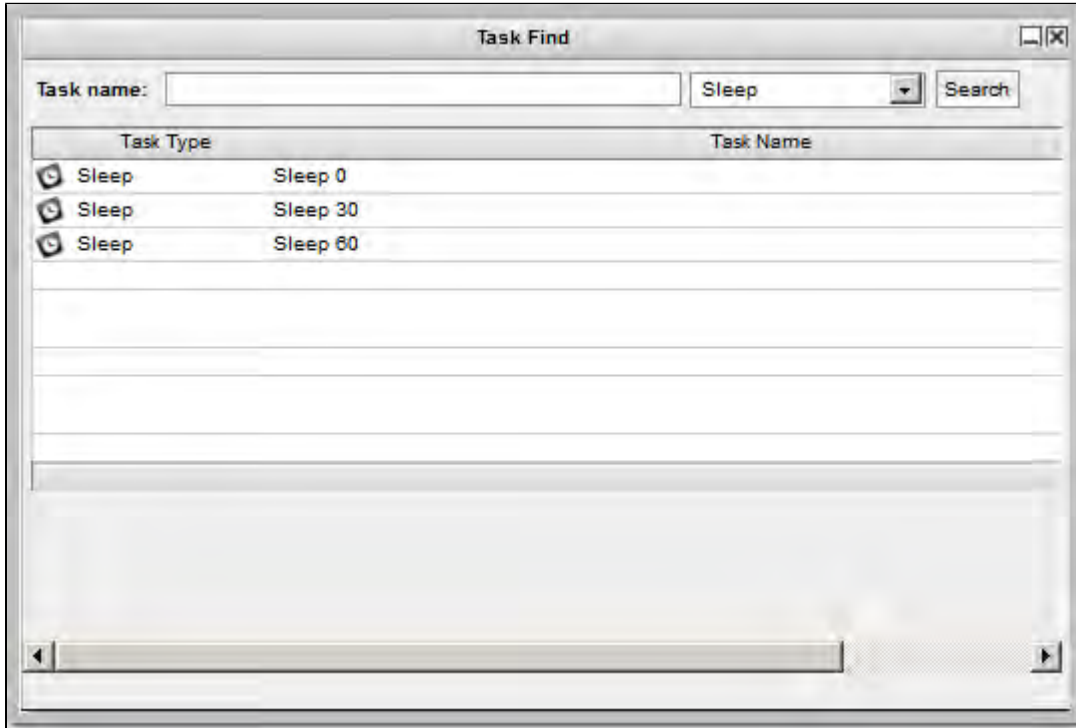
Click the **Add Task** icon . The **Task Find** dialog displays.



Step 2 Several methods are available for finding tasks:

- To find a specific task, type the name into the Task name field and click **Search**.
- To display a list of tasks whose names match a string, type the string into the Task name field and use one or more [wildcards](#). The wildcard(s) can appear anywhere in the string. For example, to find tasks whose name begins with "Fee", type **Fee** in the Task name field and click **Search**.
- To display a list of tasks of a specific type, such as Windows, select the task type from the drop down menu and click **Search**.
- To display a list of all tasks, select All Task Types (the default selection) from the drop down menu and click **Search**.

The Task Find dialog lists the task(s) that match your search criteria.





Step 3 To add a task to the workflow canvas, click the icon to the left of the task and drag it onto the canvas.

Step 4 Repeat these steps until you have added all the tasks you need.

Step 5 To close the **Task Find** window, click the **X** in the upper right corner. Or, to keep the window open but minimized, click the minimize icon.

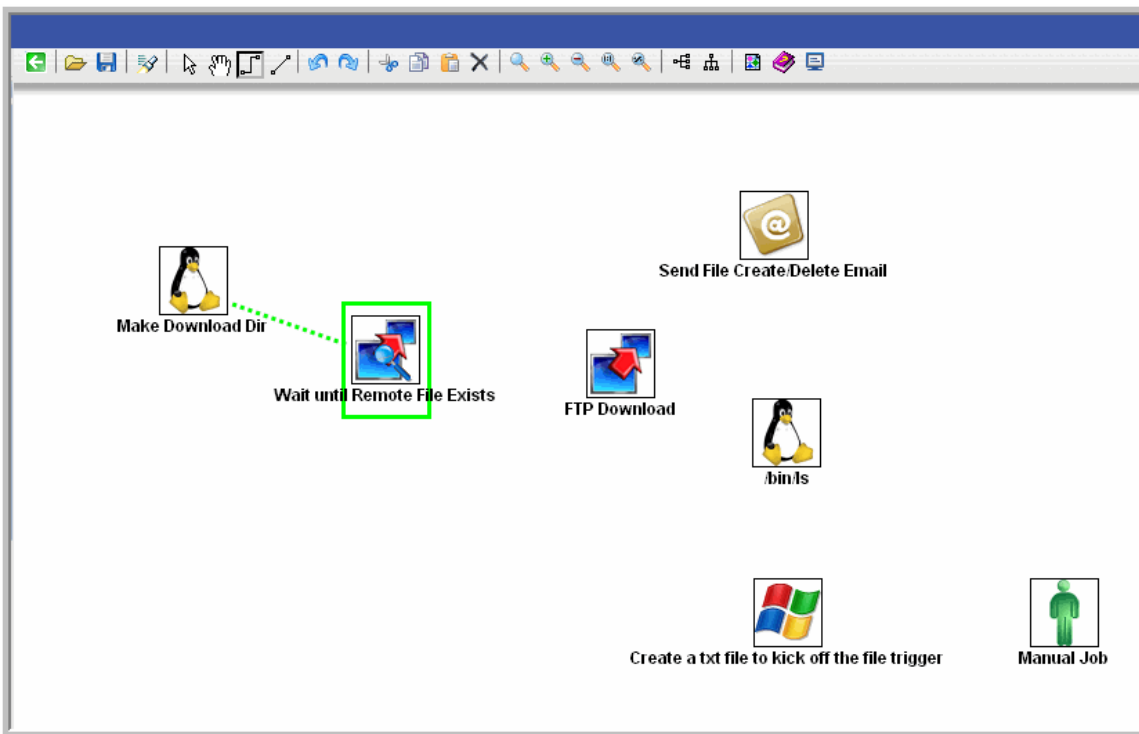
Specifying Connections

You must first drag tasks onto the canvas before specifying connections between them.

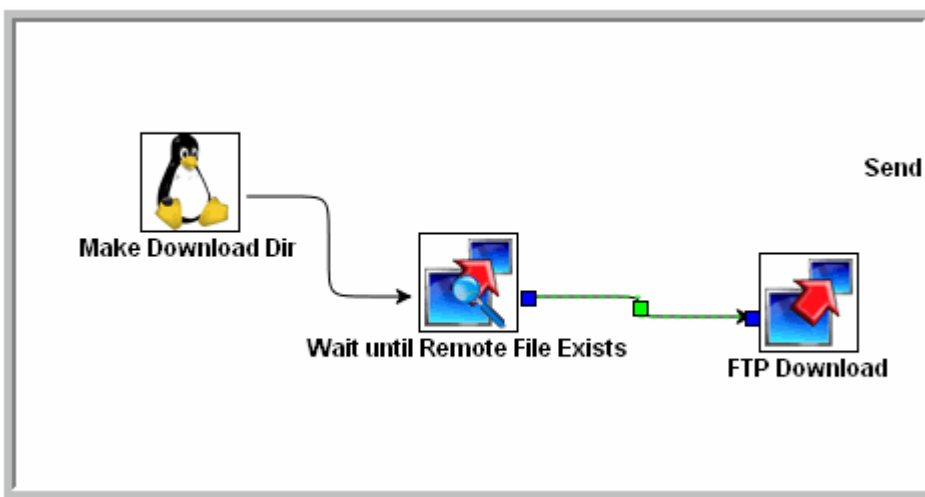
Step 1 Click one of the connectors, either  , which is either straight or diagonal, or  , which bends, if necessary. This enters **Connection** mode.

Step 2 Click the upstream task, that is, the task whose status will trigger the next task. For Firefox, make sure you click the center of the task icon; for Internet Explorer, click the task name, **hold down the cursor**, and drag the connector.

Step 3 Drag the connector to the downstream task, that is, the task that will be triggered. As you drag, a dotted line displays, indicating the connection. Drag the connector all the way to the center of the downstream, highlighted task.



Step 4 When the cursor is at the center of the downstream task icon, release the mouse button. The connector is in place and still highlighted. Optionally, you can reposition the connection by dragging it to a new location. Arrows on the connectors indicate the direction of the workflow.



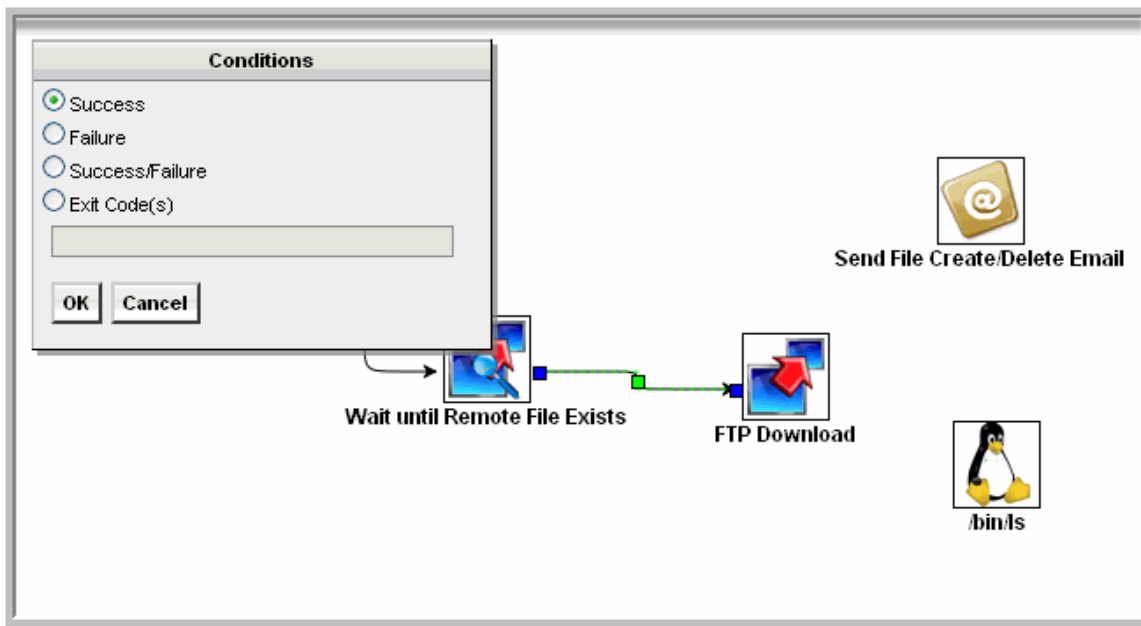
Step 5 Repeat this process for other connectors you need to add.

Specifying Conditions on Connections

Once you have created connections between the tasks on a workflow, you can now specify what kind of dependency condition exists between the upstream and downstream task. The default condition is Success. That is, the downstream task runs if the upstream task runs to Success.

Step 1 Right-click the connection for which you want to specify a condition.

Step 2 Select **Conditions**. The Conditions window displays.



Step 3 Specify one of the following conditions (as supported for each task type):

- **Success** -- Run the downstream task if the upstream task goes to Success.
- **Failure** -- Run the downstream task if the upstream task goes to Failure.
- **Success/Failure** -- Run the downstream task if the upstream task goes to Success or Failure.
- **Exit Code(s)** -- Run the downstream task if the upstream task returns one of the specified exit code(s). Format: Numeric. Use commas to list a series of exit codes; use hyphens to specify a range. Example: 1,5, 22-30.

Success conditions are indicated by a solid line in the Workflow Editor; all other conditions display as a dotted line.

Step 4 Click **OK**.

Step 5 To see the condition that is specified for a connection, hover your cursor over the connector.

Dependency Conditions per Task Types

The following table identifies the dependency conditions that are supported for each type of task:

Task Type	Success	Failure	Success/Failure	Exit Code(s)
Workflow	✓			✓
Linux/Unix	✓	✓	✓	✓
Windows	✓	✓	✓	✓
z/OS	✓	✓	✓	✓
Indesca	✓	✓	✓	✓
SAP	✓	✓	✓	✓
File Transfer	✓	✓	✓	✓
Manual	✓			✓
Sleep	✓			✓
SQL	✓	✓	✓	✓

Stored Procedures				
Email				
Task Monitor				
File Monitor				
FTP File Monitor				
System Monitor				
Application Control				

Creating Conditional Paths

Opwise allows you to specify separate processing paths for each condition. For example, you might specify a group of tasks that will run if the upstream task goes to Success and a second group of tasks that will run if the upstream task goes to Failure. When Opwise recognizes that conditional paths have been specified, the tasks in the path whose condition is met run, and the tasks in the path whose condition is not met go to a Skipped status.

Opwise identifies a conditional path when:

1. The upstream task goes to a finished status (Success or Failure).
2. As a result, at least one downstream dependency is satisfied and at least one downstream dependency is not satisfied.

For example, Task A is at the top of the workflow. Three conditional paths have been specified: one for failure and two for success. Task A executes and goes to Success status. Opwise identifies this as a conditional path and puts all the tasks in the failure path into Skipped status; the tasks in the two Success paths begin running normally.

As another example, Task A is at the top of the workflow. Two conditional paths have been specified: one for exit codes 1 through 10, the second for exit codes 11 through 20. If Task A completes with exit code 5, the first path runs and the second path is skipped. If Task A completes with exit code 15, the first path is skipped and the second path runs. If Task A completes with exit code 25, neither condition is satisfied and both paths remain in Waiting status.

Using Multiple Connections

If a task has more than one upstream connection, the task remains in a Waiting status until all of the conditions of those multiple connections are evaluated. If all of the connections are with paths that have been skipped, the task goes to a Skipped status. If at least one of the connections is with a path that has executed and all connections have been evaluated, the task executes.

Moving Workflow Elements

Once you have positioned one or more tasks and connections on the workflow canvas, you can reposition the objects as needed.

Move a Single Task

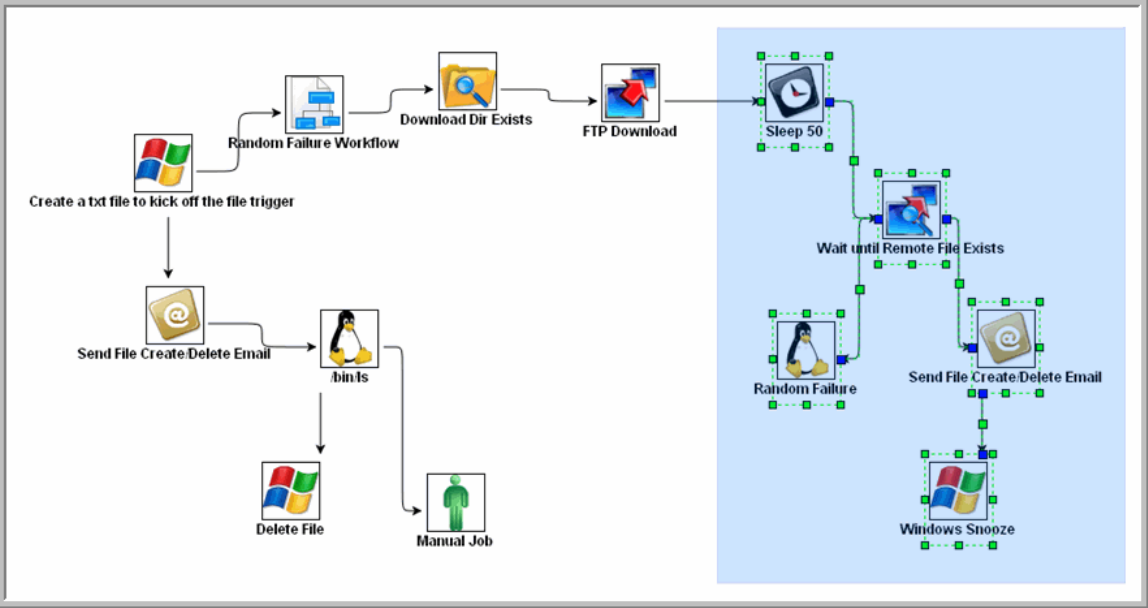
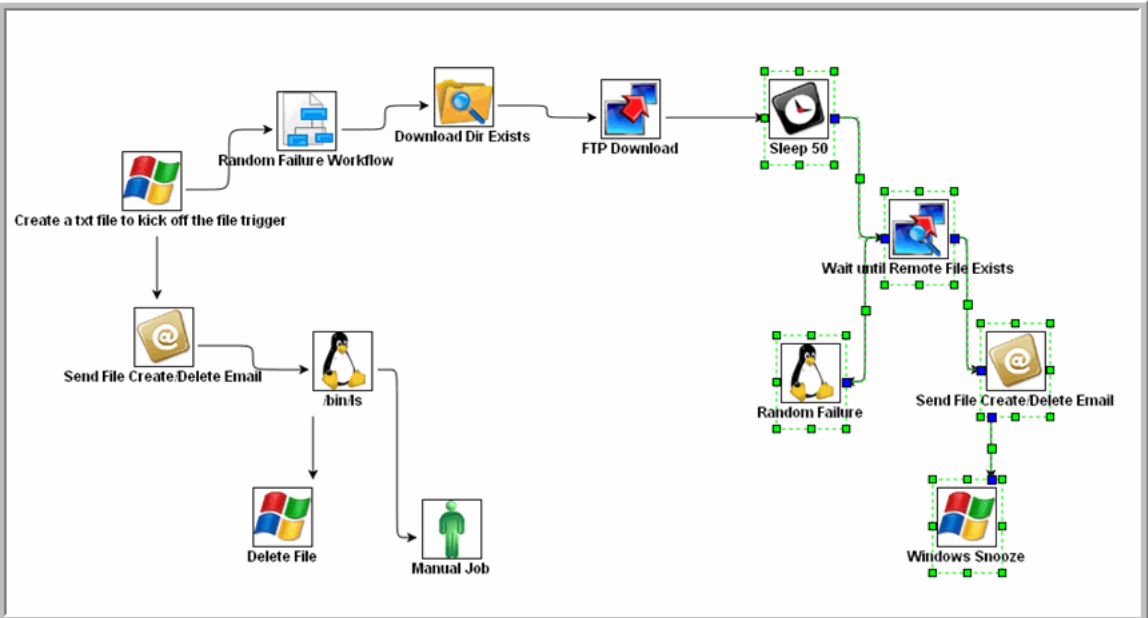
Step 1	Click the Select icon - - to enter Select mode.
Step 2	Click a task and drag it to its new location. If the task has any connectors attached, they remain connected and lengthen or shorten as necessary.

OR



Step 1	Click one of the Connect icons - - to enter Connect mode.
Step 2	Click the end of the connection, near where it connects to the task you want to move.
Step 3	Drag the connection and task to their new location. The connector remains connected and shortens or lengthens as necessary.

Move a Group of Tasks

Step 1	Click the Select icon - - to enter Select mode.
---------------	--


Step 2	Position the cursor near the group you want to select.
Step 3	Click the canvas and begin dragging. A blue shaded box appears. This identifies the selection area.
Step 4	<p>Drag the selection area over the tasks you want to select, as shown in the following example:</p>  <p>The diagram shows a workflow starting with 'Create a txt file to kick off the file trigger'. This leads to 'Send File Create/Delete Email', which then branches to 'Delete File' and 'Manual Job'. Another path goes from 'Send File Create/Delete Email' to 'binits', which leads to 'Delete File' and 'Manual Job'. A separate path starts with 'Random Failure Workflow', leading to 'Download Dir Exists', then 'FTP Download', 'Sleep 50', 'Wait until Remote File Exists', 'Random Failure', 'Send File Create/Delete Email', and 'Windows Snooze'. In the screenshot, a blue shaded box highlights the tasks from 'Sleep 50' to 'Windows Snooze'.</p>
Step 5	<p>When you have selected all the tasks you want to move, release the mouse. The tasks and connectors included in the group are highlighted, as shown:</p>  <p>The diagram is identical to the one in Step 4, but the tasks and connectors within the blue shaded selection box (from 'Sleep 50' to 'Windows Snooze') are now highlighted with a dashed green border, indicating they are selected.</p>
Step 6	Click one of the selected tasks and drag it to the new location. All the selected tasks are moved.
Step 7	To deselect the group of tasks, click elsewhere on the canvas.

OR



Step 1	Click one of the Connect icons -   - to enter Connect mode.
Step 2	Position the cursor near the group you want to select.
Step 3	Click the canvas and begin dragging. A blue shaded box appears. This identifies the selection area.

Step 4	Drag the selection area over the tasks and connectors you want to select. Make sure you select the entire connector(s), or you will not be able to move the objects.
Step 5	When you have selected all the tasks and connectors you want to move, release the mouse. The tasks and connectors included in the group are highlighted.
Step 6	Click one of the selected connectors and drag it to the new location. All the selected tasks and connectors are moved.
Step 7	To deselect the group of tasks and connectors, click elsewhere on the canvas.

Deleting Workflow Elements

Step 1	Highlight the workflow element you want to delete. Or, select a group of elements you want to delete.
Step 2	Click the Delete  icon. The element and all its associated connectors are deleted.

Copying Workflow Elements






Step 1	Highlight the workflow element you want to copy. Or, select a group of elements you want to copy.
Step 2	Click the Copy  icon. The element and all its associated connectors are copied.
Step 3	To paste the copied elements onto the workflow, click Paste  . The copied elements are pasted next to the originals.
Step 4	To move the new elements to a new location, go into Connect mode and click and drag one of the connectors or go into Select mode and click and drag one of the icons. If you have copied multiple elements, do not click elsewhere in the display before dragging the group. Clicking elsewhere deselects the copied elements.

Undoing and Redoing Workflow Changes



To undo the most recent change, click **Undo** . To redo a change you just undid, click **Redo** .


Zooming In and Out

Several features are available for zooming in and out on large workflows:


- Click **Fit**  to fit the entire workflow onto the display.
- Click **Zoom In**  to increase the size of the workflow (to view details).
- Click **Zoom Out**  to decrease the size of the workflow (to view the entire workflow)
- Click **Actual Size**  to return the workflow to its actual (original) size.
- Click **Zoom**  to enter a zoom percentage.

Panning Around in Large Workflows

For large workflows that cannot be displayed entirely on the screen, you can pan around from area to area. Two methods are provided: the **Pan**  icon and the **Outline**  icon.

Step 1	Click the Pan  icon. This enters Pan mode.
Step 2	Click the display and drag the workflow so that it displays the area you want to work on.

OR

Step 1	Click the Outline  icon. The Outline window opens.
Step 2	In the Outline window, move and/or resize the blue box to identify the area of the workflow you want to work on. The display repositions itself as indicated in the Outline window.

**Note**

You also can move to different areas of a workflow by using the [Find in Graph...](#) feature.

Workflow Location Cookies



If you use Pan mode (or [Find in Graph...](#)) to move to different locations of a workflow, Opswise preserves the coordinates of the last location that you moved to using session cookies. When you leave the workflow and return to it - in the same browser session - that last location in the workflow displays.

To restore the displayed workflow location to the default, top-left position, you can either:

- Right-click any white space in the workflow canvas to display a [pop-up menu](#) and click **Pan To Top**.
- Start a new browser session.

Automatically Formatting a Workflow

You can apply automatic formatting to your workflow. This process does not change any connections or content, but reorganizes the workflow into a more legible display.

- To create a horizontal layout, click .
- To create a vertical layout, click .

Displaying Workflow Documentation

Click the **Help**  icon.


Displaying Processing Messages

While a workflow is running, you can click the **Console**  icon to display processing messages. For more details, see [Monitoring Workflows](#).

Saving the Workflow

Click the **Save**  icon.

Switching Between Workflows

You can switch between workflows without using the navigation pane by clicking the **Open Workflow**  icon and selecting the workflow you want to switch to. It is recommended, but not required, that you save your work before switching to another workflow. You can switch back to your original workflow by clicking **Open Workflow** again.

Adding Skip/Run Criteria for Specific Tasks

You can add special instructions that specify conditions under which a specific task (or sub-workflow) within the workflow should be run or skipped. Opswise evaluates these instructions when determining whether to run each task within a workflow.

For example, you might want to skip a specific task in a workflow on a certain day or date, or you might want to run a specific task only if a certain variable is set to a specific value.

Creating New Run Criteria

Step 1	Select the workflow for which you want to specify run criteria.
Step 2	Click the Task Run Criteria tab. The Task Run Criteria List screen displays a list of specified task run criteria.

Step 3 Click **New**. The **Task Run Criteria Definition** screen displays:

The screenshot shows the 'Task Run Criteria' definition screen. It features a title bar with a back arrow, the text 'Task Run Criteria', a red exclamation mark icon, and '= Required field'. A 'Submit' button is in the top right. The form is organized into several sections:

- Type:** A dropdown menu set to 'Run Criteria'.
- Task Id:** A text input field with a magnifying glass icon.
- Vertex Id:** A dropdown menu set to 'Any'.
- Business Day:** A checkbox, currently unchecked.
- Holiday:** A checkbox, currently checked.
- Specific Day(s):** A checkbox, currently checked.
- Custom Day:** A checkbox, currently checked.
- Complex:** A checkbox, currently checked.
- Variable:** A checkbox, currently checked.
- Custom Day Choice:** A text input field with a magnifying glass icon.
- Days of the Week:** A vertical list of checkboxes for Sun, Mon, Tue, Wed, Thu, Fri, and Sat, all currently unchecked.
- Adjective:** A dropdown menu set to 'Last'.
- Noun:** A dropdown menu set to 'Day'.
- Qualifier:** A dropdown menu set to 'Year'.
- Evaluate At:** A dropdown menu set to 'Trigger Time'.
- Name:** A text input field.
- Operator:** A dropdown menu set to '='.
- Value:** A text input field.

A 'Submit' button is located at the bottom left of the form.

Step 4 Using the field descriptions provided below as a guide, complete the fields as needed. For example, to tell Opwise not to run a specific task on business days or holidays, select **Skip Criteria** in the Type field, select the task and enable **Business Day** and **Holiday**.



Step 5 Click the **Submit** button to save the record and return to the Task Run Criteria List screen, or right-click the title bar and select **Save** to save the record and remain on the Task Run Criteria List Definition screen.

Step 6 If appropriate, repeat these steps for any additional run criteria you want to add.

Task Run Criteria Field Descriptions

The following table provides descriptions for each field on the Task Run Criteria screen.

Field Name	Description
Type	User-defined. Indicates whether this is providing instructions on when to run or not run the task. Options: <ul style="list-style-type: none"> • Run Criteria - Provides instructions on when to run the task. • Skip Criteria - Provides instructions on when to skip the task.
Task	User-defined. Select the task for which you are specifying run or skip criteria. Click the magnifying glass to browse for and select a task from the task list.
Vertex ID	Each task within a workflow has a unique vertex ID, which distinguishes it from other tasks of the same name, if any.
Business Day	If enabled, the task runs or skips on all business days.
Holiday	If enabled, the task runs or skips on holidays.
Specific Day(s) – Sunday through Saturday	If enabled, the task runs or skips on the day(s) you select in the right-hand column.

Custom Day	<p>If enabled, the task runs or skips on the day you select in the Custom Day Choice field, below.</p> <div style="background-color: #ffffcc; padding: 10px; margin: 10px 0;"> <p> Note The calendar used for the workflow must contain the custom day that you specify in the Custom Day Choice field; otherwise, the task will not run or skip as expected on the custom day:</p> <ul style="list-style-type: none"> If you launch a workflow manually, the workflow Default Calendar is used. If a Default Calendar is not specified, the System Default calendar is used. If you launch a workflow using a trigger, the calendar specified in the trigger is used. </div>
Custom Day Choice	<p>If Custom Day is enabled, select the custom day for which you are specifying run or skip criteria. Click the magnifying glass to browse for and select a day from the custom day list.</p> <div style="background-color: #ffffcc; padding: 10px; margin: 10px 0;"> <p> Note You cannot use a custom day defined as a period as part of the task run criteria (see Custom Days).</p> </div>
Complex	<p>If enabled, the task runs or skips on the day(s) indicated in the Adjective, Noun, and Qualifier fields.</p>
Adjective	<p>If Complex is enabled, you can use this field to specify which in a series of days you want to select. Used in conjunction with the Noun and the Qualifier fields. For example, to specify "the 15th business day of the month," select Date Adjective: Nth, Date Noun: Business Day, Date Qualifier: Month. Options:</p> <ul style="list-style-type: none"> Every 1st 2nd 3rd 4th Nth Last
Noun	<p>If Complex is enabled, you can use this field to specify the type of day you want to select. Used in conjunction with the Adjective and the Qualifier fields. For example, to specify "the 1st business day of the month," select Adjective: 1st, Noun: Business Day, Qualifier: Month. Options:</p> <ul style="list-style-type: none"> Sunday through Saturday Day = any day Business Day = The business days specified in the calendar selected in the Calendar field. Custom Days specified in the calendar selected in the Calendar field.
Qualifier	<p>If Complex is enabled, you can use this field to specify the period for your selection formula. Used in conjunction with the Noun and the Adjective fields. For example, to specify "the 1st business day of the month," select Adjective: 1st, Noun: Business Day, Qualifier: Month. Options:</p> <ul style="list-style-type: none"> Month Year January through December Custom Period (see Creating Custom Days)
Variable	<p>If enabled, instructs Opwise to run or not run the task, depending on the value of a specific variable. Used in conjunction with fields: Evaluate At, Name, Operator, and Value (see below).</p>
Evaluate At	<p>Allows you to specify when you want Opwise to evaluate the variable. Options:</p> <ul style="list-style-type: none"> Trigger Time – Evaluate the variable when the workflow is triggered. Run Time – Evaluate the variable when the task is about to run.
Name	<p>Specifies the name of the variable being evaluated. The variable does not need to exist when this record is created.</p>
Operator	<p>Allows you to select the operator to be used in the evaluation. Options: =, !=, >, >=, <, <=, regex (regular expression).</p>

Value	Allows you to specify the value or regular expression that Opwise should look for when evaluating the variable. Up to 40 alphanumerics.
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Delete button	Deletes the current record.
Variables tab	Displays all variables associated with this record.


Specifying When a Workflow Runs

As with other task types, you can [run a workflow manually](#) or [specify triggers](#) that run the workflow task automatically based on times or events. For workflows, you can also [specify skip and run criteria](#).

Monitoring Workflow Execution

You can monitor all system activity from the [Activity](#) screen.

Modifying an Existing Workflow

Step 1	From the navigation pane, select Automation Center > Tasks > Workflow Tasks . The Workflow Tasks List screen displays.
Step 2	Click the Task Name of the workflow task that you want to modify. The Workflow Task Definition screen displays.
Step 3	Click Edit Workflow . The Workflow Editor displays.
Step 4	Modify the workflow and click the Save  icon.

Deleting a Workflow

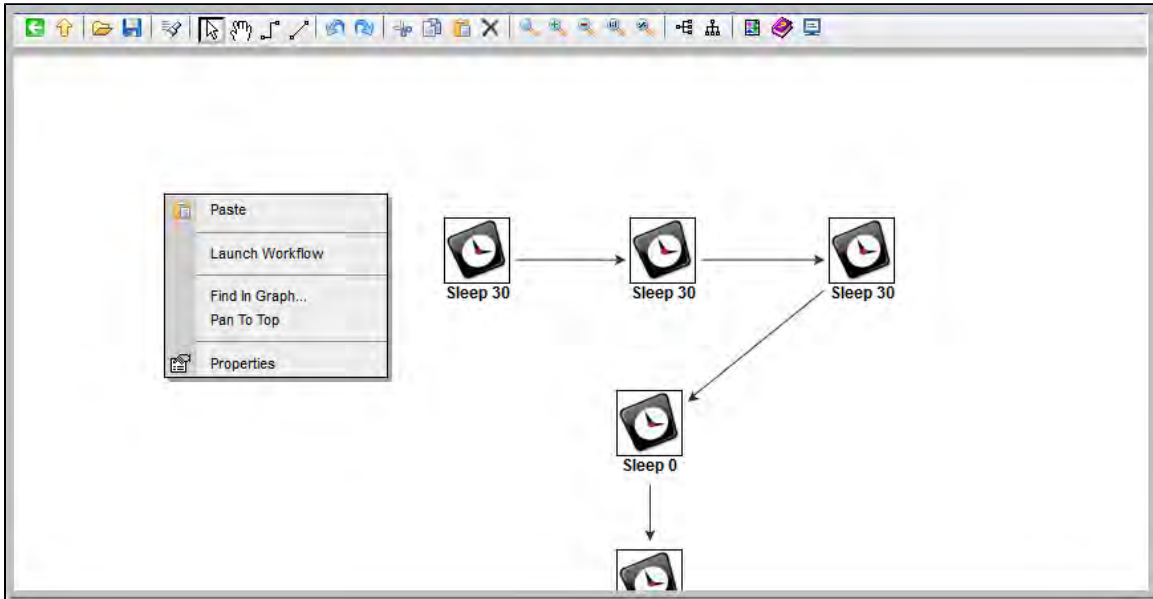
Step 1	From the navigation pane, select Automation Center > Tasks > Workflow Tasks . The Workflow Tasks List screen displays.
Step 2	Click the empty box (in the first column) of the workflow(s) that you want to delete.
Step 3	Select Delete from the Actions on selected rows... menu.
Step 4	To delete all workflows: <ol style="list-style-type: none"> 1. Click the Select All box next to the Actions on selected rows.. menu. 2. In the Actions on selected rows.. menu,click Delete.

Finding a Task in a Workflow

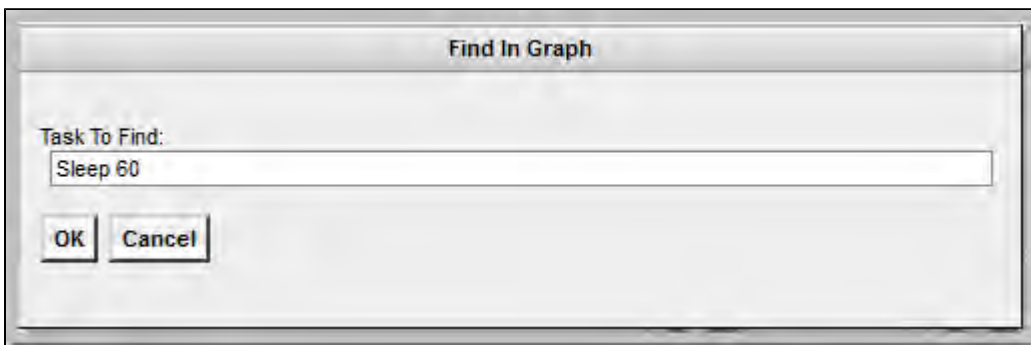
For any workflow task, or any workflow task instance (running or complete), you can find the location of any task within that workflow:

Step 1	From the navigation pane, select Automation Center > Tasks > Workflow Tasks . The Workflow Tasks List screen displays.
Step 2	Select the workflow that you want to search. The Workflow Definition screen displays.
Step 3	Click Edit Workflow. The Workflow Editor displays.

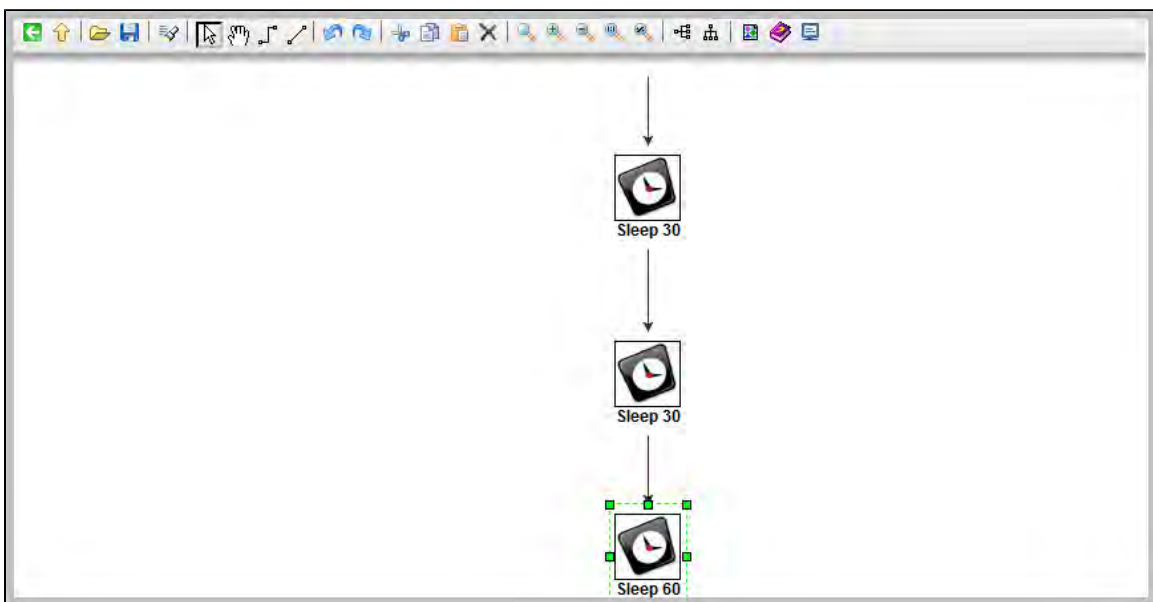
Step 4 Right-click in the Workflow Editor canvas. A pop-up menu displays.



Step 5 Click **Find in Graph...** to display a Find in Graph dialog.



Step 6 Enter the name of the task you want to find and click OK. Opswise locates and displays the task within the workflow.



**Note**

You also can locate a task in a workflow by using [Pan mode](#) feature.

Workflow Location Cookies

If you use **Finding a Task...** (or [Pan mode](#)) to move to different locations of a workflow, Opswise preserves the coordinates of the last location that you moved to using session cookies. When you leave the workflow and return to it - in the same browser session - that last location in the workflow displays.

To restore the displayed workflow location to the default, top-left position, you can either:

- Right-click any white space in the workflow canvas to display a [pop-up menu](#) and click **Pan To Top**.
- Start a new browser session.

Inserting a Task in a Workflow

After a workflow has been launched, you can insert a new task (except a workflow task) into an active workflow instance.

You can insert the task as a predecessor or successor of another task instance within the workflow instance using the [Insert Task as Predecessor](#) or [Insert Task as Successor](#) command, respectively. Alternatively, you can use the [Insert Task...](#) command to insert a task with any number of predecessors and successors.

Insert Task as Predecessor

Step 1	In the Workflow Monitor, right-click a task instance.
Step 2	Click Insert Task As Predecessor.... The Task Insert > Task Selection dialog displays.
Step 3	Find the task you want to insert (see Searching For and Adding Tasks) and drag the task's icon onto the workflow canvas where you would like it to be inserted. Opswise inserts the task as a predecessor to the selected task instance, and the inserted task will start to run when all of its dependencies (for example, virtual resources) have been met. If the selected task instance had not already started, it will wait until the inserted task completes and all of its other dependencies have been satisfied.

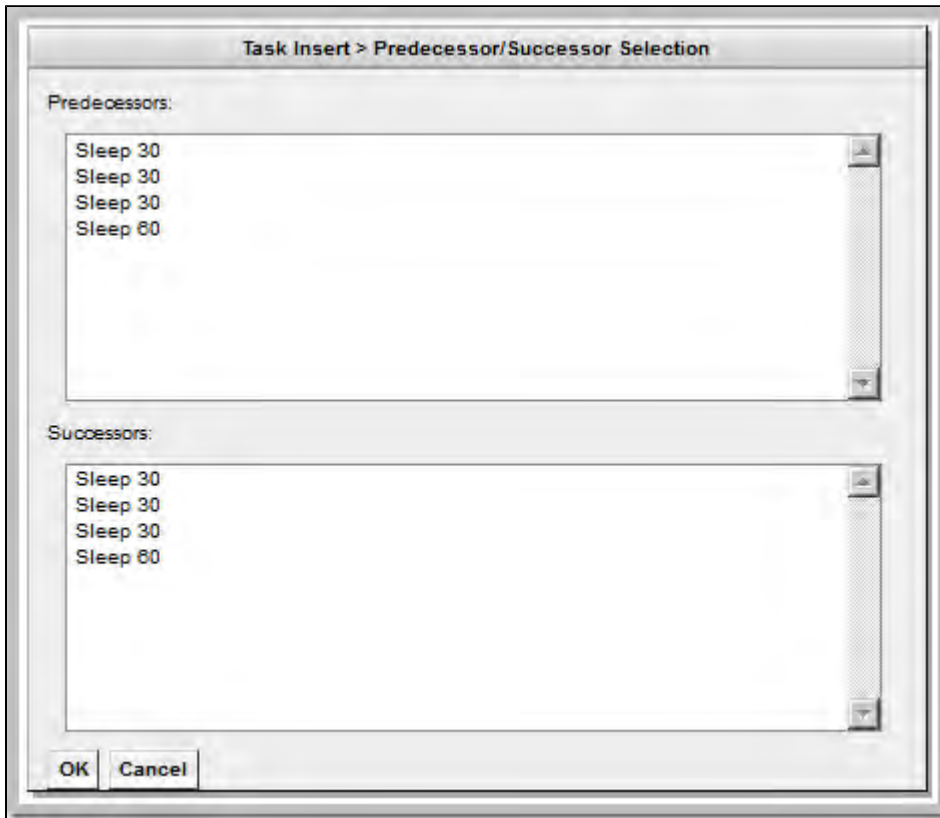
Insert Task as Successor

Step 1	In the Workflow Monitor, right-click a task instance.
Step 2	Click Insert Task As Successor.... The Task Insert > Task Selection dialog displays.
Step 3	Find the task you want to insert (see Searching For and Adding Tasks) and drag the task's icon onto the workflow canvas where you would like it to be inserted. Opswise inserts the task as a successor to the selected task instance. When the selected task instance completes, the inserted task will start to run, assuming that all of its dependencies (for example virtual resources) have been met.

Insert Task with Multiple Predecessors/Successors

Step 1	In the Workflow Monitor, right-click any white space in the workflow canvas.
Step 2	Click Insert Task.... The Task Insert > Task Selection dialog displays.
Step 3	Find the task you want to insert (see Searching For and Adding Tasks) and drag the task's icon onto the workflow canvas where you would like it to be inserted.

Step 4 The Task Insert > Predecessor/Successor Selection dialog displays.



Step 5 Select zero or more Predecessors and zero or more Successors, and then click **OK**.

Opwise inserts the task as a successor to the selected predecessor task instance(s) and as a predecessor to the selected successor task instance(s).

- When the selected predecessor task instances complete, the inserted task will start to run, assuming that all of its dependencies (for example, virtual resources) have been met.
- If the selected successor task instances had not already started, they will wait until the inserted task completes and all of their other dependencies have been satisfied.



Note

Any predecessor task that is in a SUCCESS, FINISHED, or SKIPPED status will not prevent the inserted task from running.

Modifying Tasks in a Workflow

You can make the following modifications for tasks in a workflow:

- Changing the Priority of a Task Instance
- Re-running a Task
- Cancelling a Task
- Force Finishing a Task
- Putting a Task on Hold
- Releasing a Task from Hold
- Skipping a Task
- Showing or Hiding Skipped Tasks
- Unskipping a Task
- Clearing All Predecessor Dependencies for a Task Instance
- Marking a Dependency as Satisfied

Monitoring Task Activity

Opswise lets you perform the following actions related to the monitoring of tasks and Workflows:

- [Monitoring Activity from the Activity Screen](#)
- [Monitoring Activity from the Task Instances Screen](#)
- [Monitoring Activity History](#)
- [Monitoring Workflows](#)
- [Viewing Task Instances for a Specific Task](#)
- [Displaying Task Instance Status](#)
- [Retrieving Output](#)

Monitoring Activity from the Activity Screen

- Overview
- Accessing the Activity Screen
- Changing the Sorting
- Changing the Columns
 - Moving Columns
 - Adding or Removing Columns
- Changing the Refresh Rate
- Activity Screen Column Descriptions
- Changing the Activity Screen Display
 - Selecting a Different Activity Report
 - Creating a New Activity Report
 - Updating, Copying, Deleting Activity Reports
- Displaying Details about Task Instances

Overview

The Activity screen is the Opwise central console of activity, a real-time display of task instance status. It displays a selected group of (or all) task instances, controlled by the Activity Report selected in the drop-down menu at the top of the screen. The selected report also defines what columns are displayed.

A task instance is the "run" version of a task. Each time a task runs, Opwise creates a task instance and monitors its activity on the Activity screen. Each task instance is a separate record.

The Activity screen allows you to issue commands against task instances. You can also issue commands from the [Task Instances](#) screen (and the [Task Instances](#) screen for a specific task). In cases where the task definition did not instruct Opwise to retrieve output automatically, you can retrieve output manually from any completed task.

Accessing the Activity Screen

From the navigation pane, select **Automation Center > Activity**. The Activity screen displays task status information based on the most recently selected report.

The following sample Activity screen uses the report **Today's Task Instances by Created Time**.

Instance Name	Status	Start Time	End Time	Duration	Invoked By
Print DR/CR on Collections	Running	2008-10-19 09:04:21 -0700			Trigger: Run International Tickets
Check for DR/CR on Collections	Running	2008-10-19 09:04:21 -0700			Trigger: Run International Tickets
Create a txt file to kick off the fi...	Undeliverable				Trigger: Run International Tickets
Send File Create/Delete Email	Success	2008-10-19 09:04:21 -0700	2008-10-19 09:04:23 -0700	2 Seconds	Trigger: Run International Tickets
Standby LCs	Running	2008-10-19 09:04:21 -0700			Trigger: Run International Tickets
Fees Outgoing Collections	Success	2008-10-19 09:04:21 -0700	2008-10-19 09:04:36 -0700	15 Seconds	Trigger: Run International Tickets
Welcome Msg Exists	Undeliverable				Trigger: Run International Tickets
Fees Assessment Collections	Running	2008-10-19 09:04:21 -0700			Trigger: Run International Tickets
Print DR/CR on Collections	Success	2008-10-19 09:03:21 -0700	2008-10-19 09:04:21 -0700	1 Minute 0 Seconds	Trigger: Run International Tickets
Check for DR/CR on Collections	Success	2008-10-19 09:03:21 -0700	2008-10-19 09:04:21 -0700	1 Minute 0 Seconds	Trigger: Run International Tickets
Create a txt file to kick off the fi...	Undeliverable				Trigger: Run International Tickets
Send File Create/Delete Email	Success	2008-10-19 09:03:21 -0700	2008-10-19 09:03:23 -0700	2 Seconds	Trigger: Run International Tickets
Standby LCs	Success	2008-10-19 09:03:21 -0700	2008-10-19 09:04:21 -0700	1 Minute 0 Seconds	Trigger: Run International Tickets
Fees Outgoing Collections	Success	2008-10-19 09:03:21 -0700	2008-10-19 09:03:36 -0700	15 Seconds	Trigger: Run International Tickets
Welcome Msg Exists	Undeliverable				Trigger: Run International Tickets
Fees Assessment Collections	Success	2008-10-19 09:03:21 -0700	2008-10-19 09:04:21 -0700	1 Minute 0 Seconds	Trigger: Run International Tickets
Print DR/CR on Collections	Success	2008-10-19 09:02:21 -0700	2008-10-19 09:03:21 -0700	1 Minute 0 Seconds	Trigger: Run International Tickets
Check for DR/CR on Collections	Success	2008-10-19 09:02:21 -0700	2008-10-19 09:03:21 -0700	1 Minute 0 Seconds	Trigger: Run International Tickets
Create a txt file to kick off the fi...	Undeliverable				Trigger: Run International Tickets
Send File Create/Delete Email	Success	2008-10-19 09:02:21 -0700	2008-10-19 09:02:23 -0700	2 Seconds	Trigger: Run International Tickets
Standby LCs	Success	2008-10-19 09:02:21 -0700	2008-10-19 09:03:21 -0700	1 Minute 0 Seconds	Trigger: Run International Tickets
Fees Outgoing Collections	Success	2008-10-19 09:02:21 -0700	2008-10-19 09:02:36 -0700	15 Seconds	Trigger: Run International Tickets
Welcome Msg Exists	Undeliverable				Trigger: Run International Tickets
Fees Assessment Collections	Success	2008-10-19 09:02:21 -0700	2008-10-19 09:03:21 -0700	1 Minute 0 Seconds	Trigger: Run International Tickets
Print DR/CR on Collections	Success	2008-10-19 09:01:21 -0700	2008-10-19 09:02:21 -0700	1 Minute 0 Seconds	Trigger: Run International Tickets

Grid Refresh Rate: 5

Changing the Sorting

You can change the sorting on the display and add or remove columns.

Step 1 Right-click on the title of the column you want to sort by. A pop-up menu displays.

Instance Name	Type	Status	Exit Code	Start Time	End Time
AXA Demo Linux Task	Linux/Unix	cess	0	2012-02-23 18:00:00 -0500	2012-02-23 18:00:04 -05
AXA Demo Linux Task	Linux/Unix	cess	0	2012-02-23 17:30:00 -0500	2012-02-23 17:30:03 -05
AXA Demo Linux Task	Linux/Unix	cess	0	2012-02-23 17:00:00 -0500	2012-02-23 17:00:02 -05
AXA Demo Linux Task	Linux/Unix	Suc		2012-02-23 16:30:00 -0500	2012-02-23 16:30:03 -05
AXA Demo Linux Task	Linux/Unix	Suc		2012-02-23 16:00:01 -0500	2012-02-23 16:00:02 -05
AXA Demo Linux Task	Linux/Unix	Suc		2012-02-23 15:30:00 -0500	2012-02-23 15:30:00 -05
A CS Demo	Linux/Unix	Suc		2012-02-23 15:27:33 -0500	2012-02-23 15:27:36 -05
A CS Workflow	Workflow	Rur		2012-02-23 15:17:26 -0500	
A CS Demo	Linux/Unix	Suc		2012-02-23 15:17:27 -0500	2012-02-23 15:17:27 -05
ABR_job_1	Windows	Suc		2012-02-23 15:18:19 -0500	2012-02-23 15:18:19 -05
Training - Infitran Task 1	File Transfer	Wa			
Training - Infitran Task 2	File Transfer	Ski			2012-02-23 15:17:26 -05
ACCTG 2000	Windows	Ski			2012-02-23 15:17:27 -05
#Accounting	Workflow	Rur		2012-02-23 15:18:19 -0500	
ACCTG 1000 Starting	Linux/Unix	Suc		2012-02-23 15:18:19 -0500	2012-02-23 15:18:22 -05
Payroll DB Query	SQL	Suc		2012-02-23 15:18:27 -0500	2012-02-23 15:18:28 -05
ACCTG 2010 WED Only	Windows	Ski			2012-02-23 15:17:26 -05
ACCTG 2222 LAST Day of Fical Year	Windows	Ski			2012-02-23 15:17:26 -05
ACCTG 1020 FRIDAY	Linux/Unix	Ski			2012-02-23 15:17:26 -05
ACCTG 2000	Windows	Suc		2012-02-23 15:18:26 -0500	2012-02-23 15:18:26 -05
ACCTG 1030 3rd TUESDAY	Linux/Unix	Skipped			2012-02-23 15:17:26 -05
#General Ledger Processing	Workflow	Success	0	2012-02-23 15:18:25 -0500	2012-02-23 15:23:52 -05
GENLGR 3005	Linux/Unix	Success	0	2012-02-23 15:22:28 -0500	2012-02-23 15:23:18 -05
GENLGR 3000	Linux/Unix	Success	0	2012-02-23 15:18:28 -0500	2012-02-23 15:18:59 -05
GENLGR Accruals	Windows	Success	0	2012-02-23 15:18:27 -0500	2012-02-23 15:18:27 -05

Step 2 To change the sorting, click **Sort Ascending** or **Sort Descending**.

Changing the Columns

You can move columns around on the screen or add or remove columns.

Moving Columns

Step 1 Highlight the column you want to move.

Step 2 Drag it to a new location. The arrows indicate where the column will be placed, as shown.

The screenshot shows a table titled "Today's Task Instances by Created Time". The table has columns for Instance Name, Type, Status, Start Time, End Time, and Duration. A context menu is open over the "Instance Name" column header, with "Start Time" selected. The table contains 21 rows of data, all with a status of "Success".

Instance Name	Type	Status	Start Time	End Time	Duration
5x5 linux	Workflow	Success	2008-05-19 00:00:01 -0700	2008-05-19 12:29:17 -0700	12 Hours 29 Minutes
5x5 linux	Workflow	Success	2008-05-19 00:10:01 -0700	2008-05-19 00:11:06 -0700	1 Minute 5 Seconds
5x5 linux	Workflow	Success	2008-05-19 00:20:01 -0700	2008-05-19 00:21:10 -0700	1 Minute 9 Seconds
5x5 linux	Workflow	Success	2008-05-19 00:30:01 -0700	2008-05-19 00:31:09 -0700	1 Minute 8 Seconds
5x5 linux	Workflow	Success	2008-05-19 00:40:01 -0700	2008-05-19 00:41:07 -0700	1 Minute 6 Seconds
5x5 linux	Workflow	Success	2008-05-19 00:50:01 -0700	2008-05-19 00:51:11 -0700	1 Minute 10 Seconds
5x5 linux	Workflow	Success	2008-05-19 01:00:01 -0700	2008-05-19 01:01:10 -0700	1 Minute 9 Seconds
5x5 linux	Workflow	Success	2008-05-19 01:10:01 -0700	2008-05-19 01:11:09 -0700	1 Minute 8 Seconds
5x5 linux	Workflow	Success	2008-05-19 01:20:01 -0700	2008-05-19 01:21:13 -0700	1 Minute 12 Seconds
5x5 linux	Workflow	Success	2008-05-19 01:30:01 -0700	2008-05-19 01:31:12 -0700	1 Minute 11 Seconds
5x5 linux	Workflow	Success	2008-05-19 01:40:01 -0700	2008-05-19 01:41:11 -0700	1 Minute 10 Seconds
5x5 linux	Workflow	Success	2008-05-19 01:50:01 -0700	2008-05-19 01:51:14 -0700	1 Minute 13 Seconds
5x5 linux	Workflow	Success	2008-05-19 02:00:01 -0700	2008-05-19 02:01:13 -0700	1 Minute 12 Seconds
5x5 linux	Workflow	Success	2008-05-19 02:10:01 -0700	2008-05-19 02:11:22 -0700	1 Minute 21 Seconds
5x5 linux	Workflow	Success	2008-05-19 02:20:01 -0700	2008-05-19 02:21:16 -0700	1 Minute 15 Seconds
5x5 linux	Workflow	Success	2008-05-19 02:30:01 -0700	2008-05-19 02:31:15 -0700	1 Minute 14 Seconds
5x5 linux	Workflow	Success	2008-05-19 02:40:01 -0700	2008-05-19 02:41:13 -0700	1 Minute 12 Seconds
5x5 linux	Workflow	Success	2008-05-19 02:50:01 -0700	2008-05-19 02:51:16 -0700	1 Minute 15 Seconds
5x5 linux	Workflow	Success	2008-05-19 03:00:01 -0700	2008-05-19 03:01:06 -0700	1 Minute 5 Seconds
5x5 linux	Workflow	Success	2008-05-19 03:10:01 -0700	2008-05-19 03:11:04 -0700	1 Minute 3 Seconds
5x5 linux	Workflow	Success	2008-05-19 03:20:01 -0700	2008-05-19 03:21:08 -0700	1 Minute 7 Seconds
5x5 linux	Workflow	Success	2008-05-19 03:30:01 -0700	2008-05-19 03:31:11 -0700	1 Minute 10 Seconds
5x5 linux	Workflow	Success	2008-05-19 03:40:01 -0700	2008-05-19 03:41:11 -0700	1 Minute 10 Seconds
5x5 linux	Workflow	Success	2008-05-19 03:50:01 -0700	2008-05-19 03:51:09 -0700	1 Minute 8 Seconds
5x5 linux	Workflow	Success	2008-05-19 04:00:01 -0700	2008-05-19 04:01:08 -0700	1 Minute 7 Seconds

Step 3 Drop the column when the arrows are pointing to the location you want.

Adding or Removing Columns

- Step 1** Right-click on any column title.
- Step 2** Click **Columns**.
- Step 3** Select or deselect the columns you want to add or remove, respectively.

To make more considerable changes to the column display, update the report.

Changing the Refresh Rate

The default refresh rate on the Activity display is every 5 seconds.

To change the rate:

- Step 1** Click **Set Rate**.

The dialog box is titled "Rate" and contains the text "Enter refresh rate in seconds:". Below the text is a text input field. At the bottom of the dialog are two buttons: "OK" and "Cancel".
- Step 2** Enter the new rate in seconds, and click **OK** to save or **Cancel** to cancel the change.

Activity Screen Column Descriptions

The following table describes the columns of information displayed on the Activity screen.

Column Name	Description
Instance Name	Required. Name used within Opwise to identify this task. Up to 40 alphanumeric; variables supported. It is the user's responsibility to develop a workable naming scheme for tasks.
Status	Current status of this task instance.
Invoked By	How the task instance was launched. One of the following: <ul style="list-style-type: none"> • Trigger: (Trigger Name) - The instance was launched by the named trigger. • Workflow: (Workflow Name) - The instance was launched by the named workflow. • Manually Launched - The instance was launched by a user. To determine the name of the user: <ol style="list-style-type: none"> 1. From the Activity or Task Instances screen, click the task instance name to open the record. 2. The Execution User field identifies the user who launched the task instance.
Start Time	Date and time the task instance started.
End Time	Date and time the task instance ended.
Type	Type of task instance.
Duration	Amount of time the task took to run
Agent	Required. Name used within Opwise Automation Center to identify this resource. Up to 40 alphanumeric. It is the user's responsibility to develop a workable naming scheme for resources.
Member of Business Services	User-defined. Allows you to select one or more Business Services that this record definition belongs to. Click the lock icon to unlock the field and select Business Services .

Changing the Activity Screen Display

You can change the list of task instances displayed on the Activity screen by selecting a different Activity Report or creating a new report.

Selecting a Different Activity Report

To select a different Activity Report, click a report in the drop-down menu at the top of the screen. The following default reports are available:

Active Task Instances — Displays all instances with an active status, including: Defined, Waiting, Held, Waiting for Resources, Action Required, Queued, Started, Running, In Doubt.

Active Workflow Task Instances — Displays all workflow instances with an active status, including: Defined, Waiting, Held, Waiting for Resources, Action Required, Queued, Started, Running, In Doubt.

Active/Late Task Instances — Displays all instances with a started late flag set to true and an active status, including: Defined, Waiting, Held, Waiting for Resources, Action Required, Queued, Started, Running, In Doubt.

All Task Instance(s) by Status — Displays all instances, sorted by status.

Cancelled Task Instances — Displays all instances with a status of Cancelled.

Held Task Instance(s) — Displays all instances that were created on the current date and have a status of Held.

In Doubt Task Instances — Displays all task instances with a status of In Doubt.

Queued Task Instances — Displays all instances with a status of Queued.

Running Task Instances — Displays all instances with a status of Running.

Task Instances Due to Finish in the Next 3 Hours — For tasks where [forecasting](#) information is available, lists all tasks due to finish in the next three hours.

Task Instances Due to Finish in the Next Hour — For tasks where [forecasting](#) information is available, lists all tasks due to finish in the next hour.

Today's Failed Task Instances by Status — Displays all instances that were created on the current date and have one of the following statuses: Failed, Cancelled, Start Failure.

Today's Successful Task Instances — Displays all instances that were created on the current date and have a status of Success.

Today's Task Instance(s) by Created Time — Displays all instances that were created on the current date, sorted by created time.

Today's Task Instance(s) by Type — Displays all instances that were created on the current date, sorted by type.

Undeliverable Task Instances — Displays all instances with a status of Undeliverable (agent not available).

Unsuccessful Task Instances — Displays all instances with a status of Failed, Cancelled, Start Failure, Finished.

Waiting for Resources Task Instances — Displays all instances with a status of Waiting for Resources.

Waiting Task Instances — Displays all instances with a status of Waiting.

Workflow Task Instances — Displays all workflows, not including the task instances within the workflows.

Workflow Task Instances with Problems — Displays all workflows with a status of Running Problems, not including the task instances within the workflows.

Creating a New Activity Report

Activity Reports select data about task instances from the Opswise Activity table (`ops_exec`).

To create an Activity Report, you use the same form that is used to create [reports](#). When you are generating normal reports, you can use any [table](#) from the Opswise database; when you create Activity Reports, you only use the Activity table.

Step 1	From the Activity Report screen, click New Report . The New Report screen displays.
Step 2	For an Activity Report, you must specify the following: <ul style="list-style-type: none"> • For Type, specify List. • For the Table field, select Activity ops_exec; otherwise, the saved report will not appear in the Activity Reports drop-down.
Step 3	Specify the Activity Report parameters. For assistance in selecting columns, refer to the field descriptions provided with each task type.
Step 4	To save the new Activity Report and add it to the drop-down, click Save . The report runs immediately and displays in the lower half of the screen. The new Activity Report is also added to the Activity Report drop-down menu.

Updating, Copying, Deleting Activity Reports

All Activity Reports are stored under **Automation Center > Reports**, under the **Activity** heading within one of the following sections:

- My Saved reports (if Visible to = Me)
- My Groups' reports (if Visible to = Automation Center_SysAdmin)
- Global reports (if Visible to = Everyone)

To update, copy, or delete an Activity Report, select **Automation Center > Reports** and scroll to the Activity section where the report is stored, as described above.

Updating an Activity Report

Step 1	Display the report you want to modify.
Step 2	Make your changes and click Update .

Copying an Activity Report

Step 1	Display the report you want to copy.
Step 2	Give the report a new name, specify your changes, if any, and click Insert .

Deleting an Activity Report

Step 1	Display the report you want to delete.
Step 2	Click Delete .

Displaying Details about Task Instances

To view details about any task instance displayed on the Activity screen, click the **Instance Name** of that task instance.

- If the task is a workflow, the [workflow monitor](#) opens. If you want to view record details about the workflow, such as status description, right-click in the workflow monitor white space and select **Properties** to display the [task instance screen](#) for that workflow.
- If the task is not a workflow, Opswise displays the task instance details on the [task instance screen](#) for that task.

Monitoring Activity from the Task Instances Screen

- Overview
- Accessing the Task Instances Screen
 - Task Instances Screen Column Descriptions
- Issuing Commands Against Task Instances
- Sorting and Filtering
- Accessing Task Instance Details
 - View Parent Workflow
 - View Output
 - View Complete Details
 - Viewing Specific Details

Overview

The Task Instances screen displays the same information as the [Activity screen](#), but only for task instances for which there has been a status change or a modification to the task instance record within the last 7 days (an **Updated on Last 7 Days** filter has been pre-selected for this display).

Unlike the Activity screen, the display is not automatically refreshed. However, the Task Instances screen lets you temporarily modify displayed information using extensive filtering capabilities.

Just as with the Activity screen, the Task Instances screen allows you to issue commands against multiple task instances and view details about task instances.

If you select a workflow instance from the Task Instances screen, you are immediately taken to the workflow instance screen, whereas, if you select a workflow instance from the Activity screen, you are immediately taken to the Workflow Monitor. To view the Workflow Monitor from the workflow instance screen, click the **View Workflow** button. To view the workflow instance screen from the Workflow Monitor, right-click on any white-space and select **Properties**.

You also can monitor activity for a specific task by displaying a [task-specific Task Instances screen](#).

Accessing the Task Instances Screen

From the navigation pane, select **Automation Center > Task Instances > Task Instances**. The Task Instances screen displays.

Instance Name	Reference Id	Type	Status	Invoked By	Agent	Start Time	End Time
Opwise - Linux 2x Sub Workflow	904	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:42:00 -0700	2013-09-16 13:42:07 -0700
Opwise - Linux 2x Sub Workflow	905	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:43:00 -0700	2013-09-16 13:43:07 -0700
Opwise - Linux 2x Sub Workflow	906	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:44:00 -0700	2013-09-16 13:44:07 -0700
Opwise - Linux 2x Sub Workflow	907	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:45:00 -0700	2013-09-16 13:45:07 -0700
Opwise - Linux 2x Sub Workflow	908	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:46:00 -0700	2013-09-16 13:46:07 -0700
Opwise - Linux 2x Sub Workflow	909	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:47:00 -0700	2013-09-16 13:47:07 -0700
Opwise - Linux 2x Sub Workflow	910	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:48:00 -0700	2013-09-16 13:48:07 -0700
Opwise - Linux 2x Sub Workflow	911	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:49:00 -0700	2013-09-16 13:49:07 -0700
Opwise - Linux 2x Sub Workflow	912	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:50:00 -0700	2013-09-16 13:50:07 -0700
Opwise - Linux 2x Sub Workflow	913	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:51:00 -0700	2013-09-16 13:51:07 -0700
Opwise - Linux 2x Sub Workflow	914	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:52:00 -0700	2013-09-16 13:52:07 -0700
Opwise - Linux 2x Sub Workflow	915	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:53:00 -0700	2013-09-16 13:53:07 -0700
Opwise - Linux 2x Sub Workflow	916	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:54:00 -0700	2013-09-16 13:54:07 -0700
Opwise - Linux 2x Sub Workflow	917	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:55:00 -0700	2013-09-16 13:55:07 -0700
Opwise - Linux 2x Sub Workflow	918	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:56:00 -0700	2013-09-16 13:56:07 -0700
Opwise - Linux 2x Sub Workflow	919	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:57:00 -0700	2013-09-16 13:57:07 -0700
Opwise - Linux 2x Sub Workflow	920	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:58:00 -0700	2013-09-16 13:58:07 -0700
Opwise - Linux 2x Sub Workflow	921	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 13:59:00 -0700	2013-09-16 13:59:07 -0700
Opwise - Linux 2x Sub Workflow	922	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 14:00:00 -0700	2013-09-16 14:00:07 -0700
Opwise - Linux 2x Sub Workflow	923	Workflow	SUCCESS	Trigger: subwflw		2013-09-16 14:01:00 -0700	2013-09-16 14:01:07 -0700

Task Instances Screen Column Descriptions

The following table describes the columns of information displayed on the Task Instances screen.

Column	Description
Instance Name	Required. Name used within Opwise to identify this task. Up to 40 alphanumeric; variables supported. It is the user's responsibility to develop a workable naming scheme for tasks.
Type	Type of task instance.
Status	Current status of the task instance.
Invoked By	System-supplied. How the task instance was launched. One of the following: <ul style="list-style-type: none"> • Trigger: (Trigger Name) - The instance was launched by the named trigger. • Workflow: (Workflow Name) - The instance was launched by the named workflow. • Manually Launched - The instance was launched by a user. To determine the name of the user: <ol style="list-style-type: none"> 1. From the Activity or Task Instances screen, click the task instance name to open the record. 2. The Execution User field identifies the user who launched the task instance.
Agent	Required. Name used within Opwise Automation Center to identify this resource. Up to 40 alphanumeric. It is the user's responsibility to develop a workable naming scheme for resources.
Start Time	Date and time the task instance started.
End Time	Date and time the task instance ended.

Issuing Commands Against Task Instances

Where applicable, you can manually intervene in processing by [issuing a command](#) against one or more task instances.

For information about the commands available for each type of task, see [Supported Commands](#).

Sorting and Filtering

For information about sorting, filtering, and other list options, see [Using Lists](#).

Accessing Task Instance Details

To display detailed information about a task instance, click the [Instance Name](#) of that task instance. The Task Instance screen for that task instance displays (for example):

The screenshot displays the 'Sleep Task Instance' details page. At the top, there are navigation buttons: 'Update', 'Show Details', 'Re-run', and 'Delete'. The main content area contains the following fields:

- Instance Name:** Sleep 30 - X
- Invoked By:** Trigger: aaa
- Task:** Sleep 30 - \${foo}
- Member of Business Services:** (empty)
- Instance Reference Id:** 5
- Sleep Type:** Seconds
- Sleep Time (secs):** 30
- Hold Reason:** (empty)
- Task Description:** Sleep for 30 seconds. - Y
- Status:** SUCCESS
- Status Description:** (empty)
- Start Time:** 2012-05-10 09:43:09 -0700
- End Time:** 2012-05-10 09:43:39 -0700
- Duration:** 30 Seconds

At the bottom of the form, there are four buttons: 'Update', 'Show Details', 'Re-run', and 'Delete'.

View Parent Workflow

If the task was triggered by a workflow, you can view the parent workflow task instance screen by clicking the **View Parent** button. To display the workflow monitor, click **View Workflow**.

View Output

If the task instance is complete and it generated any output, you can view the output by clicking the **Output** tab.

View Complete Details

To view complete details about the task instance, click the **Show Details** button. Opswise opens a new browser tab and displays a table containing all the detailed elements for this task instance.

For a description of the data being displayed, see the documentation for that task type.

Task Instance Sleep 30 - X

attempt_count	1
avg_estimated_end	2012-05-10 16:43:39
calendar	77171434c0a801c9016d5b2b5d17ddee
cpu_time	0
credentials_var_check	false
duration	30000
duration_seconds	30
early_finish	false
ef_enabled	false
ef_type	TIME
end_time	2012-05-10 16:43:39
execution_user	ops.admin
exit_code	0
forced	false
high_estimated_end	2012-05-10 16:43:39
invoked_by	Trigger: aaa
io_other	0
io_reads	0
io_writes	0
late_finish	false
late_start	false
lf_enabled	false
lf_type	TIME
low_estimated_end	2012-05-10 16:43:39
ls_enabled	false
ls_type	TIME
memory_peak	0
memory_used	0
name	Sleep 30 - X
priority	MEDIUM

res_state	Initial
retry_counter	0
retry_indefinitely	false
retry_interval	60
retry_maximum	0
run_called	true
run_criteria_rt	false
run_criteria_tt	false
security_name	Sleep 30 -
sleep_amount	30
sleep_duration	1970-01-01 00:00:00
sleep_type	SECONDS
start_held	false
start_time	2012-05-10 16:43:09
status_code	SUCCESS
summary	Sleep for 30 seconds. - Y
sys_class_name	ops_exec_sleep
sys_created_by	ops.admin
sys_created_on	2012-05-10 16:43:09
sys_id	37a1f397d861e5e401113e0ef2c0b4f4
sys_mod_count	3
sys_updated_by	ops.admin
sys_updated_on	2012-05-10 16:43:39
task_id	410d696bc0a801c9017e5dbf756ecbd5
task_ref_count	5
trigger_id	1877267bd861e5e400cd0a360984a0ce
type	Sleep
wait_for_resources	false

Close the browser to return to the Task Instance screen.

Viewing Specific Details

To view specific details about the task instance, click the down-arrow on the left-hand side of the title bar, or right-click in the title bar, to display a menu of commands and options for this task instance.

The command and options available on the menu depends on the status of the task instance

View Notes

Click **View Notes** to display the list of notes for that task. (See [Creating Notes](#) for information on how to create notes for tasks and scripts.)

View Exclusive Requests

Click **View Exclusive Requests** to view the list of the records in the Outstanding Exclusive Request (`ops_exclusive_order`) table for that selected task instance.

This option is available only when the Exclusive state is either:

- Requested (Exclusive Requests will have a status of `Pending`.)
- Acquired (Exclusive Requests will have a status of `Filled`.)

View Resource Dependencies

Click **View Resource Dependencies** to view the list of the records in the Task Instance Virtual Resources (`ops_exec_to_resource`) table for that selected task instance.

This option is available only when the Resource state is either:

- Initial
- Returned

View Resource Requests

Click **View Resource Requests** to view the list of the records in the Outstanding Requests (`ops_resource_order`) table for that selected task instance.

This option is available only when the Resource state is:

- Requested

View Resources in Use

Click **View Resources in Use** to view the list of the records in the Currently in Use By (`ops_resource_usage`) table for that selected task instance.

This option is available only when the Resource state is:

- Acquired


Monitoring Activity History

- [Overview](#)
- [Displaying the History Screen](#)
- [Displaying Details from the History Screen](#)

Overview

The History screen provides an historical display of all completed task activity. Only task instances with a status in an "end state" (SUCCESS, FINISHED, FAILED, CANCELLED, START FAILURE, SKIPPED) display on the History screen.

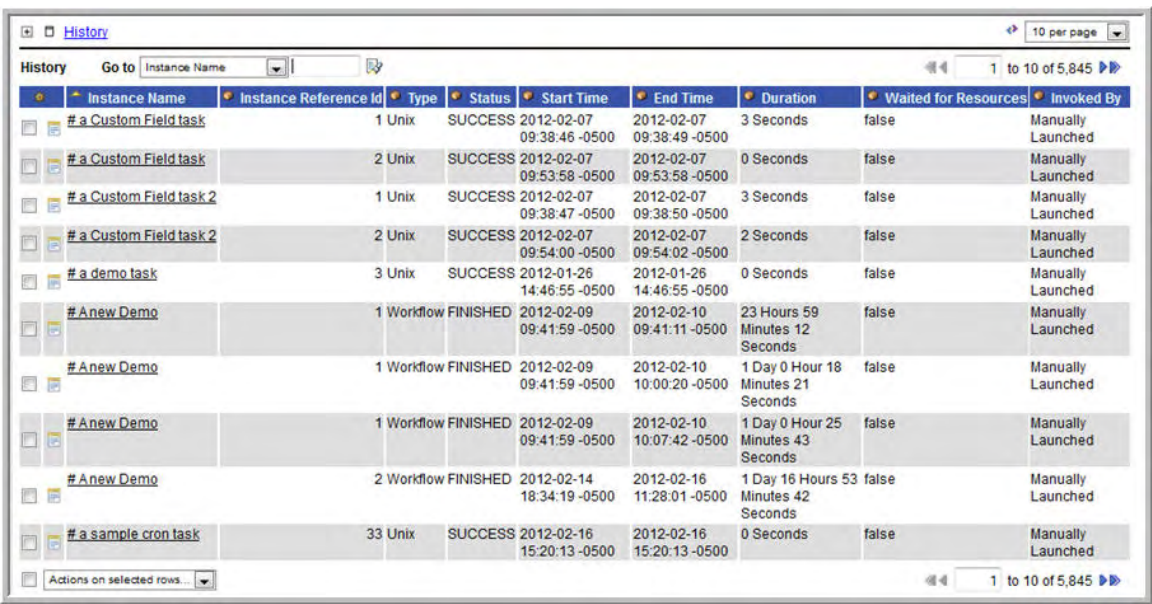
This screen allows you to track information about specific task instances, including multiple runs. For example, Task A may have failed and then was re-run by a user. This task instance will display twice on the History screen, first for the time that it ran and failed and again for the time it was re-run to success.

 **Note**
 If you want to display task activity for all tasks, for task instances in any status, and issue commands against those task instances, see the [Activity](#) screen and/or [Task Instances](#) screen

If you want to display task activity for a specific task, for task instances in any status, and issue commands against those task instances, see the [Viewing Task Instances for a Specific Task](#).

Displaying the History Screen

Step 1 From the navigation pane, select **Automation Center > Task Instances > History**. Opswise displays the History screen.



Instance Name	Instance Reference Id	Type	Status	Start Time	End Time	Duration	Waited for Resources	Invoked By
# a Custom Field task	1	Unix	SUCCESS	2012-02-07 09:38:46 -0500	2012-02-07 09:38:49 -0500	3 Seconds	false	Manually Launched
# a Custom Field task	2	Unix	SUCCESS	2012-02-07 09:53:58 -0500	2012-02-07 09:53:58 -0500	0 Seconds	false	Manually Launched
# a Custom Field task 2	1	Unix	SUCCESS	2012-02-07 09:38:47 -0500	2012-02-07 09:38:50 -0500	3 Seconds	false	Manually Launched
# a Custom Field task 2	2	Unix	SUCCESS	2012-02-07 09:54:00 -0500	2012-02-07 09:54:02 -0500	2 Seconds	false	Manually Launched
# a demo task	3	Unix	SUCCESS	2012-01-26 14:46:55 -0500	2012-01-26 14:46:55 -0500	0 Seconds	false	Manually Launched
# Anew Demo	1	Workflow	FINISHED	2012-02-09 09:41:59 -0500	2012-02-10 09:41:11 -0500	23 Hours 59 Minutes 12 Seconds	false	Manually Launched
# Anew Demo	1	Workflow	FINISHED	2012-02-09 09:41:59 -0500	2012-02-10 10:00:20 -0500	1 Day 0 Hour 18 Minutes 21 Seconds	false	Manually Launched
# Anew Demo	1	Workflow	FINISHED	2012-02-09 09:41:59 -0500	2012-02-10 10:07:42 -0500	1 Day 0 Hour 25 Minutes 43 Seconds	false	Manually Launched
# Anew Demo	2	Workflow	FINISHED	2012-02-14 18:34:19 -0500	2012-02-16 11:28:01 -0500	1 Day 16 Hours 53 Minutes 42 Seconds	false	Manually Launched
# a sample cron task	33	Unix	SUCCESS	2012-02-16 15:20:13 -0500	2012-02-16 15:20:13 -0500	0 Seconds	false	Manually Launched

Step 2 You can modify the display by filtering, sorting, adding, and removing columns, and so on (see [Using Lists](#)).

Displaying Details from the History Screen

To display execution details about a task instance on the History screen, click the underlined identifier in the left-most column. Opswise displays the execution details.

History		Delete	
Instance Name:	# a Custom Field task	Task:	# a Custom Field task
Instance Reference Id:	1	Trigger:	
Task Instance:		Agent:	abdfapp-demo
Type:	Unix	Invoked By:	Manually Launched
Status:	SUCCESS	Start Time:	2012-02-07 09:38:46 -0500
Exit Code:	0	End Time:	2012-02-07 09:38:49 -0500
Status Description:		Duration:	3 Seconds
Task Description:	Updated per J.Mange		
Created by:	glide.maint		
Delete			

Monitoring Workflows

- Overview
- Monitoring a Running Workflow
- Workflow Monitor Display Mode
 - Task Name
 - Task Type
 - Task Status
 - Changing a Task Status Color
 - Displaying Task Status Icons
- Manually Intervening in a Workflow
- Displaying Processing Information

Overview

Opswise allows you to monitor running workflows in graphical format. As the workflow progresses, the display provides up-to-date textual and color-coded status information for each task instance in the workflow. You can also intervene in processing where necessary.

Monitoring a Running Workflow



Note

A workflow already must be running in order for you to monitor its status.

Step 1	From the navigation pane, select Automation Center > Task Instances > Activity .
Step 2	Locate the workflow you want to monitor.
Step 3	Click on the underlined Instance Name of the workflow. The Workflow Monitor displays, as shown in the following example.

```

graph TD
    A[Sleep 20  
Success] --> B[Workflow of Sleep Tasks  
Running]
    A --> C[Windows File Exists Check  
Skipped]
    C --> D[Delete File  
Undeliverable]
    C --> E[Send File Create/Delete Email  
Running]
    D --> F[Manual Job  
Waiting]
    
```

Workflow Monitor Display Mode

The default display mode for the Workflow Monitor identifies each task instance in the workflow by:


















- Task name
- Task type
- Task status

Task Name

The task name is the name given to the task when it was created (see [Creating Tasks](#)).

Task Type

The task type of each task is represented by an icon that displays above the task name.

 Application Control	 Email	 File Monitor	 File Transfer
 FTP File Monitor	 Indesca	 Linux/Unix	 Manual
 SAP	 Sleep	 SQL	 Stored Procedure
 System Monitor	 Task Monitor	 Windows	 Workflow
 z/OS			



Note

You can change the Workflow Monitor so that the icon above each task name represents the current status of that task instance (see [Displaying Task Status Icons](#)).

Task Status

The current status of each task displays below the task name.

Opwise assigns a default color for each status, which you can change (see [Changing a Task Status Color](#), below).

The following table identifies the default color for each status, along with the status value you will need if you change the color.

Status	Status Value	Default Color
Defined	0	Gray
Waiting	10	Gold
Held	20	Darkorange
Exclusive Requested	22	Gold
Exclusive Wait	23	Tomato
Resource Requested	25	Gold
Resource Wait	30	Tomato
Execution Wait	33	Tomato
Undeliverable	35	Red

Queued	40	Coral
Submitted	43	Green
Action Required	60	Darkorange
Started	70	YellowGreen
Running	80	Green
Running Problems	81	Red
Cancel Pending	99	Magenta
In Doubt	110	OrangeRed
Start Failure	120	Red
Confirmation Required	125	Fuchsia
Cancelled	130	Magenta
Failed	140	Red
Skipped	180	RoyalBlue
Finished	190	CornflowerBlue
Success	200	Blue

Changing a Task Status Color

Perform the following steps to change a task status color.



Note

You may need administrative privileges to perform these steps.

Step 1

From the navigation pane, select **Automation Center Administration > Configuration > Chart Colors**.

Step 2 Use the **Go To** field to select the Element **status_code**. Each **status_code** entry identifies a **Display** color and a **Value** associated with that status. The **Task Status** table, above, identifies the status associated with each **Value**.

Name	Color	Color name	Display	Element	Value
ops_exec		Fuchsia		status_code	125
ops_exec		Gold		status_code	22
ops_exec		Tomato		status_code	23
ops_exec		OrangeRed		status_code	110
ops_exec		CornflowerBlue		status_code	190
ops_exec		Magenta		status_code	99
ops_exec		Red		status_code	81
ops_exec		Gold		status_code	25
ops_exec		Gray		status_code	0
ops_exec		YellowGreen		status_code	70
ops_exec		Green		status_code	80
ops_exec		Red		status_code	140
ops_exec		Magenta		status_code	130
ops_exec		Darkorange		status_code	20
ops_exec		Gold		status_code	10
ops_exec		Tomato		status_code	30
ops_exec		Red		status_code	120
ops_exec		Coral		status_code	40
ops_exec		Black		status_code	-1
ops_exec		Blue		status_code	200
ops_exec		RoyalBlue		status_code	180
ops_exec		Red		status_code	35
ops_exec		Darkorange		status_code	60
ops_exec		Tomato		status_code	33
ops_exec		Green		status_code	43

Step 3 Locate the entry for the status you want to change. For example, to change the color associated with **Running** status, locate the entry with an Element of **status_code** and a Value of **80** (from the table above).

Step 4 Open the record by clicking on the Name (*ops_exec*) in the leftmost column. The example below shows the entry for value **80**, which is associated with "Running" status.

Chart Colors		Update	Delete	↑ ↓
Name:	Activity [ops_exec]	Color name:	<input type="text" value="Green"/>	<input type="text"/>
Element:	Status	Color:	<input type="text"/>	
Value:	<input type="text" value="80"/>			
Update		Delete		

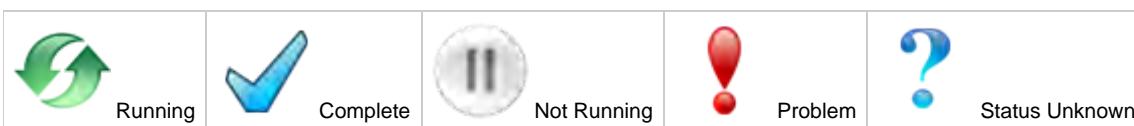
Step 5 Click on the magnifying glass to select a color or add a new color definition, or use the Color field to enter a hex representation for a color.

Step 6 Click **Update**.

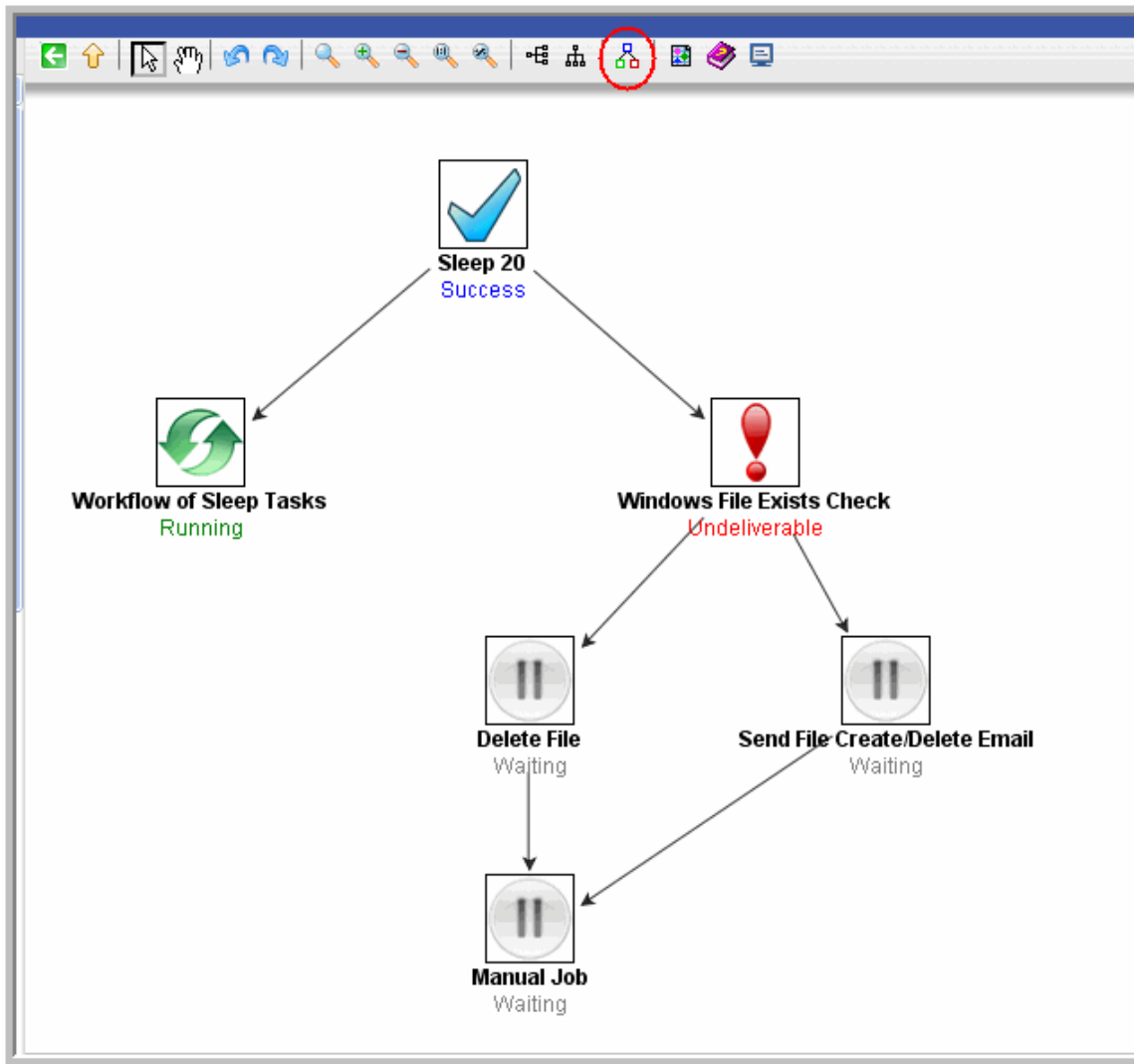
Displaying Task Status Icons


Each task status is categorized into one of five task status types: Running, Complete, Not Running, Problem, and Status Unknown.

Each task status type has a corresponding icon.



You can switch the Workflow Monitor display mode, which by default shows icons that represent *task types*, so that the icons represent task status types, as shown in the following example.




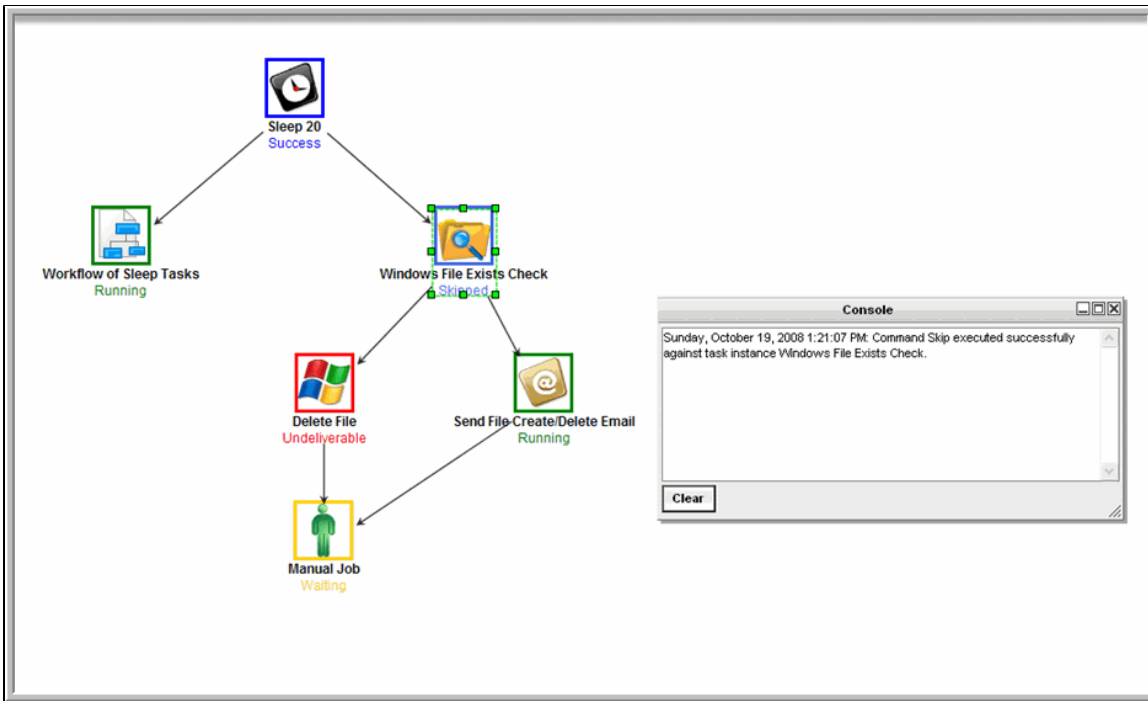
To switch display modes, click the **Toggle Vertex Style** icon .

Manually Intervening in a Workflow

You can take action on a task in a running workflow by right-clicking the task and selecting a command from the pop-up command list. When you select a command, the Console window automatically opens and displays processing information.

Displaying Processing Information

To display details about task instance processing while you monitor the workflow, click the Console  icon. The Console window then displays on the Workflow Monitor canvas.



Viewing Task Instances for a Specific Task

You can view a list of task instances for a specific task from its task definition screen.

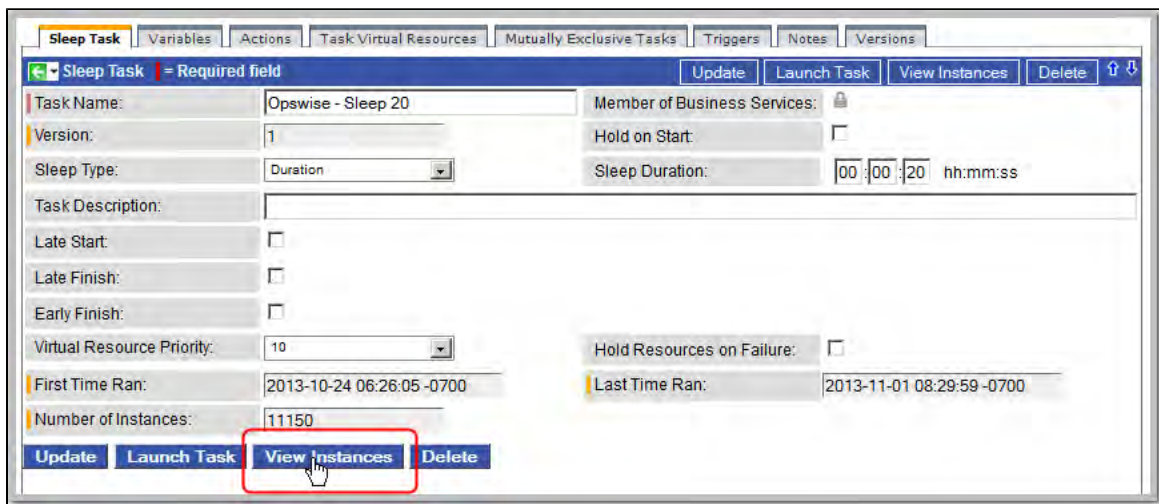
The list will display all task instances for which there has been a status change or a modification to the task instance record within the last 30 days.

From the displayed list of task instances, you can search for and display a specific instance, just as you can on the [Task Instances](#) screen.

Step 1 Select the task from the Tasks screen (**Automation Center > Tasks > All Tasks**) or the Tasks List screen for that task type. The task definition screen for that task displays.

Step 2 Display a list of task instances for that task either of two ways:

1. To navigate away from the Task Definition screen and display a Task Instances screen specific to that task, click the **View Instances** button.



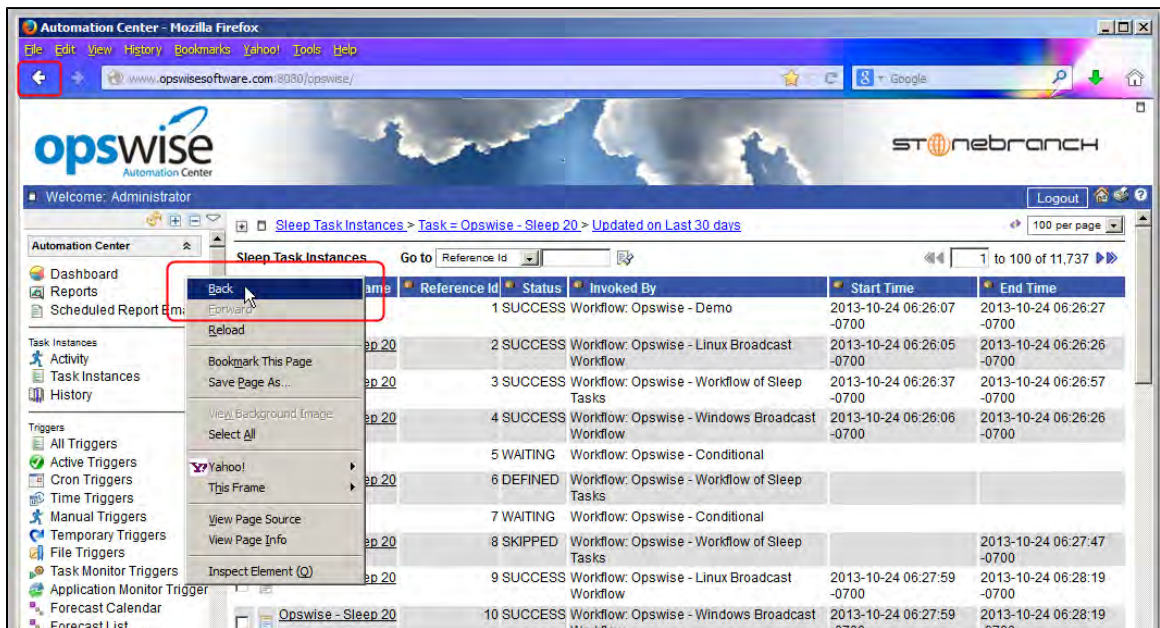
The Task Instances screen for that task displays.

The screenshot shows the 'Sleep Task Instances' screen. The breadcrumb navigation is 'Sleep Task Instances > Task = Opwise - Sleep 20 > Updated on Last 30 days'. There is a search bar and a 'Go to' field with a dropdown for 'Reference Id'. The table shows 10 instances, with columns for Instance Name, Reference Id, Status, Invoked By, Start Time, and End Time. The 'View Instances' button is highlighted with a red box and a mouse cursor.

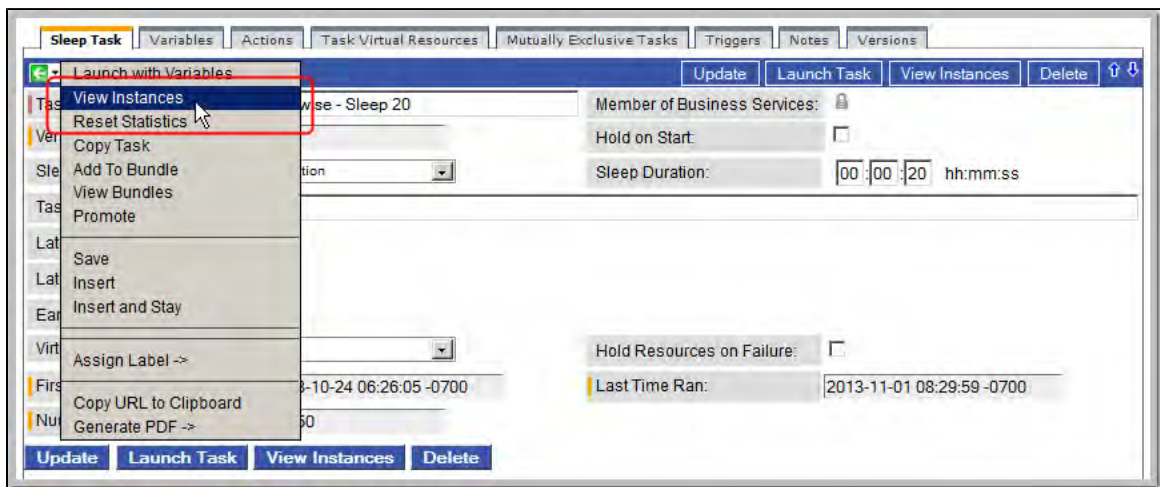
Instance Name	Reference Id	Status	Invoked By	Start Time	End Time
Sleep 20	1	SUCCESS	Workflow: Opwise - Demo	2013-10-24 06:26:07 -0700	2013-10-24 06:26:27 -0700
Opwise - Sleep 20	2	SUCCESS	Workflow: Opwise - Linux Broadcast Workflow	2013-10-24 06:26:05 -0700	2013-10-24 06:26:26 -0700
Opwise - Sleep 20	3	SUCCESS	Workflow: Opwise - Workflow of Sleep Tasks	2013-10-24 06:26:37 -0700	2013-10-24 06:26:57 -0700
Opwise - Sleep 20	4	SUCCESS	Workflow: Opwise - Windows Broadcast Workflow	2013-10-24 06:26:06 -0700	2013-10-24 06:26:26 -0700
Sleep 20	5	WAITING	Workflow: Opwise - Conditional		
Opwise - Sleep 20	6	DEFINED	Workflow: Opwise - Workflow of Sleep Tasks		
Sleep 20	7	WAITING	Workflow: Opwise - Conditional		
Opwise - Sleep 20	8	SKIPPED	Workflow: Opwise - Workflow of Sleep Tasks		2013-10-24 06:27:47 -0700
Opwise - Sleep 20	9	SUCCESS	Workflow: Opwise - Linux Broadcast Workflow	2013-10-24 06:27:59 -0700	2013-10-24 06:28:19 -0700
Opwise - Sleep 20	10	SUCCESS	Workflow: Opwise - Windows Broadcast Workflow	2013-10-24 06:27:59 -0700	2013-10-24 06:28:19 -0700

To return to the Task Definition screen, either:

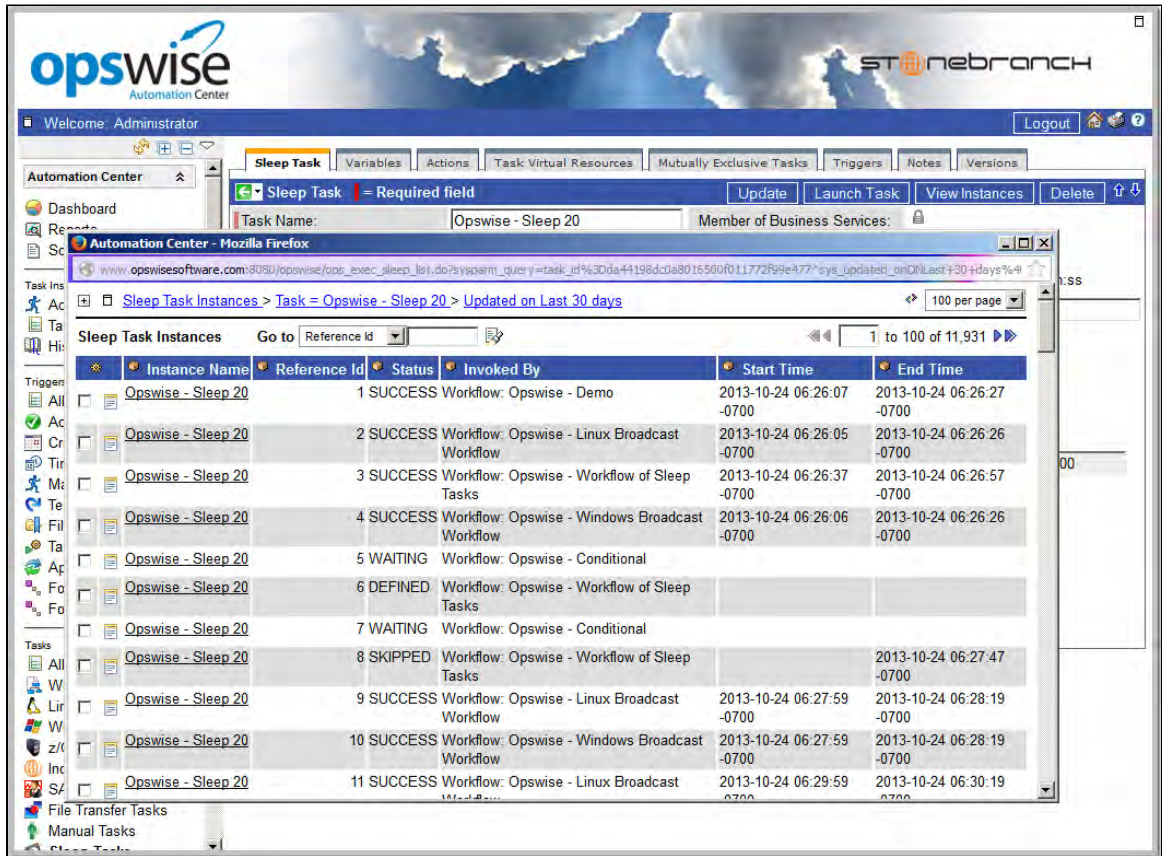
- Click your browser Back arrow.
- Right-click outside of the listed task instances and click **Back** on the browser pop-up menu.



2. To stay on the Task Definition screen and display a pop-up version of the Task Instances screen specific to that task, right-click the blue title bar - or hover your cursor over the down arrow on the blue title bar - and click **View Instances** on the drop-down Action menu.



The Task Instances pop-up screen for that task displays.



To close the Task Instances pop-up screen, either:

- Click the **Close** icon (**x**) at the top right corner of the pop-up screen.
- Click the Task Definition screen (still displayed underneath the pop-up screen).

The screenshot displays the Opwise Automation Center interface. At the top, the Opwise logo and 'Automation Center' text are visible on the left, and the Stonebranch logo is on the right. Below the header, there's a navigation bar with tabs for 'Sleep Task', 'Variables', 'Actions', 'Task Virtual Resources', 'Mutually Exclusive Tasks', 'Triggers', 'Notes', and 'Versions'. The 'Sleep Task' tab is active, showing a task named 'Opwise - Sleep 20'. Below this, there's a table of 'Sleep Task Instances' with the following data:

Instance Name	Reference Id	Status	Invoked By	Start Time	End Time
Opwise - Sleep 20	1	SUCCESS	Workflow: Opwise - Demo	2013-10-24 06:26:07-0700	2013-10-24 06:26:27-0700
Opwise - Sleep 20	2	SUCCESS	Workflow: Opwise - Linux Broadcast Workflow	2013-10-24 06:26:05-0700	2013-10-24 06:26:26-0700
Opwise - Sleep 20	3	SUCCESS	Workflow: Opwise - Workflow of Sleep Tasks	2013-10-24 06:26:37-0700	2013-10-24 06:26:57-0700
Opwise - Sleep 20	4	SUCCESS	Workflow: Opwise - Windows Broadcast Workflow	2013-10-24 06:26:06-0700	2013-10-24 06:26:26-0700
Opwise - Sleep 20	5	WAITING	Workflow: Opwise - Conditional		
Opwise - Sleep 20	6	DEFINED	Workflow: Opwise - Workflow of Sleep Tasks		
Opwise - Sleep 20	7	WAITING	Workflow: Opwise - Conditional		
Opwise - Sleep 20	8	SKIPPED	Workflow: Opwise - Workflow of Sleep Tasks		2013-10-24 06:27:47-0700
Opwise - Sleep 20	9	SUCCESS	Workflow: Opwise - Linux Broadcast Workflow	2013-10-24 06:27:59-0700	2013-10-24 06:28:19-0700
Opwise - Sleep 20	10	SUCCESS	Workflow: Opwise - Windows Broadcast Workflow	2013-10-24 06:27:59-0700	2013-10-24 06:28:19-0700
Opwise - Sleep 20	11	SUCCESS	Workflow: Opwise - Linux Broadcast	2013-10-24 06:29:59-0700	2013-10-24 06:30:19-0700

Displaying Task Instance Status

- [Displaying Task Instance Status](#)
- [Task Instance Status Types](#)

Displaying Task Instance Status

You can display the status of one or more task instances from the Activity screen, Task Instances screen, History screen, or Command Line Interface (CLI).

Activity screen	To display the status of one or more task instances on the Activity screen: <ol style="list-style-type: none"> 1. From the navigation pane, select Automation Center > Task Instances > Activity. The Activity screen displays. 2. From the drop-down menu at the top of the Activity screen, select the type of task instances to display.
Task Instances screen	To display the status of one or more task instances on the Task Instances screen: <ul style="list-style-type: none"> • From the navigation pane, select Automation Center > Task Instances > Task Instances. The Task Instances screen displays.
History screen	To display the status of one or more task instances on the History screen: <ul style="list-style-type: none"> • From the navigation pane, select Automation Center > Task Instances > History. The History screen displays.
Command Line Interface (CLI)	To display the status of one or more task instances from the Command Line Interface: <ul style="list-style-type: none"> • Use the <code>ops-task-status</code> command.

Task Instance Status Types

The following table describes all possible task instance statuses for all task types.

(See [Commands Supported for Task Instance Statuses](#) for a list of commands that you can issue against a task instance in each status.)

Status	Code	Task Type	Description
Action Required	60	Manual	When a manual task launches, it goes into Action Required status, meaning a user must perform some manual activity. For details, see Manual task .
Cancel Pending	99	Agent-based*	A process running on the agent needs to be terminated. When the Cancel command is issued, the task instance will go into a Cancel Pending status until the Agent reports back that the process has been cancelled. At that point, the task instance will transition into the Cancelled status.
Cancelled	130	All	The task was cancelled by a user.
Confirmation Required	125	z/OS	If you make JCL changes and restart a z/OS task, Opwise will put the task into Confirmation Required status and prompt you for a confirmation. For detailed processing steps, see Rerunning a z/OS Task .
Defined	0	All	The new task instance has been created (the task has been launched).
Exclusive Requested	22	All	All tasks with a mutually exclusive task defined go immediately to a status of Exclusive Requested. If the task is available to run exclusively, the task then moves to the next appropriate processing status.

Exclusive Wait	23	All	The task is mutually exclusive with one or more other tasks, and it is waiting for those tasks to finish before it will run.
Execution Wait	33	Agent-based*	The task must wait to be completed; either the Agent/Agent Cluster running the task has reached its Task Execution Limit, or the ability of the Agent/Agent Cluster to run tasks has been suspended .
Failed	140	All	The task ran to a failure status.
Finished	190	All	The task was forced by the user to finish. The user may do this in cases where the task had a Cancelled or Failed status, and the user needed to release other task instances depending on the successful completion of this task instance in a workflow. For more information, see Force Finishing a Task .
Held	20	All	The task has been put on hold by a user.
In Doubt	110	Agent-based*	The agent is "in doubt" about the current status of the task instance. This may occur if an agent or agent connection goes down. In this case, the agent restarts and reviews its data about tasks in progress. If the agent finds a task still running, it resumes normal monitoring. If the agent cannot find the task, this usually indicates that the task completed, but the agent considers the task status to be "in doubt."
Queued	40	Agent-based*	The task has been queued on a resource.
Resource Requested	25	All	All tasks with a virtual resource defined go immediately to a status of Resource Requested. If the resource is available, the task then moves to the next appropriate processing status.
Resource Wait	30	All	All tasks with a virtual resource defined go immediately to a status of Resource Requested. If the resource is not available, the task goes to a status of Resource Wait. When the resource becomes available, the task moves to the next appropriate processing status
Running	80	All	The task is running. For agent-based tasks, the agent has started running the program.
Running Problems	81	Workflow	One or more tasks within the workflow has one of the following statuses: <ul style="list-style-type: none"> • Confirmation Required • Undeliverable • Running Problems (for sub-workflows) • In Doubt • Failure • Start Failure • Cancelled
Skipped	180	All	The task was skipped by a user.
Start Failure	120	All	The task was unable to start.
Started	70	Agent-based*, Manual	The task has started. For agent-based tasks, this means the agent has received the task.
Step Restarted	45	z/OS	The task has been restarted from a specific z/OS jobstep .
Submitted	43	z/OS	The task has been submitted to the z/OS Job Entry subsystem and scheduled by the z/OS Job Scheduler.
Success	200	All	The task has completed successfully. Workflows will transition to Success status when all of its tasks have transitioned to Success, Finished, or Skipped status.
Undeliverable	35	Agent-based*	The agent is unavailable.
Waiting	10	All	The task has been loaded by a workflow and is waiting to run.

* Agent-based task types are Linux/Unix, Windows, z/OS, Indesca, SAP, File Transfer, File Monitor, FTP File Monitor, and System Monitor.

Retrieving Output

- [Overview](#)
- [Retrieve the Output](#)
- [Retrieve Output Field Descriptions](#)

Overview

For any running or completed task where output (for example, standard out and/or standard error) has been generated and you did not specify that output be [automatically retrieved](#), you can instruct Opswise to retrieve the output.

Retrieve the Output

Step 1 From the navigation pane, select **Automation Center > Task Instances > Activity**. The Activity screen displays.

Step 2 Click the task whose output you want to retrieve. The task instance screen for that task displays, as shown in the following example of a Linux/Unix task instance.

Linux/Unix Task Instance
Output

Linux/Unix Task Instance
Update
View Parent
Show Details
Retrieve Output
Re-run
Delete

Instance Name:	Opwise - Linux Success	Invoked By:	Workflow: Opwise - Linux 2x Sub Workflc
Task:	Opwise - Linux Success	Execution User:	ops.admin
Reference Id:	78503	Credentials:	<input type="text"/>
Member of Business Services:		Credentials Variable:	<input type="checkbox"/>
Agent:	server.opswissoftware.com - AGN	Run as sudo:	<input type="checkbox"/>
Agent Variable:	<input type="checkbox"/>	Agent Cluster:	Opwise - Demo Linux/Unix Cluste
Hold Reason:	<input type="text"/>		
Task Description:	<input type="text"/>		
Status:	SUCCESS	Exit Code:	0
Status Description:	<input type="text"/>		
Queued Time:	2013-09-18 11:55:03 -0700	Process ID:	11695
Start Time:	2013-09-18 11:55:04 -0700	CPU Time:	199
End Time:	2013-09-18 11:55:04 -0700	Duration:	0 Seconds
Command or Script:	Command		
Command:	<div style="border: 1px solid gray; padding: 5px; min-height: 100px;"> exit </div>		
Parameters:	<div style="border: 1px solid gray; padding: 5px; min-height: 100px;"> 0 </div>		
Runtime Directory:	<input type="text"/>		
Exit Code Processing:	Success Exitcode Range		
Exit Codes:	<input type="text" value="0"/>		

Add environment variables by inputting the name and value and clicking "Add":

Name: Value: Add

Automatic Output Retrieval:	Standard Output	Retry Indefinitely:	<input type="checkbox"/>
Start Line:	<input type="text"/>	Maximum Retries:	0
Number of Lines:	<input type="text"/>	Retry Interval (Seconds):	60
Scan Text:	<input type="text"/>	Current Retry Count:	0
Task Priority:	MEDIUM	User Estimated End Time:	2013-09-19 08:55:04 -0700
Shortest Estimated End Time:	2013-09-18 11:55:04 -0700	Average Estimated End Time:	2013-09-18 11:55:05 -0700
Longest Estimated End Time:	2013-09-18 11:55:11 -0700	Virtual Resource Priority:	10
Hold Resources on Failure:	<input type="checkbox"/>		

Update
View Parent
Show Details
Retrieve Output
Re-run
Delete

Step 3 Click the **Retrieve Output** button. If output is available, the Retrieve Output window displays.

Step 4 Using the field descriptions, below, as a guide, make your selection. Positioning can be accomplished by using **Start Line** (enter **-1** to retrieve the last **Number of lines**) or **Scan text** to see the **Number of lines** both before and after the first match of text.

Step 5 Click the **Submit** button. Opswise retrieves the output you specified and writes the record to the Output tab, as shown in the following example. If the output is of the same output type, and for the same attempt, Opswise purges the previously retrieved output.

Type	Updated	Output	Attempt
STDOUT	2012-10-29 06:52:51 -0700	[empty]	1
STDERR	2012-10-29 06:52:51 -0700	[empty]	1

Step 6 To display the output, open the Output tab and click on the record. The output record displays, as shown in the following example.

Retrieve Output Field Descriptions

Column Name	Description
Standard Output	Retrieve standard output returned by the program.
Standard Error	Retrieve standard error information returned by the program.

Standard Output and Standard Error	Retrieve both standard output and standard error information.
Start Line	Retrieve data beginning at the line indicated. If a Start Line value is not specified on the Retrieve Output window, default is 1.
Number of lines	Limit the retrieved data to the Number of Lines value on the Retrieve Output window. If a Number of Lines value is not specified, default is the value of the Retrieve Output Default Maximum Lines Opswise system property.
Scan text	Scan the data for the text specified and retrieve only that.

Variables and Functions



Variables

[Overview](#)

[User-Defined Variables](#)

[Built-In Variables](#)



Using Variables

[Setting Variables under Special Circumstances](#)

[Launching With Variables](#)

[Triggering with Variables](#)

[Setting Variables within a Workflow](#)

[Listing and Setting Variables from the Command Line](#)



Functions

[Overview](#)

[Date Functions](#)

[Mathematical Functions](#)

[System Functions](#)

[String Functions](#)

[SQL/Stored Procedure Functions](#)



Task and Trigger Built-In Variables

[Task Instances Variable](#)

[Trigger Variables](#)

[File Monitor Variables](#)

[FTP File Monitor Variables](#)

[Task Monitor Variables](#)

[Script Variables](#)

[SAP Task Variables](#)

[z/OS Task Variables](#)

[Application Monitor Variables](#)

[SQL and Stored Procedure Variables](#)

[System Monitor Variables](#)



Other Built-In Variables

[Agent Notification Variables](#)

[Cluster Node Notification Variables](#)

[Connector Notification Variables](#)



The information on these pages also is located in the [Opwise Automation Center 5.1.1 User Guide.pdf](#).

Variables and Functions Overview

Opswise supports the following types of variables, all of which can be used in free text fields within tasks:

- [User-Defined Variables](#) – These variables are created by the user.
- [Built-In Variables](#) – These variables, maintained by Opswise, allow you to access information about task instances and other related data, such as task name, task status, and trigger name.
- [Functions](#) – These variables calculate some value, such as current date and time, or perform some function, such as `_replaceAll`.

Opswise also supports several features that allow you to set variables under special circumstances:

- You can [manually launch tasks and temporarily set user-defined variables](#).
- Opswise provides two methods that allow you to manually launch all of the tasks associated with a trigger while supplying variable values used by the task(s) (see [Triggering with Variables](#)).
- [Setting Variables within a Workflow](#)
- Using the `ops-variable-set` CLI command to set variables.

User-Defined Variables

- Overview
- Variable Naming Conventions
- Resolving User-Defined Variables
 - For Tasks Launched by a Trigger
 - For Tasks Launched by a Workflow
 - For Tasks Launched Manually
- Format for Using Variables
- Defining a New Variable
- Variable Field Descriptions

Overview

You can define Opswise variables in four locations:

1. Define **Trigger variables** by clicking the **Variables** tab in a trigger. Trigger variables are stored in the table **ops_local_variable**.
2. Define **Task variables** by clicking the **Variables** tab in a task. Task variables are stored in the table **ops_local_variable**.
3. Define **Workflow variables** by clicking the **Variables** tab in a workflow. Workflow variables are stored in the table **ops_local_variable**.
4. Define **Global variables** by selecting **Automation Center > Variables** from the navigation pane. Global variables are stored in the table **ops_variable**.

Variable Naming Conventions

- Variable names must begin with a letter.
- Allowable characters are alphanumeric (upper or lower case), and underscore (_).
- White spaces are not permitted
- Variable names are not case-sensitive.



Warning

Do not define Opswise variables with the prefix **ops_**. That prefix is reserved for built-in variables.

Resolving User-Defined Variables

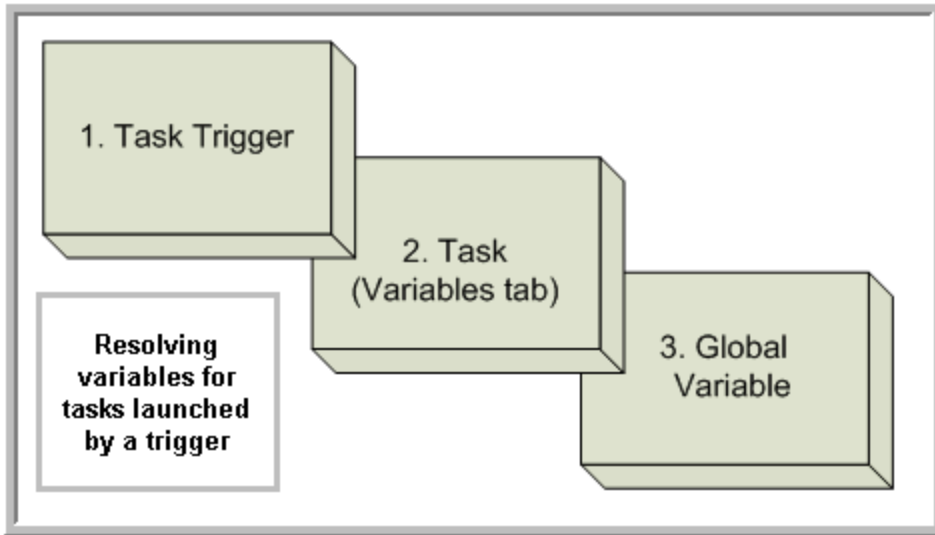
When Opswise creates a task instance from a task definition, it also resolves all variables specified in its free text fields. Because you can define variables at four different levels (trigger, task, workflow, and global), Opswise follows a prescribed formula to determine which variable takes precedence if duplicate variables have been defined. The general order of precedence, each of which may or may not exist in any given situation, is as follows:

1. Task trigger (highest precedence)
2. Task
3. Workflow trigger
4. Workflow
5. Global (lowest precedence)

The following scenarios provide more detailed information about how Opswise variables are resolved.

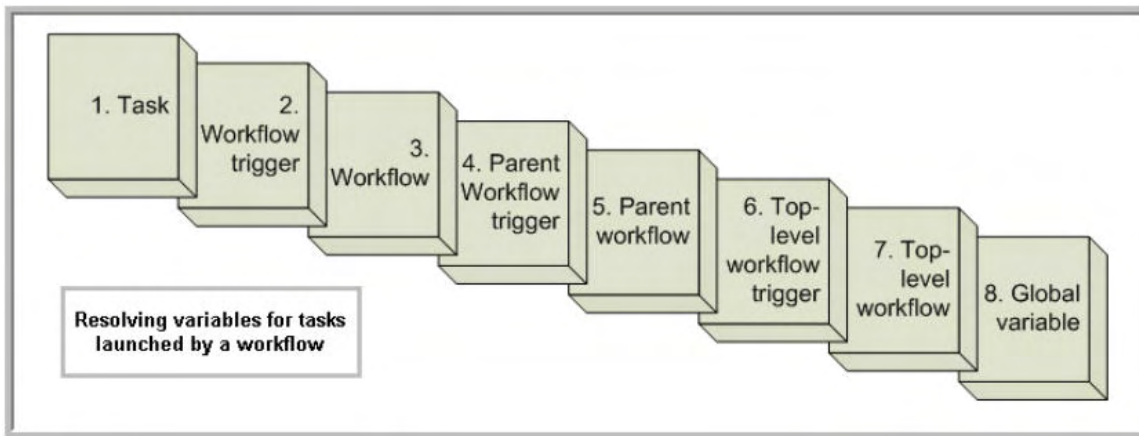
For Tasks Launched by a Trigger

1. If the trigger defines the variable in the variables tab, that value is used to resolve the variable.
2. If the trigger does not define the variable, the value from the variable tab in the task definition is used.
3. If neither the trigger nor the task define the variable, the variable definition in the global variables table is used.
4. If the global variables table does not define the variable, the variable remains unresolved.



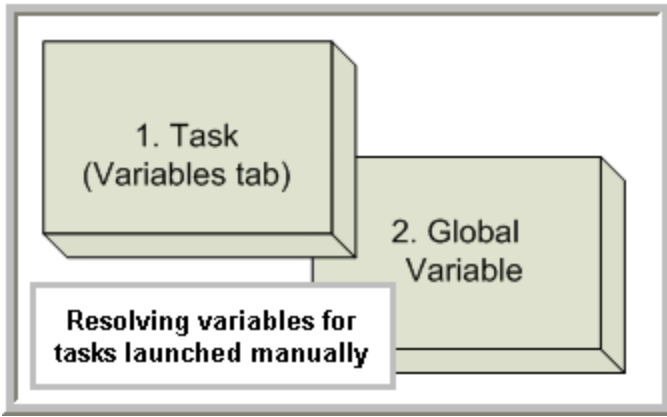
For Tasks Launched by a Workflow

1. If the task defines the variable in the variables tab, that value is used to resolve the variable.
2. If the task does not define the variable, and the workflow was launched by a trigger, the value defined in the trigger is used.
3. If the workflow's trigger does not define the variable or the workflow was not launched by a trigger, the value defined in the workflow is used.
4. If the workflow does not define the variable, and there is a parent workflow, the value defined in the parent workflow's trigger is used.
5. If the parent workflow's trigger does not define the variable or if there is no trigger, the value defined in the parent workflow is used.
6. If the parent workflow does not define the variable, Opswise checks up a level for the trigger on the next parent workflow.
7. If that trigger does not define the variable, it checks for variables associated with the workflow. (This continues until the top level workflow is reached.)
8. If the top-level workflow does not define the variable, the variable definition in the global variables table is used.
9. If the global variables table does not define the variable, the variable remains unresolved.



For Tasks Launched Manually

1. If the task defines the variable in the variables tab, that value is used to resolve the variable.
2. If the task does not define the variable, the variable definition in the global variables table is used.
3. If the global variables table does not define the variable, the variable remains unresolved.



Format for Using Variables

When you enter a variable into a text field, precede the variable with the dollar sign (\$) and enclose the variable in curly braces ({ }). You can enter a series of variables or nested variables. Examples are:

```

${variable_name}
${v1}${v2}
${${inner_variable}}
    
```

Defining a New Variable

Step 1 For global variables: From the Navigation Pane, select **Automation Center > Variables**. For trigger, task, and workflow variables: Click the **Variables** tab from the record. The Variables List screen displays.

Name	Value	Description	Updated	Updated by
demo_ops_download_dir	/home/opswise/download		2008-11-26 16:32:16 -0800	glide.maint
demo_ops_global	Global		2008-12-10 16:46:04 -0800	ops.admin
demo_ops_linux_rc	0		2008-11-26 16:32:30 -0800	glide.maint
demo_ops_os	linux		2009-03-02 09:33:22 -0800	ops.admin
demo_ops_rc	0		2008-11-26 16:32:43 -0800	glide.maint
demo_ops_remove	\${demo_ops_toolsdir}\rm.exe		2008-11-26 16:35:31 -0800	glide.maint
demo_ops_rundir	\${demo_ops_toolsdir}		2008-11-26 16:35:20 -0800	glide.maint
demo_ops_sleep_time	5		2008-11-26 16:33:14 -0800	glide.maint
demo_ops_snooze	\${demo_ops_toolsdir}\snooze_bg.bat		2008-11-26 16:35:07 -0800	glide.maint
demo_ops_toolsdir	\${demo_ops_workspace}\com.jme.opswise.plugins\tools\windows		2008-11-26 16:34:59 -0800	glide.maint
demo_ops_touch	\${demo_ops_toolsdir}\touch.exe		2008-11-26 16:34:52 -0800	glide.maint
demo_ops_workspace	C:\workspace\eclipse\opswise		2008-11-26 16:34:04 -0800	glide.maint

Step 2 Click **New**. The Variable Definition screen displays:

The screenshot shows a web-based form titled 'Variables' with a 'Versions' tab. The form contains the following fields and controls:

- Name:** A text input field containing 'cbr_cmd'.
- Value:** A text input field containing 'dir'.
- Description:** A larger text input field, currently empty.
- Version:** A text input field containing '1'.
- Buttons:** 'Update' and 'Delete' buttons are located at the top right and bottom left of the form.

Step 3 Using the field descriptions provided below as a guide, complete the fields as needed.

Step 4 Click the **Submit** button to save the record and return to the menu, or right-click and select **Save** to save the record and remain on the current display.

Step 5 If appropriate, repeat these steps for any additional variables you want to add.

Variable Field Descriptions

The table below describes the fields and buttons on the Variables screen.

Field Name	Description
Name	Required. Name of variable. Up to 40 alphanumeric. The name must begin with an alphabetic character and can consist of: alphas (a-z, A-Z), numerics 0-9, _ (underscore). White spaces are not permitted; names are not case-sensitive. Important Note – Do not define Opwise variables with the prefix "ops_". That prefix is reserved for built-in variables.
Value	Optional. The value of the variable.
Description	Optional. Description of this variable.
Version	System-supplied. The version number of the current record, which is incremented by Opwise every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Delete button	Deletes the current record.
Versions tab	Stores copies of all previous versions of the current record. See Record Versioning .

Built-In Variables

- [Overview](#)
- [Task and Task Instance Variables](#)
- [Trigger Variables](#)
- [File Monitor Variables](#)
- [FTP File Monitor Variables](#)
- [Task Monitor Variables](#)
- [Script Variables](#)
- [SAP Task Variables](#)
- [z/OS Task Variables](#)
- [Application Monitor Variables](#)
- [SQL and Stored Procedure Variables](#)
- [Agent Variables](#)
- [Cluster Node Variables](#)
- [Connector Variables](#)
- [System Monitor Variables](#)

Overview

Built-in variables are maintained by Opwise and provide information about task instances, agents, connectors (Message Hubs and Transporters), and Controllers (Cluster Nodes). They can be used in free text fields on trigger variable values, tasks, task actions, agent notifications, connector notifications, and cluster node notifications.

To propagate the value of a built-in variable in a task instance to a parent workflow variable scope, you can use the [Set Variable](#) action. This allows the values of built-in variables to be shared among task instances within the same workflow.

Supported built-in variables and their descriptions are provided below. All built-in variables are prefixed with `ops_`.

Task and Task Instance Variables

The following built-in variables are associated with task instances:

Variable	Description
<code>\${ops_attempt}</code>	Resolves to the current task instance attempt count. Each Re-run operation increments the attempt. Initial attempt is 1.
<code>\${ops_cmd}</code>	For tasks that launch a command on a Windows, Linux/Unix, or z/OS machine, resolves to the task command.
<code>\${ops_cmd_parms}</code>	For tasks that launch a command on a Windows, Linux/Unix, or z/OS machine, resolves to the task command parameters.
<code>\${ops_duration}</code>	Resolves to the task running time in milliseconds (for example: 130000).
<code>\${ops_duration_text}</code>	Resolves to the task running time in a more readable representation of the duration time (for example: 2 Minutes 10 Seconds).
<code>\${ops_end_time}</code>	Resolves to the task ending time.

<code>\${ops_execution_user}</code>	Resolves to the ID of the user who launched the task or to the ID of the user who enabled the trigger that launched the task.
<code>\${ops_exit_code}</code>	Resolves to the task instance exit code, if any.
<code>\${ops_launch_time}</code>	Resolves to the task launch time. For workflows, all descendants will have the same launch time as the top-level workflow.
<code>\${ops_retry_count}</code>	Resolves to the current retry count.
<code>\${ops_retry_interval}</code>	Resolves to the retry interval (seconds).
<code>\${ops_retry_maximum}</code>	Resolves to the maximum retry count.
<code>\${ops_start_time}</code>	Resolves to the task starting time.
<code>\${ops_status}</code>	Resolves to the current task instance status.
<code>\${ops_task_id}</code>	Resolves to the sys_id of the task instance.
<code>\${ops_task_name}</code>	Resolves to the task name.
<code>\${ops_task_ref_count}</code>	Each time an instance is created from a specific task definition, it gets a unique task reference count for that specific task. For example, if you launch a task twice, the first instance will have task reference count 1 and the second will have task reference count 2.
<code>\${ops_task_type}</code>	Resolves to the task type.
<code>\${ops_workflow_id}</code>	Resolves to the sys_id of the parent workflow task instance.
<code>\${ops_workflow_name}</code>	Resolves to the name of the parent workflow.

Trigger Variables

When a task is launched by a trigger of any type, the values of the following built-in variables, if they are specified in the task, are passed into the task instance.

Variable	Description
<code>\${ops_trigger_name}</code>	Resolves to the name of the trigger that launched the task instance.
<code>\${ops_trigger_time}</code>	Resolves to the scheduled time of the trigger or, if the trigger is not scheduled, the actual trigger time.

File Monitor Variables

When one or more tasks are launched by a [File Monitor trigger](#) after the conditions in its associated File Monitor task are met, the built-in variables described below are passed into the tasks being launched by the trigger.

For example, the File Monitor trigger may specify the launch of a Windows task each time the associated File Monitor task detects the creation of a specific file. The Windows task might use one of these built-in variables as a command argument. Or, if the File Monitor task is not associated with a trigger but is running within a workflow, on completion you can propagate one or more of these built-in variable values to the parent workflow level using the [Set Variable](#) action. This allows you to pass information from the File Monitor task to a successor task within the same workflow hierarchy.

Variable	Description
<code>\${ops_trigger_file_name}</code>	Resolves to the name of the file that fired the trigger.
<code>\${ops_trigger_file_name_nopath}</code>	Resolves to the file name without any path information.
<code>\${ops_trigger_file_name_simple}</code>	Resolves to the base file name.
<code>\${ops_trigger_file_name_extension}</code>	Resolves to the file extension of a file.
<code>\${ops_trigger_file_path}</code>	Resolves to the directory where the new file was created, but not the file itself. If the existence or non-existence of the final directory separator is a requirement, we recommend the use of <code>\${ops_trigger_file_fullpath}</code> and <code>\${ops_trigger_file_fullpath_no_separator}</code> , respectively.
<code>\${ops_trigger_file_fullpath}</code>	Resolves to the directory where the new file was created, but not the file itself; includes the final directory separator.
<code>\${ops_trigger_file_fullpath_no_separator}</code>	Resolves to the directory where the new file was created, but not the file itself; does not include the final directory separator.

<code>\${ops_trigger_file_separator}</code>	Resolves to the separator appropriate to the platform where the agent is running. For Windows, resolves to a backslash (\); for Linux/Unix, resolves to forward slash (/). This variable may be useful if you want to piece together a pathname using a combination of text and variables.
	For example: <code>\${ops_trigger_file_fullpath}sub_folder_name \${ops_trigger_file_separator}filename.txt</code>
<code>\${ops_trigger_file_size}</code>	Resolves to the file size of the file that fired the trigger.
<code>\${ops_trigger_file_date}</code>	Resolves to the file date of the file that fired the trigger.
<code>\${ops_trigger_file_scan}</code>	Resolves to the result of the file scan: FOUND or NOT_FOUND.
<code>\${ops_trigger_file_owner}</code>	Resolves to the file owner of the file that fired the trigger.
<code>\${ops_trigger_file_group}</code>	Resolves to the file group of the file that fired the trigger.

FTP File Monitor Variables

The following built-in variables are available for FTP File Monitor task instances and provide information about the file or file(s) that matched the monitor's criteria.

You can use these variables in an FTP File Monitor action or in a successor task instance by propagating one or more of these built-in variable values to a parent workflow using the [Set Variable](#) action.

Variable	Description
<code>\${ops_trigger_file_name}</code>	Resolves to the remote file name.
<code>\${ops_trigger_file_name_nopath}</code>	Resolves to the remote file name without any path information.
<code>\${ops_trigger_file_name_simple}</code>	Resolves to the base file name.
<code>\${ops_trigger_file_name_extension}</code>	Resolves to the file extension of the file.

<code>\${ops_trigger_file_path}</code>	Resolves to the directory where the remote file is located, but not the file itself. <code>\${ops_trigger_file_path}</code> is an alias of <code>\${ops_trigger_file_fullpath_no_separator}</code>
<code>\${ops_trigger_file_fullpath}</code>	Resolves to the directory where the remote file is located, but not the file itself; includes the final directory separator.
<code>\${ops_trigger_file_fullpath_no_separator}</code>	Resolves to the directory where the remote file is located, but not the file itself; does not include the final directory separator.
<code>\${ops_trigger_files}</code>	Resolves to a comma-separated list of files that matched the wildcard, if one was specified in the Remote Filename field in the FTP File Monitor task. For example: <pre>ops_trigger_files = COMPANY-2011-11-22.xls, COMPANY-2011-11-23.xls,COMPANY-2011-11-24.xls</pre>
<code>\${ops_trigger_wildcard}</code>	Resolves to the contents of the Remote Filename field from the task definition. For example: <pre>ops_trigger_wildcard = /home/prod/stonebranch/COMPANY*.xls</pre>
<code>\${ops_trigger_wildcard_path}</code>	Resolves to the path only, with the final slash but without the file name, from the Remote Filename field in the task definition. For example: <pre>ops_trigger_wildcard_path = /home/prod/stonebranch/</pre>
<code>\${ops_trigger_wildcard_path_no_separator}</code>	Resolves to the path only, without the final slash and without the file name, from the Remote Filename field in the task definition. For example: <pre>ops_trigger_wildcard_path_no_separator = /home/prod/stonebranch</pre>

Task Monitor Variables

When the conditions of a Task Monitor task are met and its associated [Task Monitor trigger](#) launches one or more tasks, the following built-in variables are passed into the task instances being launched by the trigger.

For example, the Task Monitor trigger may specify an Email task that will launch each time the conditions in the associated Task Monitor task are met. You might want to specify one or more of these variables in the body of the email.

If the Task Monitor task is not associated with a trigger but is running within a workflow, on completion you can propagate one or more of these built-in variable values to the parent workflow level by using the [Set Variable](#) action. This allows you to pass information from the Task Monitor task to a successor task within the same workflow hierarchy.

Variable	Description
<code>\${ops_trigger_task_name}</code>	Resolves to the name of the task instance that fired the trigger.
<code>\${ops_trigger_task_id}</code>	Resolves to the sys_id of the task instance that fired the trigger.
<code>\${ops_trigger_task_status}</code>	Resolves to the status of the task instance that fired the trigger.
<code>\${ops_trigger_task_type}</code>	Resolves to the type of the task instance that fired the trigger.
<code>\${ops_trigger_workflow_name}</code>	Resolves to the name of the Workflow instance that fired the trigger. This variable is available only for a Task Monitor task that has a Workflow Condition specified. If a Workflow Condition is specified, <code>\${ops_trigger_workflow_name}</code> will resolve to the name of the Workflow instance that the Workflow Condition matched.

Script Variables

For Windows, Linux/Unix, and SAP tasks where a Script or SAP Definition from the Script Library is specified, the following built-in variables are passed into the task instance. You can use these in a Windows, Linux/Unix, or SAP task action or propagate one or more of these built-in variable values to the parent workflow using the [Set Variable](#) action.

Variable	Description
<code>\${ops_script_id}</code>	Resolves to the Opwise system ID of the script.
<code>\${ops_script_name}</code>	Resolves to the Opwise name of the script.

SAP Task Variables

For an SAP task instance, where applicable, the following built-in variables resolve to the SAP jobname and SAP jobid of the job running in the SAP system. If you need to use the SAP jobname and/or the SAP jobid from one SAP task instance in a successor SAP task instance, you can use the [Set Variable](#) action to propagate these built-in variable values to the parent workflow.

Variable	Description
<code>\${ops_sap_jobid}</code>	Resolves to the SAP job ID.
<code>\${ops_sap_jobname}</code>	Resolves to the SAP job name.

<code>\${ops_sap_chainid}</code>	Resolves to the SAP Process Chain ID.
<code>\${ops_sap_logid}</code>	Resolves to the SAP Process Chain Log ID.
<code>\${ops_sap_requestid}</code>	Resolves to the SAP InfoPackage Request ID.

z/OS Task Variables

The following built-in variables are available for z/OS task instances:

<code>\${ops_jcl_location}</code>	Resolves to the file and member name containing the JCL script.
<code>\${ops_job_id}</code>	Resolves to the job number assigned to the job by JES.

Application Monitor Variables

When a task is launched by an [Application Monitor trigger](#), the following built-in variables are passed into the task being launched by the trigger:

Variable	Description
<code>\${ops_trigger_appl_name}</code>	Resolves to the name of the Application being monitored by the trigger.
<code>\${ops_trigger_appl_status}</code>	Resolves to the status of the Application being monitored by the trigger.
<code>\${ops_trigger_appl_type}</code>	Resolves to the type of Application being monitored by the trigger, as defined by the Application Type field.
<code>\${ops_trigger_appl_id}</code>	Resolves to the sys_id of the application.

SQL and Stored Procedure Variables

The following built-in variables are used in [SQL](#) tasks and [Stored Procedure](#) tasks to collect SQLException data, if any:

Field Name	Description
------------	-------------

<code>\${ops_sql_error_msg}</code>	Resolves to any error message generated by the database.
<code>\${ops_sql_rows}</code>	Resolves to the number of rows processed.
<code>\${ops_sql_state}</code>	Resolves to a return code that indicates the outcome of the most recently executed SQL statement.

Agent Variables

The following agent variables can be used to pass information into an [agent notification](#).

Some of these variables, as noted, also can be used to pass agent information into an agent-based task (Windows, Linux/Unix, z/OS, and SAP).

Field Name	Description
<code>\${ops_agent_hostname}</code>	Resolves to the agent hostname. You can also use this variable in task notifications; see Creating Email Notifications and Creating SNMP Notifications .
<code>\${ops_agent_id}</code>	Resolves to the agent queue name. You can also use this variable in task notifications; see Creating Email Notifications and Creating SNMP Notifications .
<code>\${ops_agent_ipaddr}</code>	Resolves to the agent IP address (see also <code>\${ops_agent_ip}</code>).
<code>\${ops_agent_ip}</code>	Resolves to the agent IP address. You can also use this variable in task notifications; see Creating Email Notifications and Creating SNMP Notifications .
<code>\${ops_agent_name}</code>	Resolves to the agent name. You can also use this variable in task notifications; see Creating Email Notifications and Creating SNMP Notifications .
<code>\${ops_agent_mode}</code>	Resolves to the agent operational mode (Active, Offline).

Cluster Node Variables

The following Cluster Node variables allow you to pass information into a [cluster node \(server\) notification](#):

Field Name	Description
<code>\${ops_cluster_start_time}</code>	Resolves to the date and time the cluster node (server) was started. For example: <pre>ops_cluster_start_time = 2011-09-26 17:35:01 -0400</pre>

<code>\${ops_cluster_id}</code>	Resolves to the cluster node's internally-generated build ID. For example:
	<code>ops_cluster_id = MACHINEC19A:8080-opwise</code>
<code>\${ops_cluster_uptime}</code>	Resolves to the numbers of days, hours, and minutes that this cluster node has been running since it was last started. For example:
	<code>ops_cluster_uptime = 7 Seconds</code>
<code>\${ops_cluster_name}</code>	Resolves to the URL, or Node ID, of this cluster node. For example:
	<code>ops_cluster_name = MACHINEC19A:8080-opwise</code>
<code>\${ops_cluster_hostname}</code>	Resolves to the hostname of this cluster node. For example:
	<code>ops_cluster_hostname = MACHINEC19A</code>
<code>\${ops_cluster_ipaddr}</code>	Resolves to the IP address of this cluster node. For example:
	<code>ops_cluster_ipaddr = 10.N.N.NN</code>
<code>\${ops_cluster_mode}</code>	Resolves to the current mode of this cluster node. One of the following: Offline, Active, Passive/Available, Passive/Unavailable. For example:
	<code>ops_cluster_mode = Active</code>
	For more information, see Viewing Node Status .

Connector Variables

The following Connector (Message Hub or Transporter) variables allow you to pass information into a [connector notification](#).

Field Name	Description
<code>\${ops_connector_hostname}</code>	Resolves to the connector hostname.

<code>\${ops_connector_id}</code>	Resolves to the connector queue name.
<code>\${ops_connector_ipaddr}</code>	Resolves to the connector IP address.
<code>\${ops_connector_name}</code>	Resolves to the connector name.
<code>\${ops_connector_mode}</code>	Resolves to the connector operational mode (Active, Offline).

System Monitor Variables

The following System Monitor variables show the results for "Resource Available" and "Actual Available" that can be utilized in [System Monitor](#) tasks.

Field Name	Description
<code>\${ops_sm_size}</code>	Size specified in the Resource Available field of the System Monitor task definition.
<code>\${ops_sm_int_size}</code>	Same as <code>ops_sm_size</code> , except that <code>ops_sm_int_size</code> is rounded to the nearest integer.
<code>\${ops_sm_scale}</code>	Scale specified in the By Scale field for Resource Available of the System Monitor task definition.
<code>\${ops_sm_actual_size}</code>	Actual size determined by the agent.
<code>\${ops_sm_actual_int_size}</code>	Same as <code>ops_sm_actual_size</code> , except that <code>ops_sm_actual_int_size</code> is rounded to the nearest integer.
<code>\${ops_sm_actual_scale}</code>	Scale of the actual size determined by the agent.

Launching With Variables

For information on how to launch a task with variables, see [Provide Temporary Variable Values and Launch a Task Manually on the Manually Running and Controlling Tasks](#) page.

Trigger With Variables

For information on how to use variables when manually launching tasks associated with a trigger, see [Triggering with Variables](#) (in the [Triggers and Calendars](#) section of this documentation).

Setting Variables within a Workflow

- [Introduction](#)
- [Creating a Set Variable Instruction](#)
- [Set Variables Field Descriptions](#)

Introduction

Within a workflow, you can specify instructions that set a variable to a specific value for use within the workflow. Unless you set a Global variable (see [Variable Scope](#), below), which can be accessed by any task instance or workflow instance, the value you set using this method exists only in memory for the period this workflow is running or until another Set Variable instruction sets it to another value.

You can create the Set Variable instruction at the workflow level or task level; that is, you can attach the Set Variable action to the workflow or a task running within the workflow.

- If you set it at the workflow level, you have the option of setting it based on information in the workflow record, on any of the tasks (children) in the workflow, or both.
- If you set it at the task level, you have the option of setting it based on information in the specific task, in the parent workflow, or both.

The variable need not exist in the database; you can create a new variable using the Set Variable instruction.

When creating a Set Variable instruction, you can trigger the Set Variable action based on one or more of the following:

- Status
- Exit codes
- Late start
- Late or early finish

Creating a Set Variable Instruction

Step 1	Display the workflow for which you want to create Set Variable instructions.
Step 2	Click the Actions tab. This Actions List screen displays a list of actions defined for this workflow.
Step 3	Click New . The Actions Wizard screen displays. <div data-bbox="224 1220 1373 1591" data-label="Image"> </div>

Step 4 Click **Set Variable**. The Set Variables screen displays.

Step 5 Using the field descriptions provided below as a guide, complete the fields as needed.

Step 6 Click the **Submit** button to save the record and return to the Actions list, or, right-click and select **Save** to save the record and remain on the current display.

Step 7 If appropriate, repeat these steps for any additional Set Variable instructions you want to add.

Set Variables Field Descriptions

The table below describes the fields and buttons on the Set Variables screen.

Field Name	Description
Type Details	<p>Displays - on the Actions List screen - the following information for this action:</p> <ul style="list-style-type: none"> scope (SELF, PARENT, TOP_LEVEL_PARENT, or GLOBAL) name value
Action Inheritance	<p>Workflow tasks only. Specifies what records these instructions apply to. Options:</p> <ul style="list-style-type: none"> SELF - These instructions apply only to the workflow and are not inherited by its children tasks. SELF/CHILDREN - These instructions apply to the workflow and its contained tasks (children). CHILDREN - These instructions apply only to the tasks within the workflow (children).

Status	<p>The status(es) that will trigger the action. To trigger a Set Variable action, you can specify status only, or status and exit code. You can specify as many statuses as needed. Options:</p>
Status	Description
Defined	All task types. The new task instance has been created (the task has been launched). Not yet implemented.
Waiting	All task types. The task has been loaded by a workflow and is waiting to run.
Held	All task types. The task has been put on hold by a user.
Resource Requested	All tasks with a virtual resource defined go immediately to a status of Resource Requested. If the resource is available, the task then moves to the next appropriate processing status.
Resource Wait	All tasks with a virtual resource defined go immediately to a status of Resource Requested. If the resource is not available, the task goes to a status of Resource Wait. When the resource becomes available, the task moves to the next appropriate processing status
Execution Wait	Agent-based tasks. The task must wait to be completed; either the Agent/Agent Cluster running the task has reached its Task Execution Limit, or the ability of the Agent/Agent Cluster to run tasks has been suspended.
Undeliverable	Agent-based tasks. The agent is unavailable.
Queued	Agent-based tasks only. The task has been queued on a resource.
Submitted	z/OS only. The task has been submitted to the z/OS Job Entry subsystem and scheduled by the z/OS Job Scheduler.
Action Required	Manual tasks only. When a Manual task launches, it goes into Action Required status, meaning a user must perform some manual activity.
Started	Agent-based and Manual tasks only. The task has started. For agent-based tasks, this means the agent has received the task.
Running	All task types. The task is running. For agent-based tasks, the agent has started running the program.
Running Problems	<p>Workflows only. One or more tasks within the workflow has one of the following statuses:</p> <ul style="list-style-type: none"> • Held • Undeliverable • Running Problems (for sub-workflows) • Cancel Pending • In Doubt • Start Failure • Cancelled
In Doubt	Agent-based tasks only. The agent is "in doubt" about the current status of the task instance. This may occur if an agent or agent connection goes down. In this case, the agent restarts and reviews its data about tasks in progress. If the agent finds a task still running, it resumes normal monitoring. If the agent cannot find the task, this usually indicates that the task completed, but the agent considers the task status to be "in doubt."
Start Failure	All task types. The task was unable to start.
Confirmation Required	z/OS only. If you make JCL changes and restart a z/OS task, Opwise will put the task into Confirmation Required status and prompt you for a confirmation. For detailed processing steps, see Rerunning a z/OS Task .
Cancelled	All task types. The task was cancelled by a user.
Failed	All task types. The task ran to a failure status.
Skipped	All task types. The task was skipped by a user.
Finished	All task types. The task was forced by the user to finish. The user may do this in cases where the task had "Cancelled" or "Failed" status, and the user needed to release other task instances depending on the successful completion of this task instance in a workflow. For more information, see Force Finishing a Task .
Success	All task types. The task has completed successfully.

Exit Codes	Specifies one or more exit codes that will trigger the event. If you specify an exit code, you must also specify at least one status. Use commas to separate multiple exit codes; use a hyphen to specify a range. Example: 1, 5, 22-30.
On Late Start	Generate the action or notification if the task started late, based on the Late Start Time specified in the task.
On Late Finish	Generate the action or notification if the task finishes late, based on the Late Finish time specified in the task.
On Early Finish	Generate the action or notification if the task finishes early, based on the Early Finish Time specified in the task.
Variable Scope	Applies to variables associated with a task in a workflow. Options: <ul style="list-style-type: none"> • SELF - The variable is set only within the scope of the task that executes the Set Variable action. • PARENT - The variable is set within the scope of the (immediate) parent workflow. After it is set, any task within the parent workflow can access that variable. • TOP_LEVEL_PARENT - The variable is set within the scope of the top level parent. Example: Workflow A contains workflow B and workflow B contains workflow C. If a Set Variable action is executed by a task within workflow C with Variable Scope set to TOP_LEVEL_PARENT, then the variable will be set in workflow A's scope. This means that after it is set, tasks in workflow A, workflow B and workflow C can access that variable. • GLOBAL - The variable is set at the global variable level and, as such, is accessible by any task or workflow instance. If the global variable is not already defined, it will be created.
Name	Required. Name of variable. Up to 40 alphanumeric. The name must begin with an alphabetic character and can consist of: alphas (a-z, A-Z), numerics 0-9, _ (underscore). White spaces are not permitted; names are not case-sensitive. Important Note – Do not define Opwise variables with the prefix "ops_". That prefix is reserved for built-in variables.
Value	Optional. The value of the variable.
Description	Optional. Description of this variable.
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Delete button	Deletes the current record.

Listing and Setting Variables from the Command Line

To list and set variables from the command line, use the `ops-variable-list` and `ops-variable-set` commands of the Opswise [Command Line Interface \(CLI\)](#).

Functions

- Overview
- Formatting Rules
- Tables of Functions
 - Date Functions
 - Mathematical Functions
 - System Functions
 - String Functions
 - SQL/Stored Procedure Functions

Overview

Opwise supports a number of functions that can be specified in free text fields. They are resolved when a task instance runs or when a [Set Variable](#) action containing a function is executed.

Functions are entered using the following formats:

```
$_function
$_function(arg1, ..., argN)
```

Formatting Rules

- Functions must be written either in all lower case or exactly as shown in the following tables.
- Any parameter can be quoted. Strings must be quoted with single or double quotes.
- All functions allow nesting to one level. That is, a function can be an argument to another function. A nested function has the format:

```
$_function(arg1, ..., argN)
```

- For nested functions, make sure you use a double underscore preceding the function name. For example:

```
$_substring("$_ops_trigger_file_name_simple", "$_indexOf("$_ops_trigger_file_name_simple",
"-")})")})
```



Tables of Functions

Functions are categorized into five separate tables, below:

- Date functions
- Mathematical functions
- System functions
- String functions
- SQL/Stored Procedure functions

Date Functions

Format	Description
--------	-------------

<pre> \${_businessDayOfMonth(index, [date, format, reverse])} </pre>	<p>Returns the Nth business day of month for the month of the date specified. Optionally, can start from the end of the month.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • <code>index</code> Nth business day of month. • <code>date</code> Date in format yyyy-MM-dd. Default is the current date. • <code>format</code> Format of returned date. Default is yyyy-MM-dd. (For details on the <code>format</code> parameter, see http://docs.oracle.com/javase/6/docs/api/java/text/SimpleDateFormat.html) • <code>reverse</code> Specification (<code>true</code> or <code>false</code>) for starting from the end of the month. Default is <code>false</code>. <p>Examples:</p> <pre> \${_businessDayOfMonth(1)} --> 2012-08-01 \${_businessDayOfMonth(1,"2012-09-01")} --> 2012-09-04 \${_businessDayOfMonth(1,"2012-09-01","",true)} --> 2012-09-28 </pre> <p> Whether a holiday is treated as a business day or a non-business day is specified by the Exclude Holidays for Business Days Opswise system property.</p>
<pre> \${_businessDaysBetween(date1, date2)} </pre>	<p>Returns the number of business days between <code>date1</code> and <code>date2</code>.</p> <ul style="list-style-type: none"> • If return value is <code>> 0</code>, <code>date2</code> is after <code>date1</code>. • If return value is <code>< 0</code>, <code>date2</code> is before <code>date1</code>. • If return value is <code>0</code>, <code>date1</code> is equal to <code>date2</code>. <p>The start date is inclusive, but the end date is not.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • <code>date1</code> First date in format yyyy-MM-dd. • <code>date2</code> Second date in format yyyy-MM-dd. <p>Example:</p> <pre> \${_businessDaysBetween("2012-08-01","2012-09-01")} --> 23 </pre> <p> Whether a holiday is treated as a business day or a non-business day is specified by the Exclude Holidays for Business Days Opswise system property.</p>
<pre> \${_currentTimeMillis} </pre>	<p>Resolves to the current time in milliseconds.</p>


```

${_date([format, day_offset,
hour_offset, minute_offset])}

```

Resolves to the current date and time. All parameters are optional.

Parameters:

- **format**
Date format. The default format is yyyy-MM-dd HH:mm:ss Z. For details on the format parameter, see <http://docs.oracle.com/javase/6/docs/api/java/text/SimpleDateFormat.html>
- **day_offset**
+/- number of days to offset.
- **hour_offset**
+/- number of hours to offset.
- **minute_offset**
+/- number of minutes to offset.

Examples:

```

${_date} --> 2012-07-14 12:43:06 -0400
${_date()} --> 2012-07-14 12:43:06 -0400
${_date("yyyy-MM-dd", 5)} --> 2012-07-19
${_date("yyyy-MM-dd HH:mm:ss", -2, -1)}
--> 2012-07-12 11:43:06
${_date("", 0, 0, 10)} --> 2012-07-14 12:53:06 -0400

```

```

${_dateadv([format,
year_offset,
month_offset, day_offset,
hour_offset, minute_offset])}

```

Resolves to the current date and time. All parameters are optional.

Parameters:

- **format**
Date format. The default format is yyyy-MM-dd HH:mm:ss Z. For details on the format parameter, see <http://docs.oracle.com/javase/6/docs/api/java/text/SimpleDateFormat.html>
- **year_offset**
+/- number of years to offset.
- **month_offset**
+/- number of months to offset.
- **day_offset**
+/- number of days to offset.
- **hour_offset**
+/- number of hours to offset.
- **minute_offset**
+/- number of minutes to offset.

Examples:

```

${_dateadv} --> 2012-07-29 09:31:42 -0700
${_dateadv("yyyy-MMM", -1)} --> 2011-Jul
${_dateadv("yyyy-MMM", 0, -1)}
--> 2012-Jun

```

<pre> \${_dayOfMonth(index, [date, format, reverse])} </pre>	<p>Returns the Nth day of month for the month of the date specified. Optionally, can start from the end of the month.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • <code>index</code> Nth day of month. • <code>date</code> Date in format yyyy-MM-dd. Default is the current date. • <code>format</code> Format of returned date. Default is yyyy-MM-dd. • <code>reverse</code> Specification (<code>true</code> or <code>false</code>) for starting from the end of the month. Default is <code>false</code>. <p>Examples:</p> <pre> \${_dayOfMonth(5)} --> 2012-08-05 \${_dayOfMonth(15,"2012-09-01","MM/dd/yyyy")} --> 09/15/2012 \${_dayOfMonth(1,"2012-09-01","",true)} --> 2012-09-30 </pre>
<pre> \${_dayOfWeek([date, first_dow, first_dow_value])} </pre>	<p>Returns the day of week for the specified date as a number.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • <code>date</code> Date in format yyyy-MM-dd. Default is the current date. • <code>first_dow</code> Specification for whether the week starts on Sunday or Monday. Values are sun and mon (not case-sensitive). Default is sun. • <code>first_dow_value</code> Starting value for the first day of week. Value must be a non-negative number. Default is 1. <p>Examples:</p> <pre> \${_dayOfWeek} --> 6 \${_dayOfWeek()} --> 6 \${_dayOfWeek("2012-07-04")} --> 4 \${_dayOfWeek("2012-07-04", "mon")} --> 3 </pre>
<pre> \${_daysBetween(date1, date2)} </pre>	<p>Returns the number of days between date1 and date2.</p> <ul style="list-style-type: none"> • If return value is > 0, date2 is after date1. • If return value is < 0, date2 is before date1. • If return value is 0, date1 is equal to date2. <p>The start date is inclusive, but the end date is not.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • <code>date1</code> First date in format yyyy-MM-dd. • <code>date2</code> Second date in format yyyy-MM-dd. <p>Example:</p> <pre> \${_daysBetween("2012-08-01","2012-09-01")} --> 31 </pre>

```

${_formatDate([date, format,
day_offset, use_business_days,
hour_offset, minute_offset])}

```

Returns the date after applying offsets. Optionally, can specify the output format.

Parameters:

- `date`
Date in format yyyy-MM-dd HH:mm or yyyy-MM-dd. Time (HH:mm) is optional. Default is the current date and time.
- `format`
Format of returned date. Default is the format used when specifying the date parameter: yyyy-MM-dd HH:mm or yyyy-MM-dd. For details on the `format` parameter, see <http://docs.oracle.com/javase/6/docs/api/java/text/SimpleDateFormat.html>
- `day_offset`
+/- number of days to offset.
- `use_business_days`
Specification (`true` or `false`) for whether `day_offset` is for business days. Default is `false`.
- `hour_offset`
+/- number of hours to offset.
- `minute_offset`
+/- number of minutes to offset.

Examples:

```

${_formatDate} --> 2012-08-24 15:37
${_formatDate()} --> 2012-08-24 15:37
${_formatDate("", "MMddyyyy", 5)} --> 08292012
${_formatDate("2012-09-01", "", 5)} --> 2012-09-06
${_formatDate("2012-09-01", "", -5)} --> 2012-08-27
${_formatDate("2012-09-01", "", 5, true)} --> 2012-09-10

```



Whether a holiday is treated as a business day or a non-business day is specified by the Exclude Holidays for Business Days [Opswise system property](#).

```

${_formatDateAdv([date, format,
year_offset, month_offset,
day_offset, use_business_days,
hour_offset, minute_offset])}

```

Returns the date after applying offsets. Optionally, can specify the output format.

Parameters:

- **date**
Date in format yyyy-MM-dd HH:mm or yyyy-MM-dd. Time (HH:mm) is optional. Default is the current date and time.
- **format**
Format of returned date. Default is the format used when specifying the date parameter: yyyy-MM-dd HH:mm or yyyy-MM-dd. For details on the `format` parameter, see <http://docs.oracle.com/javase/6/docs/api/java/text/SimpleDateFormat.html>
- **year_offset**
+/- number of years to offset.
- **month_offset**
+/- number of months to offset.
- **day_offset**
+/- number of days to offset.
- **use_business_days**
Specification (`true` or `false`) for whether `day_offset` is for business days. Default is `false`.
- **hour_offset**
+/- number of hours to offset.
- **minute_offset**
+/- number of minutes to offset.

Examples:

```

${_formatDateAdv} --> 2012-08-24 15:55
${_formatDateAdv()} --> 2012-08-24 15:55
${_formatDateAdv("", "MMdyyy", 1)} --> 08242013
${_formatDateAdv("2012-09-01", "", 0, 1)} --> 2012-10-01
${_formatDateAdv("2012-09-01", "", 0, -1)} --> 2012-08-01
${_formatDateAdv("2012-09-01", "", 0, 0, 5, false)} -->
2012-09-06

```



Whether a holiday is treated as a business day or a non-business day is specified by the Exclude Holidays for Business Days [Opwise system property](#).

```

${_nonBusinessDayOfMonth(index,
[date, format, reverse])}

```

Returns the Nth non-business day of month for the month of the date specified. Optional can start from the end of the month.

Parameters:

- `index`
Nth non-business day of month.
- `date`
Date in format yyyy-MM-dd. If blank, defaults to the current date.
- `format`
Format of returned date. Default is yyyy-MM-dd. For details on the `format` parameter, see <http://docs.oracle.com/javase/6/docs/api/java/text/SimpleDateFormat.html>
- `reverse`
Specification (`true` or `false`) for starting from the end of the month. Default is `false`.

Examples:

```

${_nonBusinessDayOfMonth(1)} --> 2012-08-04
${_nonBusinessDayOfMonth(1,"2012-09-01")} --> 2012-09-01
${_nonBusinessDayOfMonth(1,"2012-09-01","",true)} -->
2012-09-30

```



Whether a holiday is treated as a business day or a non-business day is specified by the Exclude Holidays for Business Days [Opswise system property](#).

Mathematical Functions

Format	Description
<code>\${_mod(dividend, divisor)}</code>	Return the modulo or remainder of the dividend divided by divisor.
<code>\${_add(augend, addend)}</code>	Return the sum of the augend added with the addend.
<code>\${_subtract(minuend, subtrahend)}</code>	Return the difference of the subtrahend subtracted from the minuend.
<code>\${_multiply(multiplicand, multiplier)}</code>	Return the product of the multiplicand multiplied with the multiplier.
<code>\${_divide(dividend, divisor)}</code>	Return the quotient of the dividend divided by divisor.
<code>\${_abs(parameter)}</code>	Return the absolute value of the parameter.

System Functions

Format	Description
<code>\$_guid</code>	Resolves to a 32-byte GUID (Globally Unique ID).
<code>\$_hostname</code>	Resolves to the hostname of the machine running the Controller, if available.
<code>\$_ipaddress</code>	Resolves to the IP address of the machine running the Controller.
<code>\$_random([max, min])</code>	<p>Generates a random number between <code>max</code> (inclusive) and <code>min</code> (inclusive).</p> <p>Parameters:</p> <ul style="list-style-type: none"> <code>max</code> Upper bound (inclusive) on the random number (default = 9). <code>min</code> Lower bound (inclusive) on the random number (default = 0).
<code>\$_resolve(variable_name, default_value)</code>	<p>Resolves the variable specified by the <code>variable_name</code> parameter and substitutes the default value if the variable cannot be resolved.</p> <p>Parameters:</p> <ul style="list-style-type: none"> <code>variable_name</code> Variable name. <code>default_value</code> Default value to use if the variable cannot be resolved.
<code>\$_resolveadv(variable_name, default_value, [use_default_if_blank])</code>	<p>Resolves the variable specified by the <code>variable_name</code> parameter and substitutes the default value if the variable cannot be resolved.</p> <p>Parameters:</p> <ul style="list-style-type: none"> <code>variable_name</code> Variable name. <code>default_value</code> Default value to use if the variable cannot be resolved. <code>use_default_if_blank</code> Specification (<code>true</code> or <code>false</code>) for whether or not to use the default value if the variable is empty or blank. (If <code>use_default_if_blank</code> is <code>false</code>, <code>\$_resolveadv</code> behaves like <code>\$_resolve</code>.)
<code>\$_scope</code>	<p>Displays all the defined and built-in variables associated with the task instance.</p> <p>Example:</p> <pre>\$_scope --> {ops_workflow_id=, ops_task_type=Unix, ops_status=DEFINED, ops_retry_interval=60, ops_exit_code=0, ops_retry_maximum=0, ops_cmd_parms=, ops_cmd=ls -la; exit \$_random('9')}; ops_retry_count=0, ops_agent_id=67e4994143d2617201cdf4ba9df9ab0a, ops_task_id=84880af243d26172019aald25988a8f9, ops_task_name=Opwise - Linux Ls}</pre>


<pre>\$_siblingid(sibling_name)</pre>	<p>Resolves to the <code>sys_id</code> of the first task instance found within the same workflow specified by the sibling name.</p> <p>Parameters:</p> <ul style="list-style-type: none"> <code>sibling_name</code> Sibling name. <p>Example:</p> <pre>\$_siblingid("Sleep 60") --> 5dbaaab943d26172015e10ab3e894e10</pre>
<pre>\$_varLookup(sibling_name, variable_name[,def])</pre>	<p>Locates the specified variable in the specified sibling task instance within the same workflow and resolves to the variable value.</p> <p>Parameters:</p> <ul style="list-style-type: none"> <code>sibling_name</code> Name of the sibling task instance from which the function is collecting the variable value. <code>variable_name</code> Name of the variable being collected by the function. <code>def</code> Optional; default value to return if the variable is not defined in the sibling task instance.




String Functions


String functions pass in either a **value** or a **variable**; for each String function that passes in a **value**, there is a corresponding String function that passes in a **variable**.




String functions that pass in a variable are prefixed with `_var`. The variables must be fully resolved; they cannot resolve to a function.


In the following table, the name of each String function that passes a **value** and the name of the corresponding String function that passes a **variable** link to each other.

Format	Description
<pre>\$_indexOf(value, str)</pre>	<p>Returns the index within the string value of the first occurrence of the specified substring, <code>str</code>.</p> <p>Parameters:</p> <ul style="list-style-type: none"> <code>value</code> Any string. <code>str</code> Substring to search for. If the <code>str</code> argument occurs as a substring within the <code>value</code>, then the index of the first character of the first such substring is returned; if it does not occur as a substring, -1 is returned. <div data-bbox="646 1570 1446 1703" style="background-color: #ffffcc; padding: 5px;"> <p> Note Indexing functions use zero-based numbering; that is, the initial element is assigned the index 0.</p> </div>

<pre> \${_indexOfWithOffset (value, str, offset)} </pre>	<p>Returns the index within this string of the first occurrence of the specified substring plus the specified offset. The integer returned is the smallest value.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • <code>value</code> Any string. • <code>str</code> Substring to search for. If the <code>str</code> argument occurs as a substring within the value, then the index of the first character of the first such substring is returned; if it does not occur as a substring, -1 is returned. • <code>offset</code> Number (positive or negative) to offset the found index. <p> Note Indexing functions use zero-based numbering; that is, the initial element is assigned the index 0.</p>
<pre> \${_lastIndexOf(value, str)} </pre>	<p>Returns the index within the string value of the rightmost occurrence of the specified substring, <code>str</code>.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • <code>value</code> Any string. • <code>str</code> Substring to search for. If the <code>str</code> argument occurs one or more times as a substring within the value, then the index of the first character of the last such substring is returned. If it does not occur as a substring, -1 is returned. <p> Note Indexing functions use zero-based numbering; that is, the initial element is assigned the index 0.</p>
<pre> \${_lastIndexOfWithOffset (value, str, offset)} </pre>	<p>Returns the index within this string of the rightmost occurrence of the specified substring, plus the specified offset. The returned index is the largest value.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • <code>value</code> Any string. • <code>str</code> Substring to search for. If the <code>str</code> argument occurs as a substring within the value, then the index of the first character of the first such substring is returned; if it does not occur as a substring, -1 is returned. • <code>offset</code> Number (positive or negative) to offset the found index. <p> Note Indexing functions use zero-based numbering; that is, the initial element is assigned the index 0.</p>
<pre> \${_length(value)} </pre>	<p>Returns the length of <code>value</code>.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • <code>value</code> Any string

<pre> \${_replaceAll(value, regex, replacement)} </pre>	<p>Replaces each substring of value that matches the specified regular expression, <code>regex</code>, with the specified replacement.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • <code>value</code> Input string. • <code>regex</code> Regular expression. • <code>replacement</code> Replacement string.
<pre> \${_substring(value, begin_index[, end_index])} </pre>	<p>Returns a new string that is a substring of value. The substring begins at <code>begin_index</code> and extends to the character at <code>end_index - 1</code>.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • <code>value</code> String to make a substring from. • <code>begin_index</code> Beginning index, inclusive. • <code>end_index</code> Ending index, exclusive. <p>Examples:</p> <pre> \${_substring("hamburger", 4, 8)} resolves to "urge". \${_substring("smiles", 1, 5)} resolves to "mile". </pre> <div data-bbox="649 1050 1446 1182" style="background-color: #ffffcc; padding: 5px;"> <p> Note Indexing functions use zero-based numbering; that is, the initial element is assigned the index 0.</p> </div>
<pre> \${_toLowerCase(value)} </pre>	<p>Converts all of the characters in the <code>value</code> to lower case using the rules of the default locale.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • <code>value</code> String to convert to lower case.
<pre> \${_toUpperCase(value)} </pre>	<p>Converts all of the characters in the <code>value</code> to upper case using the rules of the default locale.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • <code>value</code> String to convert to upper case.
<pre> \${_trim(value)} </pre>	<p>Returns a copy of <code>value</code>, with leading and trailing whitespace omitted.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • <code>value</code> String to trim.

<pre> \${_varIndexOf (variableName, str)} </pre>	<p>Returns the index within the string variable of the first occurrence of the specified substring, <code>str</code>.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • <code>variableName</code> Variable that this function is passing in. • <code>str</code> Substring to search for. If the <code>str</code> argument occurs as a substring within the variable, then the index of the first character of the first such substring is returned; if it does not occur as a substring, -1 is returned. <p> Note Indexing functions use zero-based numbering; that is, the initial element is assigned the index 0.</p>
<pre> \${_varIndexOfWithOffset (variableName, str, offset)} </pre>	<p>Returns the index within this string of the first occurrence of the specified substring plus the specified offset. The integer returned is the smallest variable.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • <code>variableName</code> Variable that this function is passing in. • <code>str</code> Substring to search for. If the <code>str</code> argument occurs as a substring within the variable, then the index of the first character of the first such substring is returned; if it does not occur as a substring, -1 is returned. • <code>offset</code> Number (positive or negative) to offset the found index. <p> Note Indexing functions use zero-based numbering; that is, the initial element is assigned the index 0.</p>
<pre> \${_varLastIndexOf (variableName, str)} </pre>	<p>Returns the index within the string variable of the rightmost occurrence of the specified substring, <code>str</code>.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • <code>variableName</code> Variable that this function is passing in. • <code>str</code> Substring to search for. If the <code>str</code> argument occurs one or more times as a substring within the variable, then the index of the first character of the last such substring is returned. If it does not occur as a substring, -1 is returned. <p> Note Indexing functions use zero-based numbering; that is, the initial element is assigned the index 0.</p>

<pre> \${ _varLastIndexOfWithOffset (variableName, str, offset)} </pre>	<p>Returns the index within this string of the rightmost occurrence of the specified substring, plus the specified offset. The returned index is the largest variable.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • <code>variableName</code> Variable that this function is passing in. • <code>str</code> Substring to search for. If the <code>str</code> argument occurs as a substring within the variable, then the index of the first character of the first such substring is returned; if it does not occur as a substring, -1 is returned. • <code>offset</code> Number (positive or negative) to offset the found index. <div style="background-color: #ffffcc; padding: 5px;"> <p> Note Indexing functions use zero-based numbering; that is, the initial element is assigned the index 0.</p> </div>
<pre> \${_varLength (variableName[, useEmptyForUndefined])} </pre>	<p>Returns the length of <code>variableName</code>.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • <code>variableName</code> Variable that this function is passing in. • <code>useEmptyForUndefined</code> Optional; Specification (true or false) for the handling of a missing variable name. Default is false. • If <code>useEmptyForUndefined = true</code>, the function will return 0. • If <code>useEmptyForUndefined = false</code>, the function will remain unresolved if the variable name does not exist.
<pre> \${_varReplaceAll (variableName, regex, replacement)} </pre>	<p>Replaces each substring of <code>variableName</code> that matches the specified regular expression, <code>regex</code>, with the specified replacement.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • <code>variableName</code> Variable that this function is passing in. • <code>regex</code> Regular expression. • <code>replacement</code> Replacement string.

<pre> \${_varSubstring (variableName, beginIndex, endIndex)} </pre>	<p>Returns a new string that is a substring of <code>variableName</code>. The substring begins at <code>begin_index</code> and extends to the character at <code>{end_index}</code> -1.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • <code>variableName</code> Variable that this function is passing in. • <code>begin_index</code> Beginning index, inclusive. • <code>end_index</code> Ending index, exclusive. <p>Examples:</p>
<pre> \${_varToLowerCase (variableName)} </pre>	<p>Converts all of the characters in the variable to lower case using the rules of the default locale.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • <code>variableName</code> Variable that this function is passing in.
<pre> \${_varToUpperCase (variableName)} </pre>	<p>Converts all of the characters in the variable to upper case using the rules of the default locale.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • <code>variableName</code> Variable that this function is passing in.
<pre> \${_varTrim (variableName)} </pre>	<p>Returns a copy of <code>variableName</code>, with leading and trailing whitespace omitted.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • <code>variableName</code> Variable that this function is passing in.

**Note**

Indexing functions use zero-based numbering; that is, the initial element is assigned the index 0.

SQL/Stored Procedure Functions

Format	Description
<pre> \${_resultsAll([separator, rowSeparator])} </pre>	<p>Returns all SQL results from the current SQL or Stored Procedure task. Columns are separated by the specified <code>separator</code> and rows are separated by a new line.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • <code>separator</code> Column separator (default = comma). • <code>rowSeparator</code> Overrides default New Line character.

<pre> \${_resultsAllFromTask(name [, separator, rowSeparator])} </pre>	<p>Returns all SQL results from a sibling SQL or Stored Procedure task, within the same workflow. Columns are separated by the specified <code>separator</code> and rows are separated by a new line.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • <code>name</code> Name of the sibling task that the results should come from. The task must be within the same workflow. • <code>separator</code> Column separator (default = comma). • <code>rowSeparator</code> Overrides default New Line character.
<pre> \${_resultsColumn(name, colname[, rownum, default_value])} </pre>	<p>Returns the string value of a row/column from a previously executed SQL task within the same workflow, or from the current SQL task.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • <code>name</code> Name of a sibling SQL task within the same workflow from which you want the function to fetch results. If you want to execute the function against the current task, use an empty string for the name parameter. • <code>colname</code> Name of column to retrieve. • <code>rownum</code> Numeric row number in result set to retrieve (default = 1). • <code>default_value</code> Default value to return if result not found.
<pre> \${_resultsColumnByNo(name, colnum[, rownum, default_value])} </pre>	<p>Returns the string value of a row/column from a previously executed SQL task within the same workflow, or from the current SQL task.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • <code>name</code> Name of a sibling SQL task within the same workflow from which you want the function to fetch results. If you want to execute the function against the current task, use an empty string for the name parameter. • <code>colnum</code> Number of column to retrieve. First column in result is 1, second is 2, and so on. • <code>rownum</code> Numeric row number in result set to retrieve (default = 1). • <code>default_value</code> Default value to return if result not found.
<pre> \${_resultsColumnsCSV (name[, rownum])} </pre>	<p>Returns the string values of columns in a specific row in CSV (comma-separated values) format, from a previously executed SQL task within the same workflow, or from the current SQL task.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • <code>name</code> Name of a sibling SQL task within the same workflow from which you want the function to fetch results. If you want to execute the function against the current task, use an empty string for the name parameter. • <code>rownum</code> Numeric row number in result set to retrieve (default = 1).
<pre> \${_SQLWarnings([separator])} </pre>	<p>Returns all SQL warnings from the current SQL or Stored Procedure task. Columns are separated by the specified <code>separator</code> and rows are separated by a new line.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • <code>separator</code> Column separator (default = comma).

```
$_SQLWarningsFromTask(name  
[, separator])
```

Returns all SQL warnings from a sibling SQL or Stored Procedure task, within the same workflow. Columns are separated by the specified `separator` and rows are separated by a new line.

Parameters:

- `name`
Name of the sibling task that the warnings should come from. The task must be within the same workflow.
- `separator`
Column separator (default = comma).

Triggers and Calendars



Triggers

Creating Triggers

Types of Triggers

Enabling and Disabling Triggers

Copying Triggers

Triggering with Variables



Calendars

Overview

Creating Calendars

Copying Calendars

Creating Custom Days



Displaying Trigger F

Overview

Forecast Calendar

Forecast List

Forecast Calculation

Forecast Re-Calculation

Setting up Forecasting

Next Scheduled Time

List Qualifying Times



The information on these pages also is located in the [Opwise Automation Center 5.1.1 User Guide.pdf](#).

Creating Triggers

- Triggers
 - Trigger Types
- Accessing a Triggers List
- Creating a Trigger
- Daylight Saving Time
 - Interval-Based Times
 - Absolute Times

Triggers

A trigger specifies times and/or events that trigger the launching one or more tasks.

When a trigger is satisfied, Opwise launches the tasks specified in the trigger. Each trigger can have an unlimited number of tasks associated with it. All of the specified tasks are run each time the trigger is satisfied. If you want to specify dependencies such as "run Task B only if Task A fails," create a [Workflow](#), which is a series of inter-connected tasks.

A built-in [trigger variable](#) is available for returning the trigger name. Additional [built-in variables](#) are supported for specific trigger types.

Trigger Types

Trigger Type	Usage
Cron	Specify dates and times, using Cron syntax, at which a task will be triggered.
Time	Specify dates and times at which a task will be triggered.
Manual	Launch task(s) immediately, while setting or overriding the value of one or more user-defined variables specified in the task(s).
Temporary	Set up a one-time trigger for a task, based on a single date and time.
File Monitor	Trigger one or more tasks based on the creation, deletion, change, existence or non-existence of a file on a particular machine.
Task Monitor	Trigger one or more tasks based on the conditions specified in an associated Task Monitor task.
Application Monitor	Trigger one or more tasks based on the status of one or more application resources.

Accessing a Triggers List

Opwise provides two lists of triggers:

- All triggers
- Active triggers

Active Triggers are triggers that have their Enabled flag set.

To access either list, select **Automation Center > Triggers > All Triggers** or **Automation Center > Triggers > Active Triggers**, as appropriate, from the navigation pane.

The following illustrates an All Triggers screen:

Trigger Name	Type	Description	Task(s)	Next Scheduled Time	Enabled
BROADCAST TMT	Task Monitor		BROADCAST_WF		✓
mytest.trigger	File Trigger		Sleep 30		✗
Opwise - 1st and 15th	Cron	4:45 on the 1st and 15th of each month	Opwise - Sleep 30		✗
Opwise - 1st Business Day of Year	Time	Run 1st Business day of the Year	Opwise - Sleep 30		✗
Opwise - 1st Sunday of Month	Time	Run the 1st Sunday of the Month	Opwise - Sleep 60		✗
Opwise - Business Days	Time	Business Days every hour	Opwise - Create Text File To Kick Off File Trigger		✗
Opwise - Daily	Time	Daily at 8 am PST	Opwise - Windows Snooze		✗
Opwise - Every 15 Minutes MWF 9-5	Time	Every 15 minutes MWF between 9-5	Opwise - Workflow of Sleep Tasks		✗
Opwise - Every 5 Minutes	Cron	Every 5 Minutes of Every Day	Opwise - Sleep 60		✗
Opwise - Every Friday 13th	Cron	Every Friday the 13th	Opwise - Sleep 0		✗
Opwise - Every Independence Day	Time	Run every Independence Day	Opwise - Workflow of Sleep Tasks		✗
Opwise - File Creation Trigger	File Trigger	Test File Monitor in Trigger and in Workflow	Opwise - Demo		✗
Opwise - Future Task	Temporary	Run task once in the future	Opwise - Demo		✗

The following table provides column descriptions for the default column display.

For information about customizing this list, including filtering, sorting, searching, and other list features, see [Using Lists](#).

Column	Description
Trigger Name	Required. Name used within Opswise to identify this trigger. It can contain a maximum of 40 alphanumeric characters. It is the user's responsibility to develop a workable naming scheme for triggers.
Type	User-defined. The type of trigger. Options: <ul style="list-style-type: none"> • CRON • Time • Temporary • Manual • File • Task Monitor • Application Monitor
Description	User-defined. Copied from the Description field in the trigger.
Task(s)	Required. Name of the task(s) being triggered when this trigger is satisfied. When selecting tasks from the definition screen, click on the lock icon to unlock the field and select tasks .
Next Scheduled Time	Supplied by Opswise. For time-based triggers, the next date and time this trigger will be satisfied. See Displaying Trigger Forecast Information .
Enabled	User-defined. Whether or not the Enabled field is checked. The user enables and disables the trigger by clicking the Enable/Disable Trigger buttons. Only enabled triggers are processed by Opswise.
Status	For the File Monitor Triggers List screen only: Task instance status of the File Monitor task instance running on behalf of an enabled File Monitor trigger. This column is blank for a File Monitor trigger that is disabled.

Creating a Trigger

You can create a trigger either of two ways.

Step 1	From the navigation pane, select Automation Center > Trigger .
Step 2	Select a trigger type.

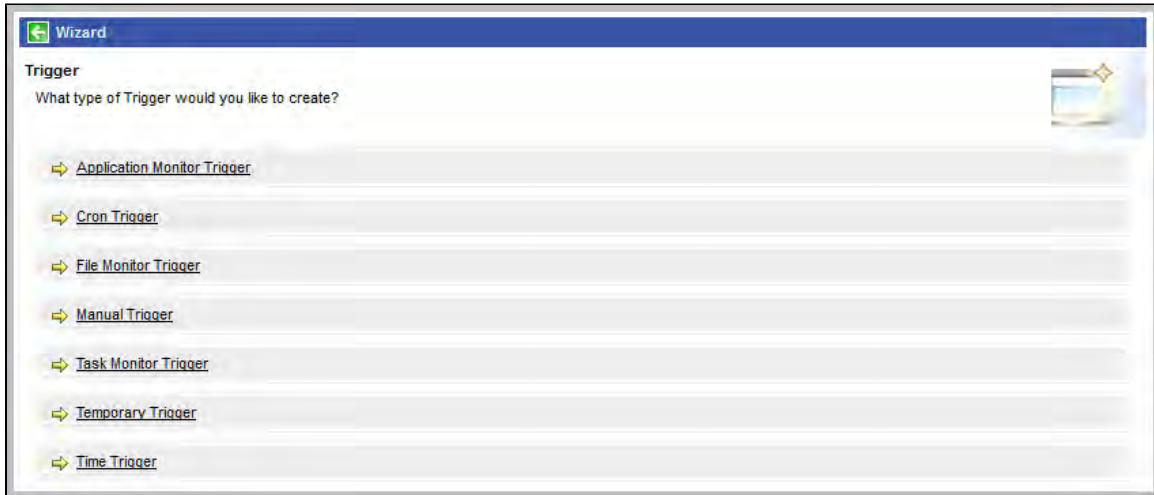
Step 3	When the Triggers List screen for that trigger type displays, click New . The Trigger Definition screen for that trigger type displays.
---------------	--

OR

Step 1	From the navigation pane, select Automation Center > Trigger .
---------------	--

Step 2	Select All Triggers .
---------------	------------------------------

Step 3	When the All Triggers List screen displays, click New . The Trigger Wizard screen displays.
---------------	--



Step 4	Click the trigger type for trigger you want to create. The Trigger List screen for that trigger type displays.
---------------	--

For detailed information on creating triggers for specific trigger types, click that trigger type in the navigation panel on the left-hand side of this page.

Daylight Saving Time

For **Cron** and **Time** triggers, Opwise handles the switch to and from Daylight Saving Time as described below.

How the time change is handled differs between **interval-based times** (such as "every 15 minutes") and **absolute times** (such as "2:30 a.m.>").

Interval-Based Times

For interval-based time Cron and Time triggers, the behavior is the same.

When Time Moves Forward

An interval-based time Cron or Time trigger defined to run at a time that is being skipped due to the time change will also be "skipped," as shown in the following example. In the example, the timezone is Eastern (EST) and the time changes from 2 a.m. EST to 3 a.m. on March 11. In this case, the 15 minute interval trigger will run at the following times:

```

Sunday, March 11, 2012 00:00:00 EST -0500
Sunday, March 11, 2012 00:15:00 EST -0500
Sunday, March 11, 2012 00:30:00 EST -0500
Sunday, March 11, 2012 00:45:00 EST -0500
Sunday, March 11, 2012 01:00:00 EST -0500
Sunday, March 11, 2012 01:15:00 EST -0500
Sunday, March 11, 2012 01:30:00 EST -0500
Sunday, March 11, 2012 01:45:00 EST -0500
Sunday, March 11, 2012 03:00:00 EDT -0400
Sunday, March 11, 2012 03:15:00 EDT -0400
Sunday, March 11, 2012 03:30:00 EDT -0400
Sunday, March 11, 2012 03:45:00 EDT -0400

```

When Time Moves Back

A Time or Cron trigger defined to run at a time that is being repeated due to the time change will also be repeated, as shown in the example below. In the example, the timezone is Eastern (EST) and the time changes from 2 a.m. EST to 1 a.m. on November 4. In this case, the 15 minute interval trigger will run at the following times:

```

Sunday, November 04, 2012 00:00:00 EDT -0400
Sunday, November 04, 2012 00:15:00 EDT -0400
Sunday, November 04, 2012 00:30:00 EDT -0400
Sunday, November 04, 2012 00:45:00 EDT -0400
Sunday, November 04, 2012 01:00:00 EDT -0400
Sunday, November 04, 2012 01:15:00 EDT -0400
Sunday, November 04, 2012 01:30:00 EDT -0400
Sunday, November 04, 2012 01:45:00 EDT -0400
Sunday, November 04, 2012 01:00:00 EST -0500
Sunday, November 04, 2012 01:15:00 EST -0500
Sunday, November 04, 2012 01:30:00 EST -0500
Sunday, November 04, 2012 01:45:00 EST -0500
Sunday, November 04, 2012 02:00:00 EST -0500
Sunday, November 04, 2012 02:15:00 EST -0500
Sunday, November 04, 2012 02:30:00 EST -0500
Sunday, November 04, 2012 02:45:00 EST -0500
Sunday, November 04, 2012 03:00:00 EST -0500
Sunday, November 04, 2012 03:15:00 EST -0500
Sunday, November 04, 2012 03:30:00 EST -0500
Sunday, November 04, 2012 03:45:00 EST -0500

```

Absolute Times

For absolute time Cron and Time triggers, the behavior is different.

Cron Trigger

Basic Behavior

The behavior of the Cron trigger follows the standard Cron behavior as described in the man page for Cron.

Each line has five time and date fields, followed by a user name if this is the system `crontab` file, followed by a command. Commands are executed by `cron(8)` when the minute, hour, and month of year fields match the current time, and at least one of the two day fields (day of month, or day of week) match the current time. This means that non-existent times, such as "missing hours" during daylight saving conversion, will never match, causing jobs scheduled during the "missing times" not to be run. Similarly, times that occur more than once (again, during daylight savings conversion) will cause matching jobs to be run twice.

When Time Moves Forward

A Cron trigger defined to run at a time that is being skipped due to the time change will also be skipped. For example: A trigger is defined for every Sunday at 2:30 a.m. On March 11, 2012, the time changes from 2 a.m. EST to 3 a.m. On March 11, the 2:30 a.m. run is skipped and runs the following Sunday at 2:30 a.m.

When Time Moves Back

A Cron trigger defined to run at a time that is being repeated due to the time change will also be repeated. For example: A trigger is defined for every Sunday at 1:30 a.m. On November 4, 2012, the time changes from 2 a.m. EDT to 1 a.m. On November 4, the 1:30 a.m. run is repeated, as shown below:

```
Sunday, November 04, 2012 01:30:00 EDT -0400  
Sunday, November 04, 2012 01:30:00 EST -0500
```

Time Trigger

When Time Moves Forward

A Time trigger defined to run at a time that is being skipped due to the time change will run as though the time did not change; however, the recorded run time will be one hour later. For example: A trigger is defined for every Sunday at 2:30 a.m. On March 11, 2012, the time changes from 2 a.m. EST to 3 a.m. EST. On March 11, the 2:30 a.m. run fires at 3:30. The following Sunday and henceforth it runs at 2:30 a.m., as shown:

```
Sunday, March 04, 2012 02:30:00 EST -0500  
Sunday, March 11, 2012 03:30:00 EDT -0400  
Sunday, March 18, 2012 02:30:00 EDT -0400  
Sunday, March 25, 2012 02:30:00 EDT -0400
```

When Time Moves Back

A Time trigger defined to run at a time that is being repeated due to the time change will not be repeated. For example: A trigger is defined for every Sunday at 1:30 a.m. On November 4, 2012, the time changes from 2 a.m. EDT to 1 a.m. EST. On November 4, the 1:30 a.m. run fires once, as shown below:

```
Sunday, October 28, 2012 01:30:00 EDT -0400  
Sunday, November 04, 2012 01:30:00 EST -0500  
Sunday, November 11, 2012 01:30:00 EST -0500  
Sunday, November 18, 2012 01:30:00 EST -0500  
Sunday, November 25, 2012 01:30:00 EST -0500
```

Cron Trigger

- Overview
- Cron Syntax
 - Cron Fields
 - Cron Special Characters
 - Cron Criteria Examples
- Creating a New Cron Trigger
- Cron Trigger Field Descriptions
- Scheduling a Time Interval
- Generating a List of Qualifying Times

Overview

The Cron trigger, similar to the [Time trigger](#), allows you to specify dates and times at which a task will be triggered.

With both Cron and Time triggers, you can define:

- Simple date and times, such as "every weekday at 12:00 a.m."
- Specific dates and times, such as "March 15 at 12:00 a.m."
- A series of dates and times, such as "every Friday at every hour."
- A mixture of specific dates/times and a series, such as "every Monday at 9 a.m."
- Complex dates and times, such as "every 3 hours between 8 a.m. and 5 p.m. on the last business day of the year."

(Read [Daylight Saving Time](#) for details about how Opwise handles Daylight Saving Time.)

It is recommended that you use a Cron trigger, rather than a Time trigger, if you want to schedule non-standard time intervals for a triggering a task (see [Scheduling a Time Interval](#), below)

Cron Syntax

The Opwise Cron trigger uses standard Cron syntax. Once the Cron trigger is entered into the system, Opwise interprets it and processes it as it would any other trigger. The trigger is satisfied when the current date and time match all the values specified in the Minutes, Hours, Day of Month, Month, and Day of Week fields.

Cron Fields

The following table identifies the allowed values for the time and date fields that are used to specify the Cron Criteria on the Cron Trigger screen.

Field Name	Required	Allowed Values	Allowed Special Characters
Minutes	Yes	0-59	* / , -
Hours	Yes	0-23	* / , -
Day of Month	Yes	1-31	* / , -
Month	Yes	1-12 or JAN-DEC	* / , -
Day of Week	Yes	0-6 or SUN-SAT	* / , -

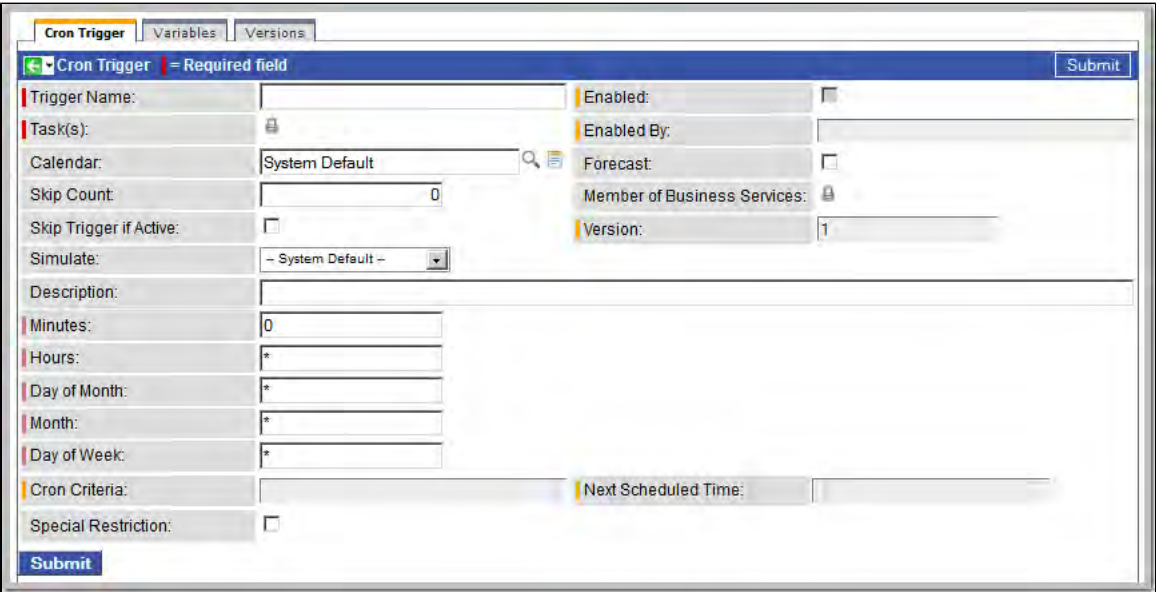
Cron Special Characters

Asterisk (*)	An asterisk indicates that the expression matches for all values of the field. For example, using * in the Month field indicates every month.
Slash (/)	A slash describes an increment of ranges. For example, 5-50/15 in the Minutes field indicate the fifth minute of the hour and every 15 minutes thereafter until the 50th minute (5,20,35,50).
Hyphen (-)	Defines a range of numbers, which are two numbers separated by a hyphen. The specified range is inclusive. For example, 9-17 in the Hours field means from 9 a.m. to 5 p.m., inclusive.
Comma (,)	Separates items in a list. A list is a set of numbers or ranges separated by commas. For example, 1,5-9,18-20 in the Hours field indicate the following hours 1,5,6,7,8,9,18,19,20.

Cron Criteria Examples

Cron Criteria	Description
0 3 30 4,6,9,11 5	At 3 a.m. on the 30th of the month, for months with exactly 30 days, if the 30th is a Friday.
0 3 31 * 0	At 3 a.m. on the 31st of the month if the 31st is a Sunday.
0 3 22-28 * 0	At 3 a.m. on the 4th Sunday of every month.
0 5-19/7 * * *	Every 7 hours between 5 a.m. and 7 p.m., daily.
0 5,12,19 * * 1,3	Every 7 hours between 5 a.m. to 7 p.m. on Monday and Wednesday.
0 9-17 * * Mon-Fri	Every hour between 9 a.m. and 5 p.m. from Monday to Friday.
0 2-11/3 * * *	Every 3 hours between 2 a.m. and 11 a.m., daily.
0 3 29 2 *	At 3 a.m. on February 29th.
30 1-3,17 * * 1,3,5	At 30 minutes past the hours of 1 a.m., 2 a.m., 3 a.m., and 5 p.m. on Monday, Wednesday, and Friday.

Creating a New Cron Trigger

Step 1	From the navigation pane, select Automation Center > Triggers > Cron Triggers . The Cron Triggers List displays.
Step 2	Click New . The Cron Trigger Definition screen displays.
	
Step 3	Using the field descriptions provided below as a guide, complete the fields as needed.
Step 4	Click the Submit button to save the record and return to the menu, or right-click the title bar and select Save to save the record and remain on the current display.
Step 5	If appropriate, repeat these steps for any additional triggers you want to add.
Step 6	Enable the trigger(s).

Cron Trigger Field Descriptions

Field Name	Description
Trigger Name	Required. Name used within Opswise to identify this trigger. It can contain a maximum of 40 alphanumeric characters. It is the user's responsibility to develop a workable naming scheme for triggers.

Enabled	User-defined. Whether or not the Enabled field is checked. The user enables and disables the trigger by clicking the Enable/Disable Trigger buttons. Only enabled triggers are processed by Opwise.
Task(s)	Required. Name of the task(s) being triggered when this trigger is satisfied. When selecting tasks from the definition screen, click on the lock icon to unlock the field and select tasks .
Enabled By	System-supplied. Displays the ID of the user who most recently enabled this trigger.
Forecast	Enabled or disabled by user. If enabled, Opwise calculates the date and time when this trigger will be satisfied for the next number days, as specified in the Forecast Period In Days Opwise system property. Opwise writes the forecasting entries to the Triggers > Forecasts display. For details, see Displaying Trigger Forecast Information .
Member of Business Services	User-defined. Allows you to select one or more Business Services that this record definition belongs to. Click the lock icon to unlock the field and select Business Services .
Calendar	If Special Restriction is selected, the calendar defines the Holidays or Non Business days. Enter a calendar name or click the magnifying glass icon either to browse for an existing calendar or to add a new calendar. To display details about the calendar specified in this field, hover over the paper icon.
Skip Count	User-defined. Allows you to specify that Opwise should skip the next <i>N</i> times this task is triggered.
Skip Trigger if Active	User-defined. Allows you to specify that Opwise should skip the next run of the specified task(s) if the previous run has not gone to a Complete status (that is, it is still active).
Version	System-supplied. The version number of the current record, which is incremented by the system every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Simulate	Optional. Enables the override of the Enable Trigger Simulation Opwise system property specification for whether or not to simulate the launching of tasks when triggers are eligible to fire. (If simulation is enabled, only the scheduled launch of the task by the trigger is inhibited.) Options: <ul style="list-style-type: none"> • -- System Default -- - Use the system default for enabling / disabling trigger simulation as specified by Enable Trigger Simulation. • True - Enable trigger simulation • False - Disable trigger simulation.
Description	User-defined. Copied from the Description field in the trigger.
Minutes	Required. Time in minutes, using standard Cron syntax.
Hours	Required. Time in hours, using standard Cron syntax. (See also Daylight Saving Time .)
Day of Month	Required. Day of the month, using standard Cron syntax.
Month	Required. Month, using standard Cron syntax.

Day of Week	Required. Day of the week, using standard Cron syntax.
Cron Criteria	System-supplied. Provides a summary of the Cron specifications. Displays in the Cron Criteria field on the Cron Triggers list.
Next Scheduled Time	Supplied by Opwise. For time-based triggers, the next date and time this trigger will be satisfied. See Displaying Trigger Forecast Information .
Special Restriction	Enable this field in order to specify additional parameters that tell Opwise how to handle exceptions such as when the trigger is satisfied on a holiday or non-business day. You can specify Simple and/or Complex Restrictions (see field descriptions below for details). For example, you can specify a Simple Restriction that disables the trigger if it is satisfied on a holiday identified in the calendar and/or a Complex Restriction that disables the trigger on the last business day of every month.
Simple Restriction	If enabled, allows you to specify an action (see Action field, below) such as Do Not Trigger on a non-business day or holiday (see Situation field, below). For example, do not trigger on a non business day.
Situation	If Simple Restriction is enabled, allows you to select the situation that causes the system to initiate the action specified in the Action field (see Action field below). Options: <ul style="list-style-type: none"> • On Non Business Day • On Holiday
Action	If Special Restriction is enabled, allows you to select an action to take on a non business day or holiday (see Situation field above). Options: <ul style="list-style-type: none"> • Do Not Trigger • Next Day (run on the next day) • Next Business Day (run on the next business day, as defined in the calendar) • Previous Day (run on the previous day) • Previous Business Day (run on the previous business day, as defined in the calendar)
Complex Restriction	If enabled, allows you to specify a set of parameters that determine one or more situations when this trigger should not be satisfied. Used in conjunction with the following fields: Restriction Mode, Restriction Adjective, Restriction Noun, Restriction Qualifier (see details below). For example, you may specify that you do not want to satisfy this trigger on the last business day of the year or the first day of each month.
Restriction Mode	If both Simple and Complex Restriction are enabled, specifies whether you want to use both restriction types (AND) or one or the other (OR). Options: <ul style="list-style-type: none"> • And • Or

Restriction Adjective	<p>If Complex Restriction is enabled, the type of selection.</p> <p>Options:</p> <ul style="list-style-type: none"> • 1st • 2nd • 3rd • 4th • Last <p>Example: The last business day of the month.</p>
Restriction Noun	<p>If Complex Restriction is enabled, the day you want to select.</p> <p>Options:</p> <ul style="list-style-type: none"> • Sunday through Saturday • Day • Business Day • Custom day (see Creating Custom Days) <p>Example: The last business day of the month.</p>
Restriction Qualifier	<p>If Complex Restriction is enabled, the period you are selecting from.</p> <p>Options:</p> <ul style="list-style-type: none"> • Month • Year • January through December • Custom period (see Creating Custom Days) <p>Example: The last quarter of the year.</p>
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
List Qualifying Times button	Opens a new browser tab and displays a list of the next 20 dates and times this trigger will be satisfied.
Enable Trigger button	Activates this trigger and writes your UserID to the Enabled By field.
Disable Trigger button	Deactivates this trigger.
Trigger Now button	Immediately triggers all the tasks specified in this trigger.
Delete button	Deletes the current record.
Variables tab	Displays all variables associated with this record.
Versions tab	Stores copies of all previous versions of the current record. See Record Versioning .

Scheduling a Time Interval

A Cron trigger lets you schedule a time interval for how often a task will be triggered. You also can select a time frame that restricts the time during which the trigger is active, so that a task will be triggered only at the time intervals within that time frame.

The time interval for a Cron trigger resets at the end of the day; when the 24-hour clock expires, the time interval count begins again at 12 a.m. on the next specified day. Therefore, if you select a time interval for multiple days, the task will be triggered at the same times each day.

Although you also can use a [Time trigger](#) to schedule a time interval for a task to be triggered on multiple days, you should use a Cron trigger if the time interval is not one by which the 24-hour clock is even divisible (2, 3, 4, 6, 8, and 12), such as in the [Cron criteria examples](#), above. Using a Time trigger to schedule this type of time interval could produce unexpected results, since the time interval for a Time trigger does not reset at the end of the day. It continues into the next day, regardless of the 24-hour clock (see [Scheduling a Time Interval with a Time Trigger](#)).

Conversely, if you want to trigger a task on multiple days at a time interval without regard to the time of day, and the interval is not one by which the 24-hour clock is even divisible, you must use a Time trigger, which will not reset at the end of the day. For example, if you want to trigger a task every 5 hours, from Monday to Friday, without regard to the time of day, a Time trigger will allow you to trigger the task on Monday at 12 a.m., 5 a.m., 10 a.m., 3 p.m., 8 p.m. and then next (5 hours later) on Tuesday at 1 a.m.. This time interval scheduling cannot be accomplished with a Cron trigger.

Generating a List of Qualifying Times

Opwise allows you to generate a list of future dates and times that this trigger will trigger the specified task. Click the **List Qualifying Times** button. Opwise opens a new browser tab and displays a list of the next 20 dates and times.

Time Trigger

- Overview
- Creating a New Time Trigger
- Time Trigger Field Descriptions
- Scheduling a Time Interval
 - Restrict Times
- Generating a List of Qualifying Times

Overview

The Time trigger, similar to the [Cron trigger](#), allows you to specify dates and times at which a task will be triggered.

With both Time and Cron triggers, you can define:

- Simple date and times, such as "every weekday at 12:00 a.m."
- Specific dates and times, such as "March 15 at 12:00 a.m."
- A series of dates and times, such as "every Friday at every hour."
- A mixture of specific dates/times and a series, such as "every Monday at 9 a.m."
- Complex dates and times, such as "every 3 hours between 8 a.m. and 5 p.m. on the last business day of the year."

(Read [Daylight Saving Time](#) for details about how Opswise handles Daylight Saving Time.)


If you want to schedule time intervals for triggering a task on multiple days, use the trigger type (Time or a Cron) that allows you to most accurately select the scheduling parameters (see [Scheduling a Time Interval](#), below).

Creating a New Time Trigger

Step 1	From the navigation pane, select Automation Center > Triggers > Time Triggers . The Time Triggers List screen displays.
Step 2	Click New . The Time Trigger Definition screen displays.
Step 3	Using the field descriptions provided below as a guide, complete the fields as needed.
Step 4	Click the Submit button to save the record and return to the menu, or, right-click the title bar and select Save to save the record and remain on the current display.
Step 5	If appropriate, repeat these steps for any additional triggers you want to add.
Step 6	Enable the trigger(s).

Time Trigger Field Descriptions

Field Name	Description
Trigger Name	Required. Name used within Opwise to identify this trigger. It can contain a maximum of 40 alphanumeric characters. It is the user's responsibility to develop a workable naming scheme for triggers.
Enabled	User-defined. Whether or not the Enabled field is checked. The user enables and disables the trigger by clicking the Enable/Disable Trigger buttons. Only enabled triggers are processed by Opwise.
Task(s)	Required. Name of the task(s) being triggered when this trigger is satisfied. When selecting tasks from the definition screen, click on the lock icon to unlock the field and select tasks .
Enabled By	System-supplied. Displays the ID of the user who most recently enabled this trigger.
Calendar	If Special Restriction is selected, the calendar defines the Holidays or Non Business days. Enter a calendar name or click the magnifying glass icon either to browse for an existing calendar or to add a new calendar. To display details about the calendar specified in this field, hover over the paper icon.
Forecast	Enabled or disabled by user. If enabled, Opwise calculates the date and time when this trigger will be satisfied for the next number days, as specified in the Forecast Period In Days Opwise system property. Opwise writes the forecasting entries to the Triggers > Forecasts display. For details, see Displaying Trigger Forecast Information .
Skip Count	User-defined. Allows you to specify that Opwise should skip the next <i>N</i> times this task is triggered.
Member of Business Services	User-defined. Allows you to select one or more Business Services that this record definition belongs to. Click the lock icon to unlock the field and select Business Services .
Skip Trigger if Active	User-defined. Allows you to specify that Opwise should skip the next run of the specified task(s) if the previous run has not gone to a Complete status (that is, it is still active).
Version	System-supplied. The version number of the current record, which is incremented by the system every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Simulate	Optional. Enables the override of the Enable Trigger Simulation Opwise system property specification for whether or not to simulate the launching of tasks when triggers are eligible to fire. (If simulation is enabled, only the scheduled launch of the task by the trigger is inhibited.) Options: <ul style="list-style-type: none"> • -- System Default -- - Use the system default for enabling / disabling trigger simulation as specified by Enable Trigger Simulation. • True - Enable trigger simulation • False - Disable trigger simulation.
Description	User-defined. Copied from the Description field in the trigger.
Time Zone	User-defined. Allows you to specify the timezone that will be applied to the time(s) specified in the trigger. For example, if you specify 23:00 and a time zone of Canada/Central, the task will run at 11:00 p.m. Central Canada time.

Time Style	<p>Specifies whether this trigger is a specific time or a series of times. Options:</p> <ul style="list-style-type: none"> • Time - Triggers the task at a specific time. Required field: Time. • Time Interval - Triggers the task at specific intervals of times. Required fields: Time Interval, Time Interval Units. Optional fields: Enable Offset, Restrict Times.
Time	<p>Required if Time Style = Time. Specifies the time of the trigger in hours, minutes, and seconds, using 24-hour time. For example, 01:45:00 means trigger the task at 1:45 a.m.; 13:45:00 means trigger the task at 1:45 p.m. (See also Daylight Saving Time.)</p>
Time Interval	<p>Required if Time Style = Time Interval. Specify a number indicating the number of Time Interval Units (see below). For example, for a Time Interval of every three hours, specify 3 in the Time Interval Field and select Hours in the Time Interval Units field.</p> <div style="background-color: #ffffcc; padding: 10px; margin: 10px 0;"> <p> Note See Scheduling a Time Interval, below, for information on using a Time Trigger to schedule a time interval for triggering a task.</p> </div>
Time Interval Units	<p>Required if Time Style = Time Interval. Select a type of time interval. Used in conjunction with the Time Interval field (see above). For example, for a Time Interval of every three hours, specify 3 in the Time Interval Field and select Hours in the Time Interval Units field. Options are:</p> <ul style="list-style-type: none"> • Minutes • Hours
Enable Offset	<p>If Time Style = Time Interval, select this field if you want define (in the Initial Time Offset field) a starting time, in minutes offset from the hour, for the trigger to run.</p>
Initial Time Offset (hh:mm)	<p>If Enable Offset is selected, use this field to define a starting time, in minutes offset from the hour.</p> <p>The default value (*) lets you select a starting hour (0 to 23) other than the next hour.</p> <p>For example:</p> <ul style="list-style-type: none"> • If you want the task to run every 30 minutes at the :15 and :45 minute mark, you would select Time Interval: 30, Time Interval Units: minutes, and Initial Time Offset: *:15. • If you want the task to run every 30 minutes at the :15 and :45 minute mark starting at 6:15 p.m., you would select Time Interval: 30, Time Interval Units: minutes, and Initial Time Offset: 18:15.
Restrict Times	<p>If Time Style = Time Interval, you can specify a period during which the trigger is active. Enable the Restrict Times field and specify the start and end times in the Enabled Start and Enabled End fields (see below).</p>
Enabled Start	<p>If Restrict Times is enabled, specify the start time of the period during which the trigger should be active. Use 24-hour time.</p>
Enabled End	<p>If Restrict Times is enabled, specify the end time of the period during which the trigger should be active. Use 24-hour time.</p>

Day Style	<p>Allows you to indicate when this trigger will be run:</p> <p>Options:</p> <ul style="list-style-type: none"> • Simple - The trigger is run every day, on business days, or on one or more specific days, depending on what you select in the Daily, Business Days, and Custom Days fields (see below). • Complex - The trigger is run on one or more days selected by a formula specified using the Date Adjective, Date Noun, and Date Qualifier fields (see below). • Every - The trigger is run at an interval of a specified number of days (see Day Interval, below) starting on a specified date (see Interval Start, below).
Daily	<p>If Date Style = Simple, you can select this field to specify that the trigger is active every day of the week.</p>
Business Days	<p>If Date Style = Simple, you can select this field to specify that the trigger is active on the business days specified in the calendar selected in the trigger's Calendar field.</p>
Date Adjective	<p>If Day Style = Complex, you can use this field to specify which in a series of days you want to select. Used in conjunction with the Date Noun and the Date Qualifier fields.</p> <p>For example, to specify "the 15th business day of the month," select Date Adjective: Nth, Date Noun: Business Day, Date Qualifier: Month.</p> <p>Options:</p> <ul style="list-style-type: none"> • Every • 1st • 2nd • 3rd • 4th • Nth • Last
Nth Amount	<p>If Day Adjective = Nth, use this field to specify the value of N.</p>
Date Noun	<p>If Day Style is Complex, you can use this field to specify the type of day you want to select. Used in conjunction with the Date Adjective and the Date Qualifier fields.</p> <p>For example, to specify "the 15th business day of the month," select Date Adjective: Nth, Date Noun: Business Day, Date Qualifier: Month.</p> <p>This drop-down menu is populated as follows:</p> <ul style="list-style-type: none"> • Sunday through Saturday • Day = any day • Business Day = The business days specified in the calendar selected in the trigger's Calendar field. • Any Custom Days specified in the calendar selected in the trigger's Calendar field.
Date Qualifier	<p>If Day Style is Complex, you can use this field to specify the period for your selection formula. Used in conjunction with the Date Noun and the Date Adjective fields.</p> <p>For example, to specify "the 15th business day of the month," select Date Adjective: Nth, Date Noun: Business Day, Date Qualifier: Month.</p> <p>Options:</p> <ul style="list-style-type: none"> • Month • Year • January through December • Custom Period (see Creating Custom Days)

Date Adjustment	<p>If Day Style is Complex, you can use this field to adjust your date setting by a less or plus number of Days or Business Days.</p> <p>For example, to specify the 2nd to last day of the month (last day of the month less one day), select Date Adjective: Last, Date Noun: Day, Date Qualifier: Month, Date Adjustment: Less, Adjustment Amount: 1, Adjustment Type: Day.</p> <p>Options:</p> <ul style="list-style-type: none"> • None • Less • Plus <p>Default is None.</p>
Adjustment Amount	<p>If Day Adjustment = Less or More, use this field to specify the number of Days or Business Days to adjust your date setting.</p>
Adjustment Type	<p>If Day Adjustment = Less or More, use this field to specify the type of day by which to adjust your date setting.</p> <p>Options:</p> <ul style="list-style-type: none"> • Day • Business Day
Day Interval	<p>If Day Style = Every, use this field to specify the interval (in days) at which this trigger will run.</p>
Interval Start	<p>If Day Style = Every, use this field to specify the first day of the interval on which this trigger will run.</p>
Special Restriction	<p>Enable this field in order to specify additional parameters that tell Opwise how to handle exceptions such as when the trigger is satisfied on a holiday or non-business day. You can specify Simple and/or Complex Restrictions (see field descriptions below for details). For example, you can specify a Simple Restriction that disables the trigger if it is satisfied on a holiday identified in the calendar and/or a Complex Restriction that disables the trigger on the last business day of every month.</p>
Simple Restriction	<p>If enabled, allows you to specify an action (see Action field, below) such as Do Not Trigger on a non-business day or holiday (see Situation field, below). For example, do not trigger on a non business day.</p>
Situation	<p>If Simple Restriction is enabled, allows you to select the situation that causes the system to initiate the action specified in the Action field (see Action field below).</p> <p>Options:</p> <ul style="list-style-type: none"> • On Non Business Day • On Holiday
Action	<p>If Special Restriction is enabled, allows you to select an action to take on a non business day or holiday (see Situation field above).</p> <p>Options:</p> <ul style="list-style-type: none"> • Do Not Trigger • Next Day (run on the next day) • Next Business Day (run on the next business day, as defined in the calendar) • Previous Day (run on the previous day) • Previous Business Day (run on the previous business day, as defined in the calendar)

Complex Restriction	If enabled, allows you to specify a set of parameters that determine one or more situations when this trigger should not be satisfied. Used in conjunction with the following fields: Restriction Mode, Restriction Adjective, Restriction Noun, Restriction Qualifier (see details below). For example, you may specify that you do not want to satisfy this trigger on the last business day of the year or the first day of each month.
Restriction Mode	If both Simple and Complex Restriction are enabled, specifies whether you want to use both restriction types (AND) or one or the other (OR). Options: <ul style="list-style-type: none">• And• Or
Restriction Adjective	If Complex Restriction is enabled, the type of selection. Options: <ul style="list-style-type: none">• 1st• 2nd• 3rd• 4th• Last Example: The last business day of the month.
Restriction Noun	If Complex Restriction is enabled, the day you want to select. Options: <ul style="list-style-type: none">• Sunday through Saturday• Day• Business Day• Custom day (see Creating Custom Days) Example: The last business day of the month.
Restriction Qualifier	If Complex Restriction is enabled, the period you are selecting from. Options: <ul style="list-style-type: none">• Month• Year• January through December• Custom period (see Creating Custom Days) Example: The last quarter of the year .
Next Scheduled Time	Supplied by Opwise. For time-based triggers, the next date and time this trigger will be satisfied. See Displaying Trigger Forecast Information .
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
List Qualifying Times button	Opens a new browser tab and displays a list of the next 20 dates and times this trigger will be satisfied.

Enable Trigger button	Activates this trigger and writes your UserID to the Enabled By field.
Disable Trigger button	Deactivates this trigger.
Trigger Now button	Immediately triggers all the tasks specified in this trigger.
Delete button	Deletes the current record.
Variables tab	Displays all variables associated with this record.
Versions tab	Stores copies of all previous versions of the current record. See Record Versioning .

Scheduling a Time Interval

A Time trigger [time interval](#) lets you specify how often a task will be triggered, but for triggering a task on multiple days, you may not be able to specify the same time every day that the task will be triggered. This could produce unexpected results.

By default, a time interval count begins at 12 a.m.. If you schedule a time interval for a task to be triggered on multiple days, the task will be triggered at the first specified time interval, and then again whenever the time interval is reached. When the 24-hour clock expires, the time interval count does not reset to 12 a.m.; it continues into the next day. If the time interval is not one by which the 24-hour clock is even divisible (2, 3, 4, 6, 8, and 12), the task will be triggered at different times than on the first day.

For example, if you want a task to be triggered at the same time every 4 hours from Monday to Friday, a Time trigger will trigger the task on Monday at 4 a.m., 8 a.m., 12 p.m., 4 p.m., 8 p.m., and Tuesday at 12 a.m.. 4 hours later, at 4 a.m. - it will again start triggering the task every 4 hours. Since the time interval (4) divides evenly into 24, the task is triggered at the same time every day, and results will be as expected.

However, if you want a task to be triggered every 7 hours from Monday to Friday, a Time trigger will trigger the task on Monday at 7 a.m., 2 p.m., and 9 p.m., and then - 7 hours later - Tuesday at 4 a.m. The time interval "rolls over" to the next day. It does not restart at 12:00 a.m. when the 24-hour clock expires, and so the task will not be triggered at the same time every day. The same results will occur if you use [Restrict Times](#), below, for the time interval.

Therefore, for scheduling time intervals by which the 24-hour clock is not evenly divisible, it is recommended that instead you use a [Cron trigger](#).

Restrict Times

If you are triggering a task on a time interval, the [Restrict Times](#) field lets you select a time frame during which the trigger is active. The task will be triggered at the specified time interval only when the time interval occurs during the specified Restrict Times.

For example, if you want a task to be triggered every 4 hours from Monday to Friday and only between the hours of 8 a.m. (Enabled Start) and 7 p.m. (Enabled End), a Time trigger will check the time on Monday at 4 a.m., 8 a.m., 12 p.m., 4 p.m., 8 p.m., and Tuesday at 12 a.m., but only will trigger the task at 8:00 a.m., 12 p.m., and 4 p.m. (the only three 4-hour interval times between 8 a.m. and 7 p.m.). On Tuesday at 12 a.m., it will continue checking the time every 4 hours and will trigger the task at the same times it did on Monday.

However, if you want a task to be triggered every 7 hours from Monday to Friday and only between the hours of 8 a.m. and 7 p.m., a Time trigger will check the time on Monday at 12 a.m., 7 a.m., 2 p.m., and 9 p.m., but only will trigger the task at 2 p.m. (the only 7-hour interval time between 8 a.m. and 7 p.m.). On Monday at 9 p.m., it will continue checking the time every 7 hours, beginning on Tuesday at 4 a.m., and will trigger the task on Tuesday at 11 a.m. and 6 p.m. (both of which are 7-hour interval times between 8 a.m. and 7 p.m.).

Generating a List of Qualifying Times

Opswise allows you to generate a list of future dates and times that a trigger will trigger the specified task. Click the **List Qualifying Times** button. Opswise opens a new browser tab and displays a list of the next 20 dates and times.

Manual Trigger

- Overview
- Creating a New Manual Trigger and Triggering One or More Tasks
 - Entering Variables
- Manual Trigger Field Descriptions

Overview

The Manual trigger allows you to launch a task(s) immediately, while setting or overriding the value of one or more user-defined variables specified in the task. You will use this trigger if you want to manually launch a task but cannot use the Launch Task or Trigger Now buttons because you need to set or override one or more variables. For example, you might choose to use a Manual trigger to launch a "generic" workflow that you run occasionally with a slight variation in specific details. In this case, you will launch the workflow and pass in the appropriate details using variables.

You can use the Manual trigger to set values to pre-existing variables or create new variables. The variable values you enter here override all others. However, the change in value only persists while this launched task instance(s) is running. Future executions of the task(s), unless they are launched by a Manual trigger, will use the [standard methods for resolving user-defined variables](#).

The audit message created when you use Manual trigger is the same as Trigger Now.

Creating a New Manual Trigger and Triggering One or More Tasks

Step 1	From the navigation pane, select Automation Center > Triggers > Manual Triggers . The Manual Triggers List screen displays.
Step 2	Click New . The Manual Trigger Definition screen displays. <div data-bbox="224 968 1373 1276" data-label="Image"> </div>
Step 3	Using the field descriptions provided below as a guide, complete the fields as needed.
Step 4	Right-click the title bar and select Save to save the record and remain on the current display.

Entering Variables

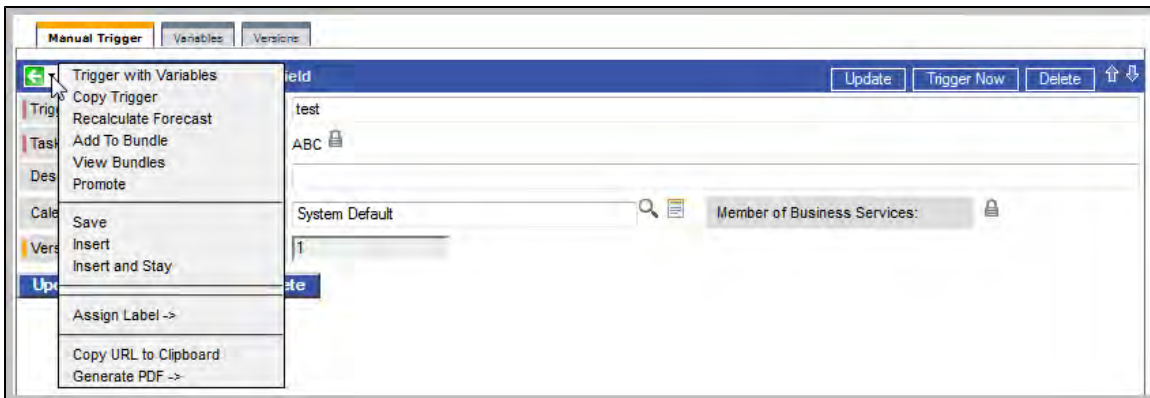
Two methods are available for entering variables:

1. Use the **Trigger With Variables** menu option.
2. Use the Variables tab and **Trigger Now** button.

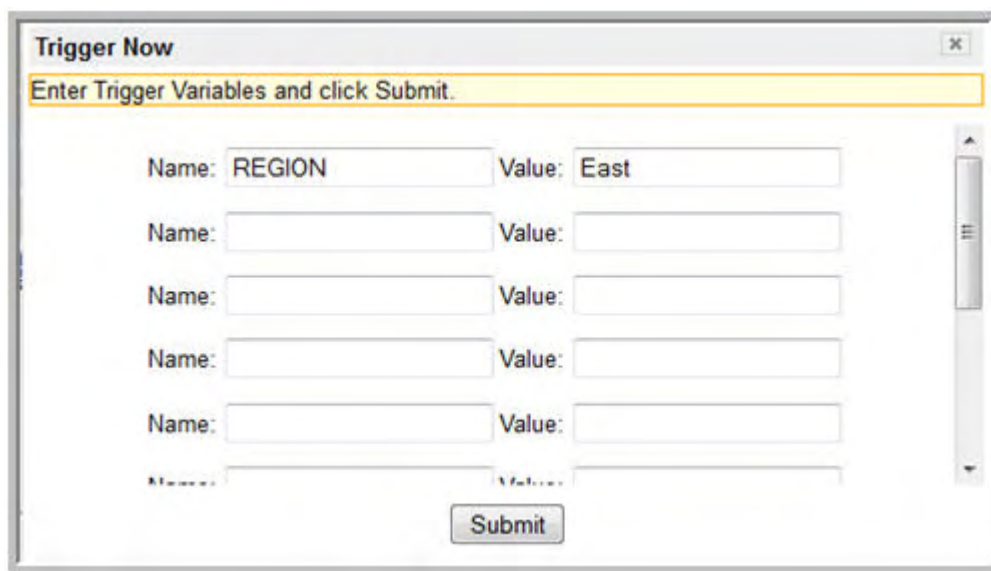
If you want to preserve information about the variables you are setting or overriding (the name and value), or if you want to specify default values, use the Variables tab. If you set up a Manual trigger with default values in the Variables tab, any values you set using the Trigger With Variables popup window override the values in the Variables tab. Each method is described below.

Using the Trigger with Variables Menu Option

Step 1 Click the down-arrow next to **Manual Trigger** to display a menu of options.



Step 2 Select **Trigger With Variables**. A pop-up window displays that allows you to set or override the values of up to ten variables that will be used in the execution of the task(s) named in the Manual trigger. These can be existing or new variables. Any existing variables are automatically populated in the window.



Step 3 When you are finished entering the variables, click **Submit** in the pop-up window to launch the tasks named in the trigger. When you click **Submit**, the variable information in this window is passed into the task instance(s) where referenced and the contents of the window are deleted.

Using the Variables Tab

Step 1 Click the Variables tab.

Step 2 Use the **New** button to add the variables you want to set.

Step 3 When you are finished, return to the main Trigger page and click **Trigger Now** to launch the tasks named in the trigger.

Manual Trigger Field Descriptions

Field Name	Description
Trigger Name	Required. Name used within Opwise to identify this trigger. It can contain a maximum of 40 alphanumeric characters. It is the user's responsibility to develop a workable naming scheme for triggers.

Task(s)	Required. Name of the task(s) being triggered when this trigger is satisfied. When selecting tasks from the definition screen, click on the lock icon to unlock the field and select tasks .
Description	User-defined. Copied from the Description field in the trigger.
Calendar	If Special Restriction is selected, the calendar defines the Holidays or Non Business days. Enter a calendar name or click the magnifying glass icon either to browse for an existing calendar or to add a new calendar. To display details about the calendar specified in this field, hover over the paper icon.
Member of Business Services	User-defined. Allows you to select one or more Business Services that this record definition belongs to. Click the lock icon to unlock the field and select Business Services .
Version	System-supplied. The version number of the current record, which is incremented by the system every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Trigger Now button	Immediately triggers all the tasks specified in this trigger.
Delete button	Deletes the current record.
Variables tab	Displays all variables associated with this record.
Versions tab	Stores copies of all previous versions of the current record. See Record Versioning .

Temporary Trigger

- Overview
- Creating a New Temporary Trigger
- Temporary Trigger Field Descriptions

Overview

The Temporary trigger allows you to set up a one-time trigger for a task, based on a single date and time. You will use this trigger if you want to set up a task to run once at some time in the future.

Creating a New Temporary Trigger

Step 1	From the navigation pane, select Automation Center > Triggers > Temporary Triggers . The Temporary Triggers List screen displays.
Step 2	Click New . The Temporary Trigger Definition screen displays. <div data-bbox="222 747 1373 1205" data-label="Image"> </div>
Step 3	Using the field descriptions provided below as a guide, complete the fields as needed.
Step 4	Click the Submit button to save the record and return to the menu, or, right-click the title bar and select Save to save the record and remain on the current display.
Step 5	If appropriate, repeat these steps for any additional triggers you want to add.
Step 6	Enable the trigger(s).

Temporary Trigger Field Descriptions

Field Name	Description
Trigger Name	Required. Name used within Opwise to identify this trigger. It can contain a maximum of 40 alphanumeric characters. It is the user's responsibility to develop a workable naming scheme for triggers.
Enabled	User-defined. Whether or not the Enabled field is checked. The user enables and disables the trigger by clicking the Enable/Disable Trigger buttons. Only enabled triggers are processed by Opwise.
Task(s)	Required. Name of the task(s) being triggered when this trigger is satisfied. When selecting tasks from the definition screen, click on the lock icon to unlock the field and select tasks .

Enabled By	System-supplied. Displays the ID of the user who most recently enabled this trigger.
Calendar	If Special Restriction is selected, the calendar defines the Holidays or Non Business days. Enter a calendar name or click the magnifying glass icon either to browse for an existing calendar or to add a new calendar. To display details about the calendar specified in this field, hover over the paper icon.
Forecast	Enabled or disabled by user. If enabled, Opwise calculates the date and time when this trigger will be satisfied for the next number days, as specified in the Forecast Period In Days Opwise system property. Opwise writes the forecasting entries to the Triggers > Forecasts display. For details, see Displaying Trigger Forecast Information .
Simulate	Optional. Enables the override of the Enable Trigger Simulation Opwise system property specification for whether or not to simulate the launching of tasks when triggers are eligible to fire. (If simulation is enabled, only the scheduled launch of the task by the trigger is inhibited.) Options: <ul style="list-style-type: none"> • -- System Default -- - Use the system default for enabling / disabling trigger simulation as specified by Enable Trigger Simulation. • True - Enable trigger simulation • False - Disable trigger simulation.
Member of Business Services	User-defined. Allows you to select one or more Business Services that this record definition belongs to. Click the lock icon to unlock the field and select Business Services .
Version	System-supplied. The version number of the current record, which is incremented by the system every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Description	User-defined. Copied from the Description field in the trigger.
Date	Date you want the trigger to be satisfied.
Time (hh.mm)	Required. Specifies the time of the trigger in hours and minutes. For example, 01:45 means trigger the task at 1:45 a.m.; 13:45 means trigger the task at 1:45 p.m.
Next Scheduled Time	Supplied by Opwise. For time-based triggers, the next date and time this trigger will be satisfied. See Displaying Trigger Forecast Information .
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Enable Trigger button	Activates this trigger and writes your UserID to the Enabled By field.
Disable Trigger button	Deactivates this trigger.
Trigger Now button	Immediately triggers all the tasks specified in this trigger.
Delete button	Deletes the current record.

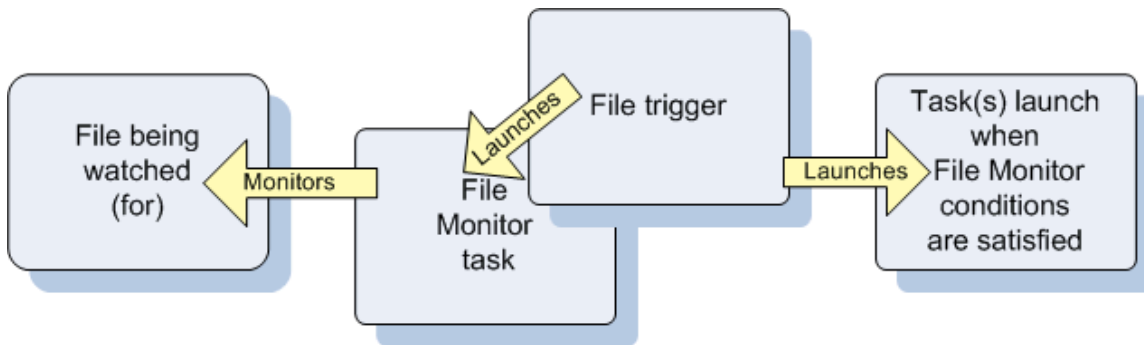
Variables tab	Displays all variables associated with this record.
Versions tab	Stores copies of all previous versions of the current record. See Record Versioning .

File Monitor Trigger

- [Overview](#)
- [Built-In Variables](#)
- [Prerequisites](#)
- [Creating a New File Monitor Trigger](#)
- [File Monitor Trigger Field Descriptions](#)

Overview

The File (Monitor) trigger allows you to trigger one or more tasks based on the creation, deletion, change, existence or non-existence of a file on a particular machine. The trigger works in conjunction with the File Monitor task, as illustrated in the image below. For a detailed description, see [Launching a File Monitor Task Using a File \(Monitor\) Trigger](#).



Built-In Variables

The built-in variables outlined below can be used to pass data where appropriate:

- [Task and Task Instance Variables](#)
- [File Monitor Variables](#).

Prerequisites

Before you can use a File Monitor Trigger, you need the following:

- A [Windows](#), [Linux/Unix](#), or [z/OS](#) agent, which will execute the File Monitor task.
- A [File Monitor task](#), which watches for the creation, deletion, change, existence, or non-existence of a file.

Creating a New File Monitor Trigger

Step 1 From the navigation pane, select **Automation Center > Triggers > File Triggers**. The File Triggers List screen displays.

Step 2 Click **New**. The File Monitor Trigger Definition screen displays.

Step 3 Using the field descriptions provided below as a guide, complete the fields as needed.

Step 4 Click the **Submit** button to save the record and return to the menu, or, right-click the title bar and select **Save** to save the record and remain on the current display.

Step 5 If appropriate, repeat these steps for any additional triggers you want to add.

Step 6 Enable the trigger(s).

File Monitor Trigger Field Descriptions

Field Name	Description
Trigger Name	Required. Name used within Opwise to identify this trigger. It can contain a maximum of 40 alphanumeric characters. It is the user's responsibility to develop a workable naming scheme for triggers.
Enabled	User-defined. Whether or not the Enabled field is checked. The user enables and disables the trigger by clicking the Enable/Disable Trigger buttons. Only enabled triggers are processed by Opwise.
File Monitor	Required. File Monitor task being executed. Enter a task name or click the magnifying glass either to browse for an existing task or add a new task. To display details about the task specified in this field, hover over the paper icon.
Enabled By	System-supplied. Displays the ID of the user who most recently enabled this trigger.
Task(s)	Required. Name of the task(s) being triggered when this trigger is satisfied. When selecting tasks from the definition screen, click on the lock icon to unlock the field and select tasks .
Member of Business Services	User-defined. Allows you to select one or more Business Services that this record definition belongs to. Click the lock icon to unlock the field and select Business Services .
Calendar	If Special Restriction is selected, the calendar defines the Holidays or Non Business days. Enter a calendar name or click the magnifying glass icon either to browse for an existing calendar or to add a new calendar. To display details about the calendar specified in this field, hover over the paper icon.

Version	System-supplied. The version number of the current record, which is incremented by the system every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Running Monitor	System-supplied. Lists File Monitor tasks currently running that were launched by this trigger.
Skip Count	User-defined. Allows you to specify that Opswise should skip the next <i>N</i> times this task is triggered.
Skip Trigger if Active	User-defined. Allows you to specify that Opswise should skip the next run of the specified task(s) if the previous run has not gone to a Complete status (that is, it is still active).
Description	User-defined. Copied from the Description field in the trigger.
Restrict Times	If Time Style = Time Interval, you can specify a period during which the trigger is active. Enable the Restrict Times field and specify the start and end times in the Enabled Start and Enabled End fields (see below).
Time Zone	User-defined. Allows you to specify the timezone that will be applied to the times specified in the Restrict Times parameters. For example, if you specify an Enabled Start of 23:00, an Enabled End of 24:00, and a time zone of Canada/Central, the trigger is enabled at 11:00 p.m. and disabled at 12:00 a.m., Central Canada time.
Enabled Start	If Restrict Times is enabled, specify the start time of the period during which the trigger should be active. Use 24-hour time.
Enabled End	If Restrict Times is enabled, specify the end time of the period during which the trigger should be active. Use 24-hour time.
Special Restriction	Enable this field in order to specify additional parameters that tell Opswise how to handle exceptions such as when the trigger is satisfied on a holiday or non-business day. You can specify Simple and/or Complex Restrictions (see field descriptions below for details). For example, you can specify a Simple Restriction that disables the trigger if it is satisfied on a holiday identified in the calendar and/or a Complex Restriction that disables the trigger on the last business day of every month.
Simple Restriction	If enabled, allows you to specify an action (see Action field, below) such as Do Not Trigger on a non-business day or holiday (see Situation field, below). For example, do not trigger on a non business day.
Situation	If Simple Restriction is enabled, allows you to select the situation that causes the system to initiate the action specified in the Action field (see Action field below). Options: <ul style="list-style-type: none"> • On Non Business Day • On Holiday
Action	If Special Restriction is enabled, allows you to select an action to take on a non business day or holiday (see Situation field above). Options: <ul style="list-style-type: none"> • Do Not Trigger • Next Day (run on the next day) • Next Business Day (run on the next business day, as defined in the calendar) • Previous Day (run on the previous day) • Previous Business Day (run on the previous business day, as defined in the calendar)

Complex Restriction	If enabled, allows you to specify a set of parameters that determine one or more situations when this trigger should not be satisfied. Used in conjunction with the following fields: Restriction Mode, Restriction Adjective, Restriction Noun, Restriction Qualifier (see details below). For example, you may specify that you do not want to satisfy this trigger on the last business day of the year or the first day of each month.
Restriction Mode	If both Simple and Complex Restriction are enabled, specifies whether you want to use both restriction types (AND) or one or the other (OR). Options: <ul style="list-style-type: none">• And• Or
Restriction Adjective	If Complex Restriction is enabled, the type of selection. Options: <ul style="list-style-type: none">• 1st• 2nd• 3rd• 4th• Last Example: The last business day of the month.
Restriction Noun	If Complex Restriction is enabled, the day you want to select. Options: <ul style="list-style-type: none">• Sunday through Saturday• Day• Business Day• Custom day (see Creating Custom Days) Example: The last business day of the month.
Restriction Qualifier	If Complex Restriction is enabled, the period you are selecting from. Options: <ul style="list-style-type: none">• Month• Year• January through December• Custom period (see Creating Custom Days) Example: The last quarter of the year .
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Enable Trigger button	Activates this trigger and writes your UserID to the Enabled By field.
Disable Trigger button	Deactivates this trigger.

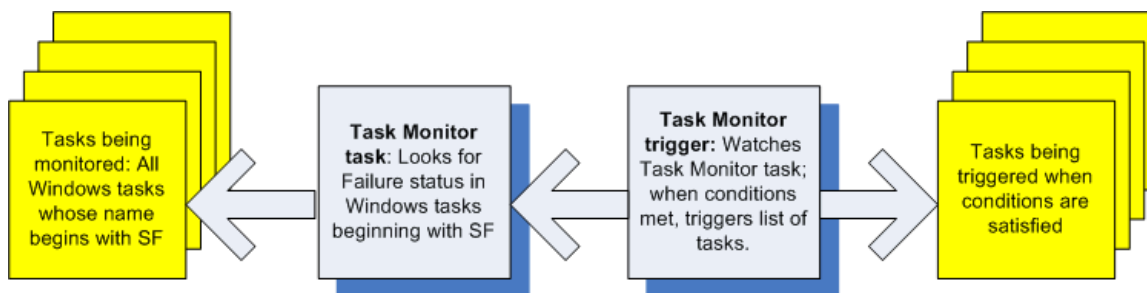
Trigger Now button	Immediately triggers all the tasks specified in this trigger.
Delete button	Deletes the current record.
Variables tab	Displays all variables associated with this record.
Versions tab	Stores copies of all previous versions of the current record. See Record Versioning .

Task Monitor Trigger

- Overview
- Built-In Variables
- Prerequisites
- Creating a New Task Monitor Trigger
- Task Monitor Trigger Field Descriptions

Overview

The Task Monitor Trigger allows you to trigger one or more tasks based on the conditions specified in an associated Task Monitor task, as illustrated in the example below. For details, see [Launching a Task Monitor Task Using a Task Monitor Trigger](#).



Built-In Variables

The built-in variables outlined below can be used to pass data where appropriate:

- Task and Task Instance Variables
- Task Monitor Variables.

Prerequisites

Before you can use a Task Monitor Trigger, you need a [Task Monitor task](#), which defines the statuses being monitored for and the tasks being monitored.

Creating a New Task Monitor Trigger

Step 1 From the navigation pane, select **Automation Center > Triggers > Task Monitor Triggers**. The Task Monitor Triggers List screen displays.

Step 2 Click **New**. The Task Monitor Trigger Definition screen displays.

Step 3	Using the field descriptions provided below as a guide, complete the fields as needed.
Step 4	Click the Submit button to save the record and return to the menu, or, right-click the title bar and select Save to save the record and remain on the current display.
Step 5	If appropriate, repeat these steps for any additional triggers you want to add.
Step 6	Enable the trigger(s). When you enable the trigger, its associated Task Monitor task launches.

Task Monitor Trigger Field Descriptions

Field Name	Description
Trigger Name	Required. Name used within Opwise to identify this trigger. It can contain a maximum of 40 alphanumeric characters. It is the user's responsibility to develop a workable naming scheme for triggers.
Enabled	User-defined. Whether or not the Enabled field is checked. The user enables and disables the trigger by clicking the Enable/Disable Trigger buttons. Only enabled triggers are processed by Opwise.
Task Monitor	Required. Task Monitor task being executed. Enter a task name or click the magnifying glass either to browse for an existing task or add a new task. To display details about the task specified in this field, hover over the paper icon.
Enabled By	System-supplied. Displays the ID of the user who most recently enabled this trigger.
Task(s)	Required. Name of the task(s) being triggered when this trigger is satisfied. When selecting tasks from the definition screen, click on the lock icon to unlock the field and select tasks .
Member of Business Services	User-defined. Allows you to select one or more Business Services that this record definition belongs to. Click the lock icon to unlock the field and select Business Services .
Calendar	If Special Restriction is selected, the calendar defines the Holidays or Non Business days. Enter a calendar name or click the magnifying glass icon either to browse for an existing calendar or to add a new calendar. To display details about the calendar specified in this field, hover over the paper icon.
Version	System-supplied. The version number of the current record, which is incremented by the system every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Skip Count	User-defined. Allows you to specify that Opwise should skip the next <i>N</i> times this task is triggered.
Skip Trigger if Active	User-defined. Allows you to specify that Opwise should skip the next run of the specified task(s) if the previous run has not gone to a Complete status (that is, it is still active).
Description	User-defined. Copied from the Description field in the trigger.
Restrict Times	Allows you to specify a period during which the trigger is active. Enable the Restrict Times field and specify the start and end times in the Enabled Start and Enabled End fields (see below).

Enabled Start	If Restrict Times is enabled, specify the start time of the period during which the trigger should be active. Use 24-hour time.
Enabled End	If Restrict Times is enabled, specify the end time of the period during which the trigger should be active. Use 24-hour time.
Timezone	Specifies the timezone for the times specified in the Restrict Time parameters.
Special Restriction	Enable this field in order to specify additional parameters that tell Opwise how to handle exceptions such as when the trigger is satisfied on a holiday or non-business day. You can specify Simple and/or Complex Restrictions (see field descriptions below for details). For example, you can specify a Simple Restriction that disables the trigger if it is satisfied on a holiday identified in the calendar and/or a Complex Restriction that disables the trigger on the last business day of every month.
Simple Restriction	If enabled, allows you to specify an action (see Action field, below) such as Do Not Trigger on a non-business day or holiday (see Situation field, below). For example, do not trigger on a non business day.
Situation	If Simple Restriction is enabled, allows you to select the situation that causes the system to initiate the action specified in the Action field (see Action field below). Options: <ul style="list-style-type: none"> • On Non Business Day • On Holiday
Action	If Special Restriction is enabled, allows you to select an action to take on a non business day or holiday (see Situation field above). Options: <ul style="list-style-type: none"> • Do Not Trigger • Next Day (run on the next day) • Next Business Day (run on the next business day, as defined in the calendar) • Previous Day (run on the previous day) • Previous Business Day (run on the previous business day, as defined in the calendar)
Complex Restriction	If enabled, allows you to specify a set of parameters that determine one or more situations when this trigger should not be satisfied. Used in conjunction with the following fields: Restriction Mode, Restriction Adjective, Restriction Noun, Restriction Qualifier (see details below). For example, you may specify that you do not want to satisfy this trigger on the last business day of the year or the first day of each month.
Restriction Mode	If both Simple and Complex Restriction are enabled, specifies whether you want to use both restriction types (AND) or one or the other (OR). Options: <ul style="list-style-type: none"> • And • Or

Restriction Adjective	<p>If Complex Restriction is enabled, the type of selection.</p> <p>Options:</p> <ul style="list-style-type: none"> • 1st • 2nd • 3rd • 4th • Last <p>Example: The last business day of the month.</p>
Restriction Noun	<p>If Complex Restriction is enabled, the day you want to select.</p> <p>Options:</p> <ul style="list-style-type: none"> • Sunday through Saturday • Day • Business Day • Custom day (see Creating Custom Days) <p>Example: The last business day of the month.</p>
Restriction Qualifier	<p>If Complex Restriction is enabled, the period you are selecting from.</p> <p>Options:</p> <ul style="list-style-type: none"> • Month • Year • January through December • Custom period (see Creating Custom Days) <p>Example: The last quarter of the year.</p>
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Enable Trigger button	Activates this trigger and writes your UserID to the Enabled By field.
Disable Trigger button	Deactivates this trigger.
Trigger Now button	Immediately triggers all the tasks specified in this trigger.
Delete button	Deletes the current record.
Variables tab	Displays all variables associated with this record.
Versions tab	Stores copies of all previous versions of the current record. See Record Versioning .

Enabling and Disabling Triggers

- Introduction
- Enabling/Disabling One or More Triggers
- Enabling/Disabling a Single Trigger
- Enabling/Disabling One or More Triggers from the Command Line

Introduction

When you define and submit a new trigger, you must enable it in order for Opwise to begin processing it. Opwise only processes triggers that are flagged as Enabled (Enabled triggers are Active triggers).

For tracking and compliance purposes, you must manually enable and disable triggers either by using:

- **Enable Trigger** and **Disable Trigger** buttons or menu options on the Trigger screen.
- `ops-trigger-enable` and `ops-trigger-disable` CLI commands.

This process saves an audit record detailing the event. The trigger record also displays the ID of the user who enabled it.



This does not apply to [Manual triggers](#).

Enabling/Disabling One or More Triggers

Step 1	Display the Triggers list or Active Triggers list.
Step 2	For each trigger you want to enable or disable, click the box in the leftmost column.
Step 3	From the Actions on selected rows... menu at the bottom of the list, select Enable Triggers or Disable Triggers as appropriate. The Enabled flag on the trigger is modified as appropriate.

Enabling/Disabling a Single Trigger

Step 1	From the triggers list, right-click the trigger you want to enable or disable. A drop-down menu displays.
Step 2	Select Enable Trigger or Disable Trigger as appropriate. The Enabled flag on the trigger is modified.

OR

Step 1	Display the trigger you want to enable or disable.
Step 2	Click the Enable Trigger or Disable Trigger button as appropriate. The Enabled flag on the trigger is modified.

Enabling/Disabling One or More Triggers from the Command Line

See the `ops-trigger-enable` and `ops-trigger-disable` CLI commands for instructions.

Copying Triggers

- Overview
 - Copying One or More Triggers on the Trigger List
 - Copying a Trigger on the Trigger Definition Form

Overview

You can make copies of all Opswise records, including triggers, using the standard methods for copying: Insert, Insert and Stay (see [Saving, Updating, Deleting, and Copying Records](#)).

However, these methods do not make copies of other records that may be attached to the trigger, such as Notifications, Actions, Variables, and so on.

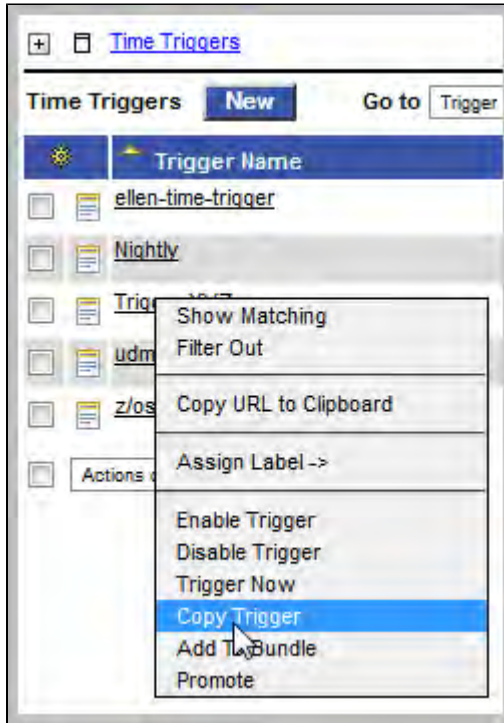
The Copy Trigger option allows you to make a complete copy of a trigger, including all of its associated records, such as variables and notes. It does not copy referenced records, such as virtual resources, but retains the relationship to these records for the copied trigger.

Copying One or More Triggers on the Trigger List

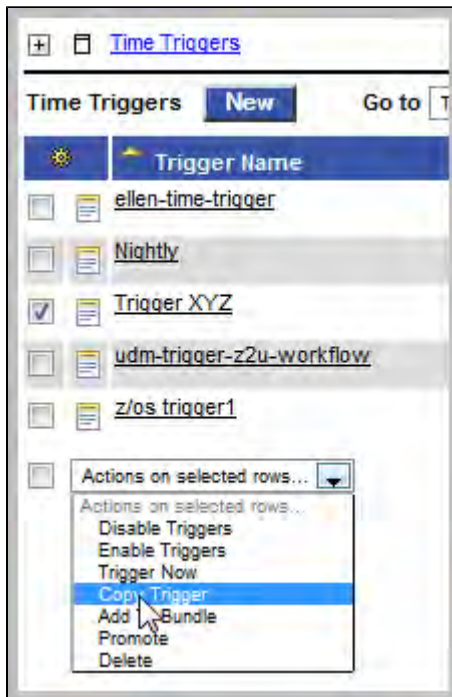
Step 1	From the navigation pane, select a trigger type from Automation Center > Triggers . The Triggers List for the selected trigger type displays.
Step 2	Locate the trigger(s) you want to copy (see Searching for Records).

Step 3 Copy the trigger(s) using either of two methods:

1. To copy a single trigger, hover over the trigger name and right-click. From the pop-up menu, select **Copy Trigger**.



2. To copy one or more triggers, click the box to the left of each trigger name. From the **Action on selected rows...** drop-down list at the bottom of the page, select **Copy Trigger**.



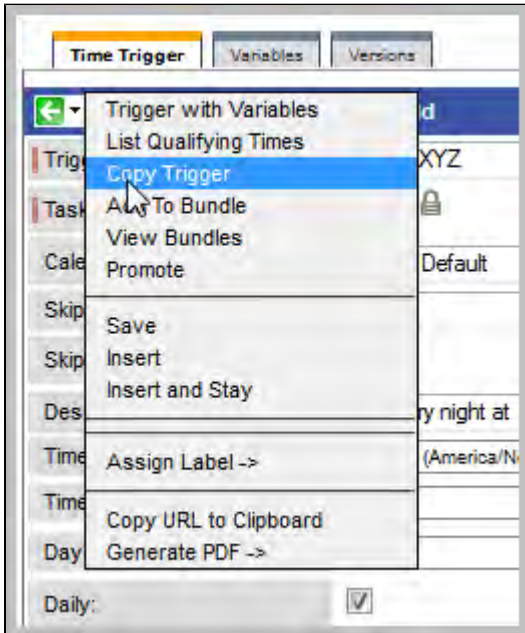
Step 4 Opwise copies the trigger(s), automatically creating the new name by prepending the original name with "Copy of" (for example, "Copy of Trigger XYZ"), and adds it to the list. If the new name already exists, the system appends a counter to the name, such as "Copy of Trigger XYZ 1", "Copy of Trigger XYZ 2", and so on, until it finds a name that is available.

Step 5 To modify the name or any other information in the trigger, open the new trigger, make your changes, and click **Update**.

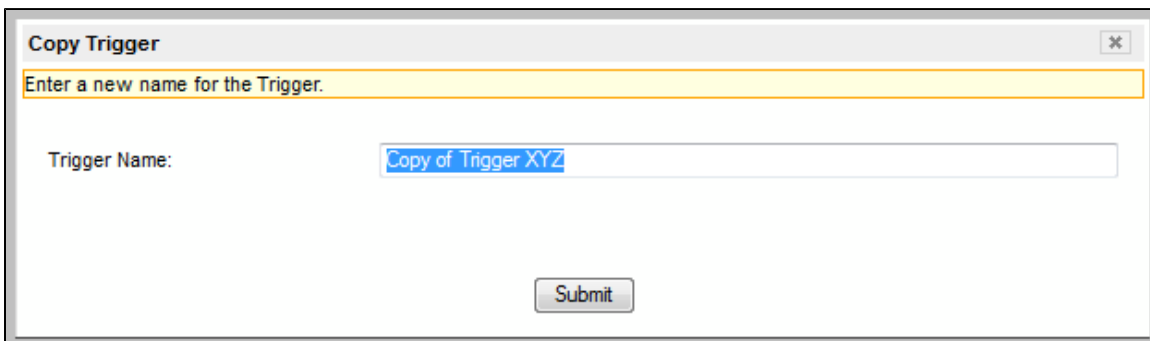
Copying a Trigger on the Trigger Definition Form

Step 1 Open the trigger you want to copy.

Step 2 Hover over the down arrow above **Trigger Name:**. A pop-up menu displays:



Step 3 Click **Copy Trigger**. Another window appears, prompting for a name for the new trigger. The default is the original trigger name, prepended with "Copy of," as shown in the following example:



Step 4 Enter a new name for the trigger and click **Submit**. The system copies the trigger and all its attachments and saves it under the new name. (If the new name already exists, the copy will fail.)

Triggering with Variables

- Overview
- Using the Trigger with Variables Pop-up Method
- Using the Variables Tab Method

Overview

Opswise provides two methods for manually launching all of the tasks associated with a trigger while supplying values for variables used by the task(s):

- Use the [Trigger with Variables pop-up](#) method if you do not want the values that you enter for variables to persist. The values will apply only for the time the task(s) is running.
- Use the [Variables tab](#) method if you want to preserve the information (name and value) about the variables you are setting.

Both methods are available for all trigger types. You can use either method to manually launch task(s) when you cannot use the **Launch Task** button (on the task screen) because you want to override one or more variables.

The values that you enter when using either method override the all other values, set elsewhere, for those variables.

Variables set with the **Trigger with Variables pop-up** method override any specified with the **Variables tab** method, but only for that run of the task(s).

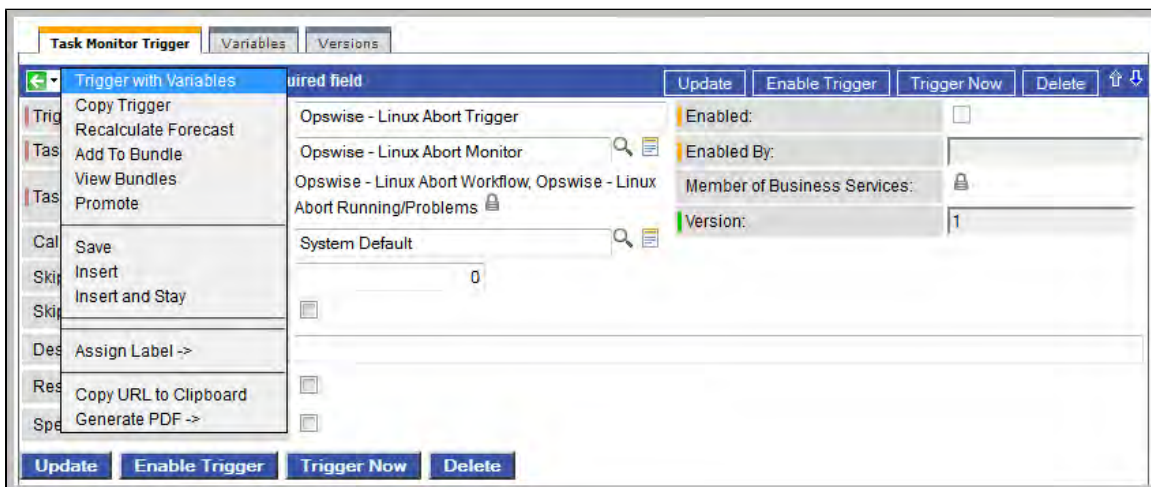
The audit message created when you use either method is the same.

Using the Trigger with Variables Pop-up Method

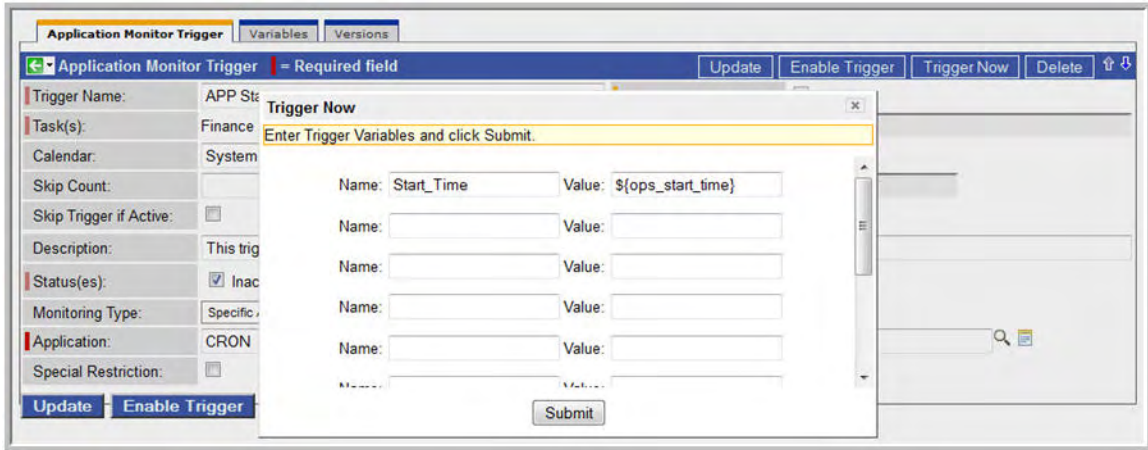
Step 1 From the navigation pane, select **Automation Center > Triggers** and a trigger type.

Step 2 Select the trigger whose tasks you want to launch.

Step 3 Hover your cursor over the the down-arrow or right-click on the title bar. The pop-up Trigger menu displays.



Step 4 Click **Trigger with Variables**. The Trigger Variables pop-up dialog displays. Any variables attached to the trigger automatically are displayed in alphabetic order (a-z).



Step 5 Enter the name and value for up to 10 variables that will be used when executing the task(s) named in the trigger.

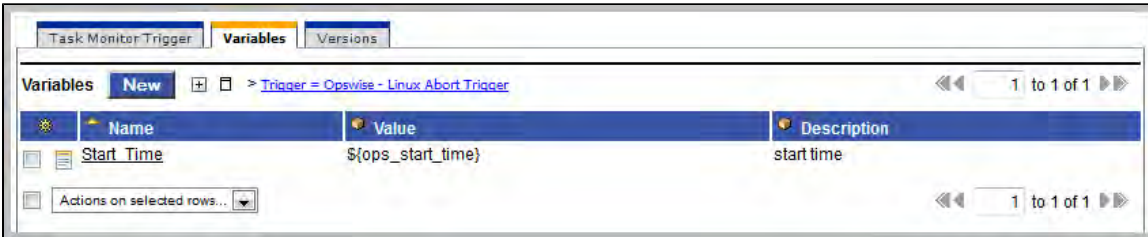
Step 6 Click **Submit** to launch the tasks named in the trigger. The variable information in this window is used where referenced in the tasks. After launching the tasks, the system deletes the contents of the window.

Using the Variables Tab Method

Step 1 From the navigation pane, select **Automation Center > Triggers** and a trigger type.

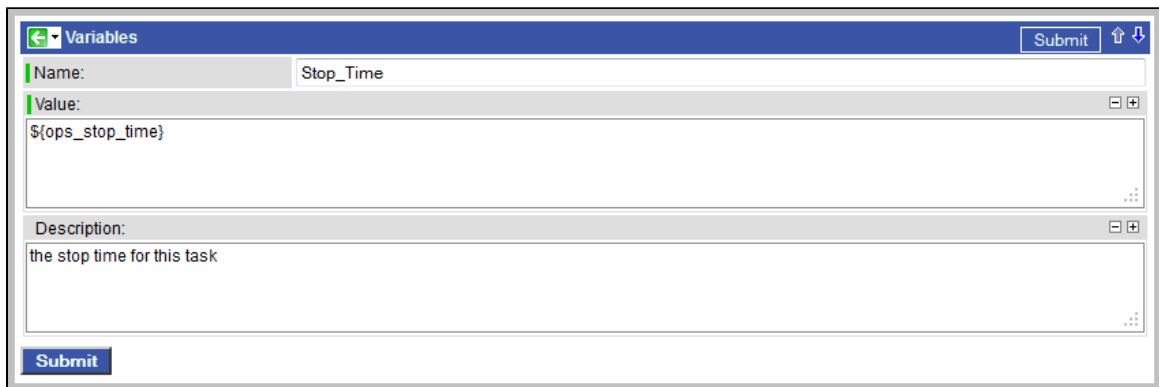
Step 2 Select the trigger whose tasks you want to launch.

Step 3 Click the **Variables** tab. The Trigger Variables List screen displays.



Step 4 To add a variable:

1. Click **New**. The Trigger Variables Definition screen displays.



2. Enter a name, value, and description for the variable and click **Submit**.

Step 5	To update a variable: <ol style="list-style-type: none">1. Click the variable name on the Trigger Variables List screen.2. Change the name, values, and/or description of the variable and click Update.
Step 6	Click the <trigger type> Trigger tab.
Step 7	Click Trigger Now to launch the tasks named in the trigger.

Displaying Trigger Forecast Information

- Overview
- Forecast Calendar
 - Customizing the Forecast Calendar to Include All Tasks
 - Displaying Forecast Details
 - Forecast Information Screen
- Forecast List
- Forecast Calculation
- Forecast Re-Calculation
- Setting up Forecasting
- Next Scheduled Time
- List Qualifying Times
 - Modifying the Qualifying Times List

Overview

Four methods are available for displaying forecasting information on time-based triggers and the tasks they launch:

- [Forecast Calendar](#) is a report that displays a calendar showing tasks scheduled to run based on Time, Cron, and Temporary triggers. Where data is available, the estimated end time for each task also is calculated and displayed.
- [Forecast List](#) is a report that displays a sequential list of the tasks scheduled to run based on Time, Cron, and Temporary triggers.
- [Next Scheduled Time](#) field on a trigger definition screen that identifies the next time a trigger will launch its task(s).
- [List Qualifying Times](#) is a button on the Time and Cron triggers screens that opens a new browser tab and displays a list of the next 30 qualifying dates and times.

Each of these methods is described below.

Forecast Calendar

For enabled Time, Temporary, and Cron triggers where forecasting has been specified, Opswise writes an entry to the Forecast Calendar for each time that a trigger task is scheduled to run within the next *N* days, where *N* is the forecast period specified in the configurable [Forecast Period in Days](#) Opswise system property.

To display the Forecast Calendar, select **Automation Center > Triggers > Forecast Calendar** from the navigation pane.

Week	Sun	Mon	Tue	Wed	Thu	Fri	Sat
40	29	30	October 1	2	3	4	5
41	6	7	8	9	10	11	12
42	13	14	15	16	17	18	19
43	20	21	22	23	24	25 PAYROLL_WF	26 PAYROLL_WF
44	27 PAYROLL_WF	28 PAYROLL_WF	29 PAYROLL_WF	30 PAYROLL_WF	31 PAYROLL_WF	November 1	2

Customizing the Forecast Calendar to Include All Tasks

By default, the Forecast Calendar displays only those tasks launched directly by the trigger, or more specifically, it does not display tasks within a workflow launched by the trigger. (For a more detailed view of forecast records, see the [Forecast List](#) screen.)

However, you can customize the Forecast Calendar to include all tasks:

Step 1	Click the Display (+) icon next to Reports > Forecast - Calendar at the top of the page to display the information fields for the Forecast Calendar.
Step 2	Delete the Workflow is empty entry in the Filter and Order section by clicking the Delete icon (X).
Step 3	Click the Save button.
Step 4	Click the Hide (-) icon next to Reports > Forecast - Calendar.



Warning

When removing the **Workflow is empty** condition with large amounts of forecast data, the loading of the calendar can take considerably longer. In such situations, it is recommended to use the [Forecast List](#) as an alternative view instead.

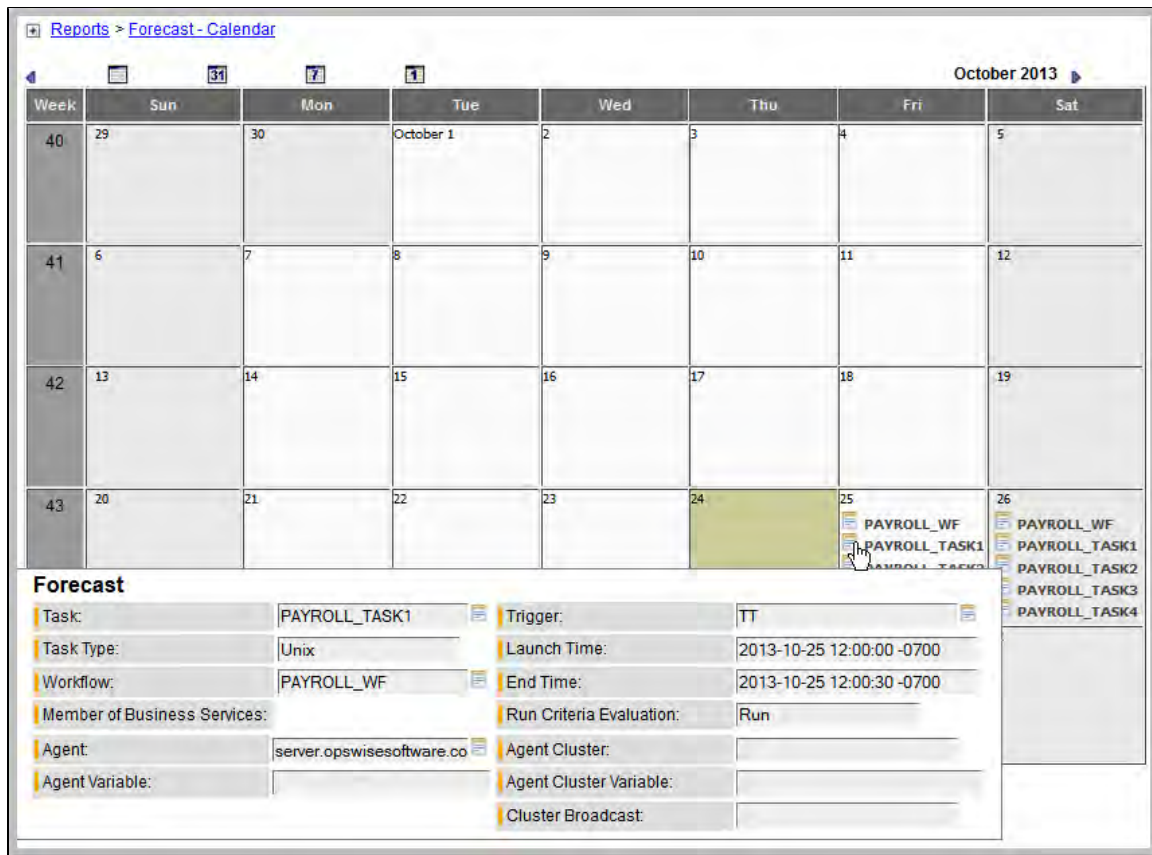
Week	Sun	Mon	Tue	Wed	Thu	Fri	Sat
40	29	30	October 1	2	3	4	5
41	6	7	8	9	10	11	12
42	13	14	15	16	17	18	19
43	20	21	22	23	24	25 PAYROLL_WF PAYROLL_TASK1 PAYROLL_TASK2 PAYROLL_TASK3 PAYROLL_TASK4	26 PAYROLL_WF PAYROLL_TASK1 PAYROLL_TASK2 PAYROLL_TASK3 PAYROLL_TASK4
44	27 PAYROLL_WF PAYROLL_TASK1 PAYROLL_TASK2 PAYROLL_TASK3 PAYROLL_TASK4	28 PAYROLL_WF PAYROLL_TASK1 PAYROLL_TASK2 PAYROLL_TASK3 PAYROLL_TASK4	29 PAYROLL_WF PAYROLL_TASK1 PAYROLL_TASK2 PAYROLL_TASK3 PAYROLL_TASK4	30 PAYROLL_WF PAYROLL_TASK1 PAYROLL_TASK2 PAYROLL_TASK3 PAYROLL_TASK4	31 PAYROLL_WF PAYROLL_TASK1 PAYROLL_TASK2 PAYROLL_TASK3 PAYROLL_TASK4	November 1	2

This Forecast calendar example has been customized to show all tasks. It shows only those tasks scheduled to be run in the current month.

To see any additional tasks scheduled to be run in the specified forecast period, click the right arrow next to the month and year to show the following month.

Displaying Forecast Details

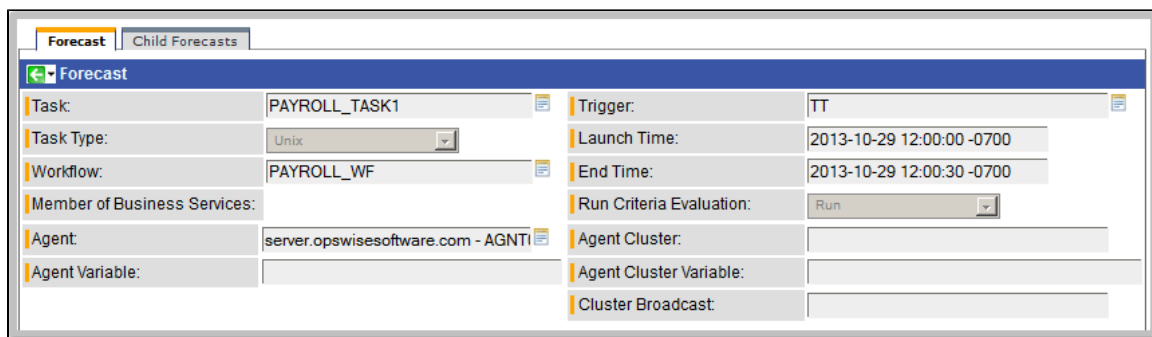
To display details of an entry in the Forecast calendar, hover your mouse over the entry icon. The following example shows details for the PAYROLL_TASK1 task of the PAYROLL_WF workflow. The same type of information displays for workflows.



Forecast Information Screen

If you want to see details about the task, workflow, agent, or trigger associated with a calendar entry, click the entry itself. (For a workflow, only details about tasks and triggers are available.) A Forecast Information screen displays information specific to that type of entry.

The following example shows the Forecast Information screen for the PAYROLL_TASK1 task of the PAYROLL_WF workflow.



Hover your cursor over a Task, Workflow, Agent, or Trigger field icon to display detailed information about it, or click the icon to display editable information about it on its Definition screen.

Forecast Information Screen Field Descriptions

Field Name	Description
Task	Name of the task selected in the Forecast Calendar. The icon is a link to the Task Definition screen for this task.
Task Type	Task type of this task.

Workflow	For tasks included in a workflow: Name of the workflow in which this task is included. The icon is a link to the Workflow Task Definition screen for this workflow.
Trigger	Name of the trigger that will launch this task (or the workflow in which this task is included). The icon is a link to the Trigger Definition screen for this trigger.
Launch Time	Calculated start time of this task.
End Time	Calculated end time of this task.
Member of Business Services	One or more Business Services that this task belongs to.
Run Criteria Evaluation	Evaluation, based on run criteria specified for this task via the Workflow Task Definition screen, of whether this task run or skip when the workflow is run. (Tasks, including workflows, launched directly by the trigger will always have a run criteria evaluation of Run . Likewise, tasks within a launched workflow that do not have any run criteria defined will always have a run criteria evaluation of Run . Any task within a workflow with run criteria utilizing variables will have a run criteria evaluation of Not Evaluated .)
Agent	Name of the agent resource definition that identifies the machine where the task will run.
Agent Variable	If enabled on the Task Definition screen for this task, the Agent field converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable.
Agent Cluster	Group of agents, one of which Opwise will choose to run this task.
Agent Cluster Variable	If enabled on the Task Definition screen for this task, the Agent Cluster field converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable.
Cluster Broadcast	Specification for a Cluster Broadcast in addition to or in place of a specific Agent and/or Agent Cluster. An agent cluster specified in this field causes Opwise to run the task on all the agents in the cluster. Each instance of the task running on its own agent becomes a separate task instance record in the database and displays separately in the Activity monitor.
Child Forecasts tab	For workflow tasks only; Displays a list of forecast information for tasks within this workflow (see Forecast List , below).

Forecast List

The Forecast List displays information about every task in the Forecast Calendar, including tasks within a workflow launched by a trigger.

To display the Forecast List, select **Automation Center > Tasks > Forecast List** from the navigation pane.

Reports > Forecast - All - List with Run Criteria Evaluation

Forecasts 1 to 100 of 155

Trigger	Task	Task Type	Workflow	Launch Time	End Time	Run Criteria Evaluation	Simulation
IT	PAYROLL_WF	Workflow		2013-10-25 12:00:00-0700	2013-10-25 12:01:43-0700	Run	false
IT	PAYROLL_TASK1	Unix	PAYROLL_WF	2013-10-25 12:00:00-0700	2013-10-25 12:00:30-0700	Run	false
IT	PAYROLL_TASK2	Unix	PAYROLL_WF	2013-10-25 12:00:31-0700	2013-10-25 12:01:01-0700	Run	false
IT	PAYROLL_TASK3	Unix	PAYROLL_WF	2013-10-25 12:00:43-0700	2013-10-25 12:01:13-0700	Skip	false
IT	PAYROLL_TASK4	Unix	PAYROLL_WF	2013-10-25 12:01:14-0700	2013-10-25 12:01:43-0700	Skip	false
IT	PAYROLL_WF	Workflow		2013-10-26 12:00:00-0700	2013-10-26 12:01:43-0700	Run	false
IT	PAYROLL_TASK1	Unix	PAYROLL_WF	2013-10-26 12:00:00-0700	2013-10-26 12:00:30-0700	Skip	false
IT	PAYROLL_TASK2	Unix	PAYROLL_WF	2013-10-26 12:00:31-0700	2013-10-26 12:01:01-0700	Run	false
IT	PAYROLL_TASK3	Unix	PAYROLL_WF	2013-10-26 12:00:43-0700	2013-10-26 12:01:13-0700	Skip	false
IT	PAYROLL_TASK4	Unix	PAYROLL_WF	2013-10-26 12:01:14-0700	2013-10-26 12:01:43-0700	Skip	false
IT	PAYROLL_WF	Workflow		2013-10-27 12:00:00-0700	2013-10-27 12:01:43-0700	Run	false
IT	PAYROLL_TASK1	Unix	PAYROLL_WF	2013-10-27 12:00:00-0700	2013-10-27 12:00:30-0700	Skip	false
IT	PAYROLL_TASK2	Unix	PAYROLL_WF	2013-10-27 12:00:31-0700	2013-10-27 12:01:01-0700	Run	false
IT	PAYROLL_TASK3	Unix	PAYROLL_WF	2013-10-27 12:00:43-0700	2013-10-27 12:01:13-0700	Skip	false
IT	PAYROLL_TASK4	Unix	PAYROLL_WF	2013-10-27 12:01:14-0700	2013-10-27 12:01:43-0700	Skip	false
IT	PAYROLL_WF	Workflow		2013-10-28 12:00:00-0700	2013-10-28 12:01:43-0700	Run	false
IT	PAYROLL_TASK1	Unix	PAYROLL_WF	2013-10-28 12:00:00-0700	2013-10-28 12:00:30-0700	Run	false
IT	PAYROLL_TASK2	Unix	PAYROLL_WF	2013-10-28 12:00:31-0700	2013-10-28 12:01:01-0700	Run	false

Forecast Calculation

As the tasks are run, Opwise calculates the end time of each forecast entry. The calculation is the average run time, based on task instances that already have run. This information is updated each time you display the forecast.

As task instances run within Automation Center, task instance durations are collected, allowing Automation Center to calculate their average run time. The average run time is used to determine the estimated end time provided on each forecast entry. For task instances that run within a triggered workflow, an average start offset within the workflow, along with the average run time, are used to determine the estimated launch time and end time.

To reset the statistics information collected by Automation Center for a particular task or workflow, use the **Reset Statistics** command (permission to use this command is assigned under [Task Permissions](#)).

Forecast Re-Calculation

Certain changes in the system will automatically cause a re-calculation of forecast data. However, not all changes will result in immediate re-calculation.

Changes to the definition of an enabled trigger that impact the schedule of that trigger or the tasks launched by that trigger will result in an immediate re-calculation of the forecast data for that trigger.

Changes to the agent, agent variable, agent cluster, agent cluster variable, or broadcast cluster fields of a task will be reflected immediately in the any forecast data referring to that particular task.

Changes to the definition of a workflow launched by a trigger or a calendar used by a trigger (including the custom days within the calendar) will result in the forecast data of an associated trigger being flagged for re-calculation, as indicated by the Forecast Recalculation Required field. Any forecast data flagged for re-calculation will be re-calculated automatically at 12:00 a.m. (midnight) daily.

Statistics for a particular task may not be available at the time the original forecast calculation occurs. Therefore, the accuracy of estimated end times for triggered tasks, as well as the estimated start and end times of tasks launched within a triggered workflow, may be inaccurate. The

current accuracy of a forecast record is indicated by the End Time Accuracy field. The End Time Accuracy is based upon the number of task instance runs for which historical data has been collected *at the time of forecast calculation*. It can have one of the following values.

Runs	Accuracy
0	0
1	Low
2-9	Medium
10+	High

Any forecast data with accuracy that can be improved significantly through re-calculation will be re-calculated automatically at 12:00 a.m. (midnight) daily.

For any forecast data for which you wish to force an immediate re-calculation, use the **Recalculate Forecast** command (permission to use this command is assigned under [Trigger Permissions](#) and [Task Permissions](#)).



Note

By default, the Forecast Recalculation Required and End Time Accuracy fields are not included in the Forecast List report. To customize the report, either:

- Click the **Personalize default** icon next to the name of the list (Forecast) and add those fields to the Selected list.
- Select the report from the [Reports](#) screen and add those columns to the Selected list.

Setting up Forecasting



Warning

We strongly discourage enabling forecasting for very frequent and predictable trigger schedules.

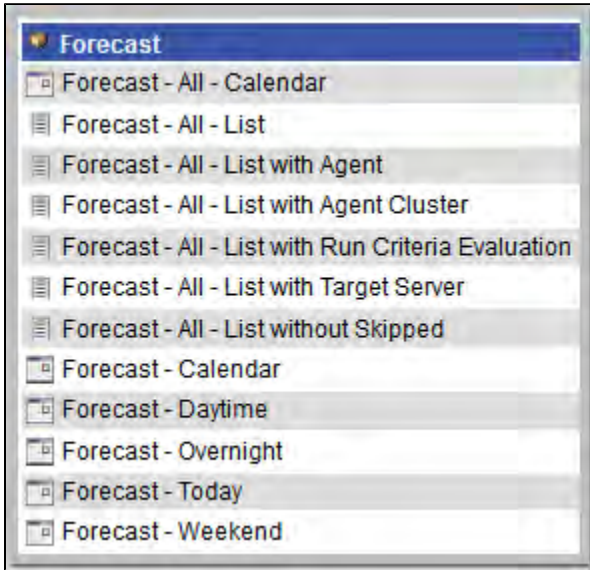
For example, if you enable forecasting on a trigger that runs every 30 seconds, that would generate - at a minimum - 89,280 forecast records, based on the default configuration of 31 days of forecasting. If that trigger launches a workflow task, it would generate an extra 89,280 forecast records per task within the workflow.

For these types of triggers, the forecast feature does not provide much insight, yet it requires a very large amount of processing to compute.

Use the following points as a checklist when setting up forecasting:

- Forecasting is supported for the following trigger types: Time, Temporary, and Cron.
- In the trigger, enable the Forecast field.
- Specify the number of days for which you want scheduled tasks to display in the Forecast Calendar / Forecast List (default is 31):
 1. Select **Automation Center Administration > Configuration > Properties**. (You need administrative privileges to access this function.)
 2. Click **Forecast Period In Days** and enter the number of days you want in the forecast period.
 3. Click **Update**.
- Enable the trigger. (Disabling the trigger removes all related entries from the Forecast Calendar / Forecast List.)
- The Forecast calendar is generated by the report **Forecast - All - Calendar**. The Forecast List is generated by the report **Forecast - List with Run Criteria Evaluation**. You also can navigate to the Reports menu and run several other pre-defined forecasting reports, as

shown in the following section of the Reports page:



Next Scheduled Time

For enabled Time, Temporary, and Cron triggers, Opswise calculates the next scheduled time and displays it on the Triggers List screen, as well as on the All Triggers and Active Triggers screens, for those trigger types.

A screenshot of a web application's 'Triggers' list. The table has the following columns: Trigger Name, Type, Description, Task, Next Scheduled Time, and Enabled. The 'Next Scheduled Time' column is circled in red. The table contains six rows of data, all with 'Enabled' status checked.

Trigger Name	Type	Description	Task	Next Scheduled Time	Enabled
Stress 1	Cron	Every 1 Minute of Every Day.	Linux Simple	2008-05-30 13:29:00 -0700	✓
Stress 2	Time	Run every minute	Linux Simple	2008-05-30 13:30:21 -0700	✓
Stress 3	Cron	Every 5 Minutes of Every Day	5x5 linux	2008-05-30 13:30:00 -0700	✓
Stress 4	Cron	Every 8 Minutes of Every Day.	5x5x5 linux	2008-05-30 13:32:00 -0700	✓
Stress 5	Cron	Every 12 minutes	5x5x5x5 Linux	2008-05-30 13:36:00 -0700	✓
Stress 6	Cron	Every 24 minutes	5x5x5x5 Linux	2008-05-30 13:48:00 -0700	✓

The next scheduled time also displays within the trigger record:

The screenshot shows the 'Time Trigger' configuration interface. At the top, there are tabs for 'Time Trigger', 'Variables', and 'Versions'. Below the tabs is a toolbar with buttons: 'Update', 'List Qualifying Times', 'Disable Trigger', 'Trigger Now', and 'Delete'. The main form contains the following fields:

- Trigger Name: Opswise - Every Independence Day
- Task(s): Opswise - Workflow of Sleep Tasks
- Calendar: Opswise - American Calendar
- Skip Count: 0
- Skip Trigger if Active:
- Simulate: - System Default -
- Description: Run every Independence Day
- Time Zone: System (US/Pacific)
- Time Style: Time
- Day Style: Complex
- Date Adjective: Every
- Date Noun: Ops - Independence Day
- Date Qualifier: Year
- Date Adjustment: None
- Special Restriction:
- Next Scheduled Time: 2014-07-04 03:00:00 -0700 (highlighted with a red box)

At the bottom, there is another set of buttons: 'Update', 'List Qualifying Times', 'Disable Trigger', 'Trigger Now', and 'Delete'.

List Qualifying Times

For Time, Temporary, and Cron triggers, you can display a list of the next 30 dates and times when the trigger will be satisfied by clicking the **List Qualifying Times** button on the trigger definition screen. Opswise displays the list in a new browser tab.



Note

This display differs from the Forecast list, which shows scheduled task instances as opposed to qualifying times.

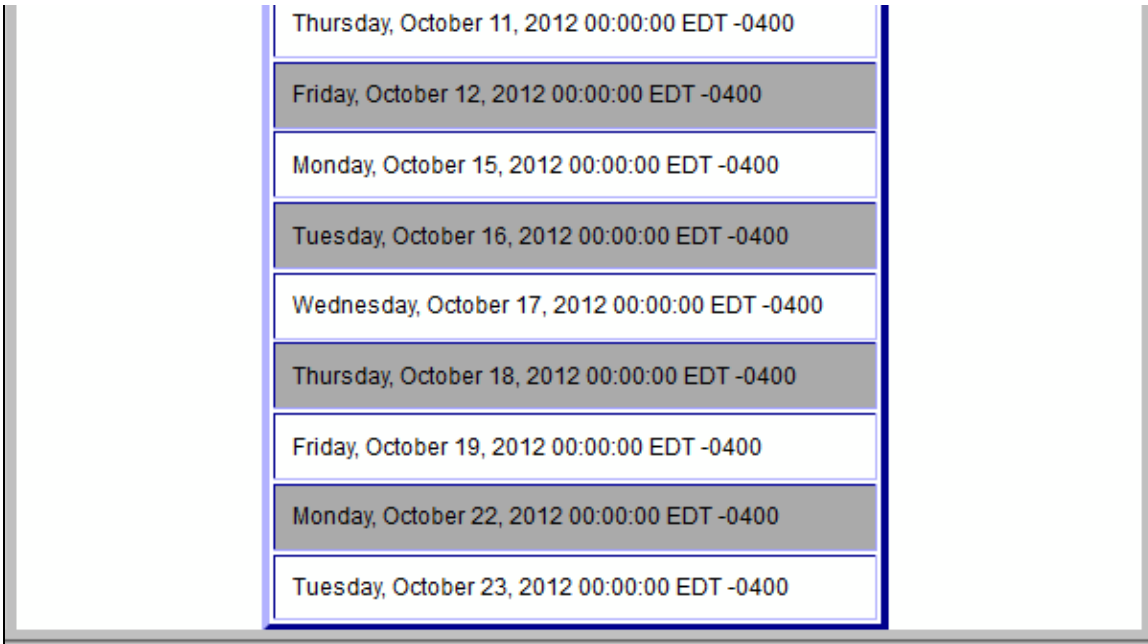
The following example shows the next 30 dates and times when a Time trigger will be satisfied.

Trigger Name: Workflow #1 Trigger

Trigger for Workflow #1

Listing From "Tuesday, September 11, 2012 17:22:14 EDT -0400"

User/Trigger Timezone: America/New_York
Wednesday, September 12, 2012 00:00:00 EDT -0400
Thursday, September 13, 2012 00:00:00 EDT -0400
Friday, September 14, 2012 00:00:00 EDT -0400
Monday, September 17, 2012 00:00:00 EDT -0400
Tuesday, September 18, 2012 00:00:00 EDT -0400
Wednesday, September 19, 2012 00:00:00 EDT -0400
Thursday, September 20, 2012 00:00:00 EDT -0400
Friday, September 21, 2012 00:00:00 EDT -0400
Monday, September 24, 2012 00:00:00 EDT -0400
Tuesday, September 25, 2012 00:00:00 EDT -0400
Wednesday, September 26, 2012 00:00:00 EDT -0400
Thursday, September 27, 2012 00:00:00 EDT -0400
Friday, September 28, 2012 00:00:00 EDT -0400
Monday, October 01, 2012 00:00:00 EDT -0400
Tuesday, October 02, 2012 00:00:00 EDT -0400
Wednesday, October 03, 2012 00:00:00 EDT -0400
Thursday, October 04, 2012 00:00:00 EDT -0400
Friday, October 05, 2012 00:00:00 EDT -0400
Monday, October 08, 2012 00:00:00 EDT -0400
Tuesday, October 09, 2012 00:00:00 EDT -0400
Wednesday, October 10, 2012 00:00:00 EDT -0400



Modifying the Qualifying Times List

You can modify the Qualifying Times List to show greater or fewer qualifying dates/times and to specify a specific date for when the list will start:

Step 1	Hover your cursor over the down arrow on the title bar or right-click on the title bar.
Step 2	On the displayed pop-up menu, click List Qualifying Times to display the List Qualifying Times dialog.
Step 3	Select the next number of date/times when the trigger will be satisfied and/or a different start date for when to begin listing this number of qualifying dates/times.
Step 4	Click submit to display the Qualifying Times list.

Calendars

Calendars define business days, holidays, and other special days. Opwise uses calendars, in conjunction with [triggers](#), to define when tasks are run.

Using Calendars

Opwise uses the calendar specified in a trigger to determine the run dates for the task(s) specified in that trigger:

- If you select Business Days in a trigger, the calendar identifies those business days.
- If you select Day Style = Complex in a trigger:
 - All [custom days](#) - for a single day - attached to the calendar are selectable day types (in the trigger's Date Noun drop-down menu).
 - All custom days - for a period of days - attached to the calendar are selectable day types (in the trigger's Date Qualifier drop-down menu).
- If you select Special Restriction in a trigger, the calendar defines the Holidays or Non Business days.

Setting up Calendars

The process for setting up your calendars is:

Step 1	Define the custom days you will need.
Step 2	Create one or more calendars .
Step 3	Create one or more copies of any calendar, as desired.
Step 4	Assign the custom days to the calendar(s).

Creating Custom Days

- [Creating Custom Days](#)
- [Custom Days Field Descriptions](#)

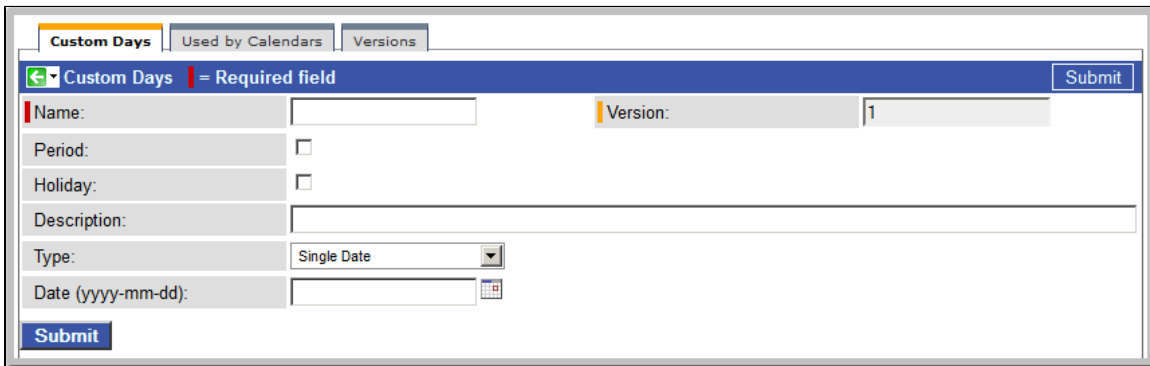
Creating Custom Days

A custom days definition defines a single one-time date, a repeating date, or a list of dates. Custom days are attached to calendars.

1. From the navigation pane, select **Automation Center > Custom Days**. The Custom Days List screen displays:



2. Click **New**. The Custom Days Definition screen displays.




3. Using the field descriptions provided below, fill in the fields.
4. Click the **Submit** button to save the record and return to the menu or right-click on the title bar and select **Save** to save the record and remain on the current display.

Custom Days Field Descriptions

The following table provides descriptions of the fields on the Custom Days form.

Field Name	Description
Name	Required. Name for this Custom Day.

Version	System-supplied. The version number of the current record, which is incremented by the system every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Period	<p>Indication of whether or not this Custom Days record defines a custom period of days (for example: quarters, fiscal year, or 4-5-4 calendar).</p> <p>Custom periods can be selected in:</p> <ul style="list-style-type: none"> • Date Qualifier field for Time Triggers. • Restriction Qualifier field for Time, Cron, File, Task Monitor, and Application Monitor triggers. <div style="background-color: #ffffcc; padding: 10px; margin: 10px 0;"> <p> Note If the Custom Days record is for a specific day rather than a period, it can be selected in:</p> <ul style="list-style-type: none"> • Date Noun field for Time triggers. • Restriction Noun field for Time, Cron, File, Task Monitor, and Application Monitor triggers. </div>
Holiday	Indication of whether or not this Custom Days record is defining a holiday. Dates flagged as holidays come into play when the user enables Special Restriction on a trigger and selects a situation of On Holiday.
Description	Description of this day, which displays on the Custom Days list.
Type	<p>Options:</p> <ul style="list-style-type: none"> • Single Date - Any one-time date. • Relative Repeating Date - An annual (repeating) date that changes from year to year. For example, the U.S. Thanksgiving falls on the 4th Thursday of November, and is therefore on a different date every year. • Absolute Repeating Date - An annual (repeating) date that does not change from year to year. For example, the Canadian holiday Canada Day falls on July 1st of every year. • List of Dates - The dates are listed below.
Date (yyyy-mm-dd)	<p>If Type = Single Date: Specific date for this Custom Days definition (entered manually or selected from Calendar tool).</p> <p>If Type = List of Dates: Multiple specific dates for this Custom Days definition (entered manually or selected from Calendar tool).</p>
When	If Type = Relative Repeating Date: Occurrence of this day in the month. Options: 1st, 2nd, 3rd, 4th, Last, Every. Example: The 4th Thursday of November.
Day Of Week	If Type = Relative Repeating Date: Day of the week. Example: The 4th Thursday of November.
Month	If Type = Relative Repeating Date or Absolute Repeating Date: Month of the year, or All. Example: The fourth Thursday of November.
Day	If Type = Absolute Repeating Date: Day of the month. Options: 1 through 31. Example: December 25.
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
List Qualifying Dates button	Opens a new browser tab and displays a list of the next 20 dates on which this day occurs.
Delete button	Deletes the current record.
Used by Calendars tab	Displays all calendars that use this custom day.

Versions tab	Stores copies of all previous versions of the current record. See Record Versioning .
---------------------	---

Creating Calendars

- Creating a New Calendar
 - Calendar Field Descriptions
- Assigning Existing Custom Days to the Calendar
- Adding New Custom Days and Assigning Them to this Calendar

Creating a New Calendar

Step 1 From the Navigation Pane, select **Automation Center > Calendars**. The Calendar list screen displays:

Name	Description	Updated by	Updated
Opswise - American Calendar	American Calendar	ops.admin	2008-11-26 14:35:04 -0800
System Default	Default System Calendar	glide.maint	2007-11-25 05:55:21 -0800

Note
SystemDefault, the default system calendar, defines the work week.

Calendar Definition Form:

- Name: System Default
- Description: Default System Calendar
- Business Days: Sunday Monday Tuesday Wednesday Thursday Friday Saturday
- Version: 1

Step 2 Click **New**. The Calendar Definition screen displays.

Calendar Definition Form (New):

- Name: [Empty]
- Description: [Empty]
- Business Days: Sunday Monday Tuesday Wednesday Thursday Friday Saturday
- Version: 1

Step 3 Give the calendar a name and description, and select the business days for this calendar. The default selection is Monday through Friday.

Step 4 Right-click on the title bar and select **Save** to save the record and remain on the current display.

Step 5 Follow the appropriate instructions below to [assign existing custom days](#) to this calendar or [add new custom days](#) to the calendar.

Step 6 If appropriate, repeat these steps for any additional calendars you want to add.

Calendar Field Descriptions

The following table provides field descriptions for the calendar form.

Field Name	Description
Name	Required. Name used within Opwise to identify this calendar. Up to 40 alphanumerics. It is the user's responsibility to develop a workable naming scheme for calendars.
Member of Business Services	User-defined. Allows you to select one or more Business Services that this record definition belongs to. Click the lock icon to unlock the field and select Business Services .
Description	User-defined. Provides a description for the calendar.
Business Days	User-defined. Allows the user to select which days of the week constitute business days for this calendar.
Version	Task definition only; system-supplied. The version number of the current record, which is incremented by Opwise every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Calendar Preview button	Opens a new browser and displays a calendar for the next four years (including the current year). All the dates specified in this calendar are highlighted and identified.
Delete button	Deletes the current record.
Custom Days tab	Displays all custom days associated with this calendar.
Triggers tab	Displays a list of all triggers that use this calendar.
Versions tab	Stores copies of all previous versions of the current record. See Record Versioning .

Assigning Existing Custom Days to the Calendar

Step 1	On the Calendar Definition screen, click the Custom Days tab.
---------------	--

Step 2 On the Has Custom Days screen, click the **Edit** button. The Edit Members screen displays:

Step 3 The days listed under Collection are existing Custom Days that do not already belong to this calendar. The days listed under Has Custom Days List are days that belong to this calendar.

Step 4 To filter the days listed under Collection:

1. Select filter conditions in the --choose field--, --oper--, and --value-- fields. (See [Create a Filter](#) for information about how to construct a filter.)
2. If you want to add more filter conditions, click **Add Filter**.
3. When you have defined the filter you want, click **Run Filter**. The Collection list now displays only those days that match the filter.
4. To remove filter conditions, click the X (Delete) icon that displays to the right of each set of filter conditions, and then click **Run Filter**.

Step 5 To add to or remove days from the Has Custom Days List:

- To add a day to the list, double-click on the day in the Collection list.
- To remove a day from the list, double-click on the name in the Has Custom Days List.

Step 6 As you click on a day the system displays details about the day at the bottom of the form.

Step 7 When you are finished, click **Save**.

Adding New Custom Days and Assigning Them to this Calendar

Step 1	Open the calendar to which you want to add and assign custom days.
Step 2	Click the Custom Days tab to display the Has Custom Days screen.
Step 3	Click New to display the Custom Days Definition screen.
Step 4	Fill in the fields (see Creating Custom Days).
Step 5	Click Submit to automatically add the Custom Day to the Has Custom Days List.

Copying Calendars

- Overview
 - Copying One or More Calendars on the Calendar List
 - Copying a Calendar on the Calendar Definition Form

Overview

You can make copies of all Opswise records, including calendars, using the standard methods for copying: Insert, Insert and Stay (see [Saving, Updating, Deleting, and Copying Records](#)).

However, these methods do not make copies of other records that may be attached to the calendar, such as Notifications, Actions, Variables, and so on.

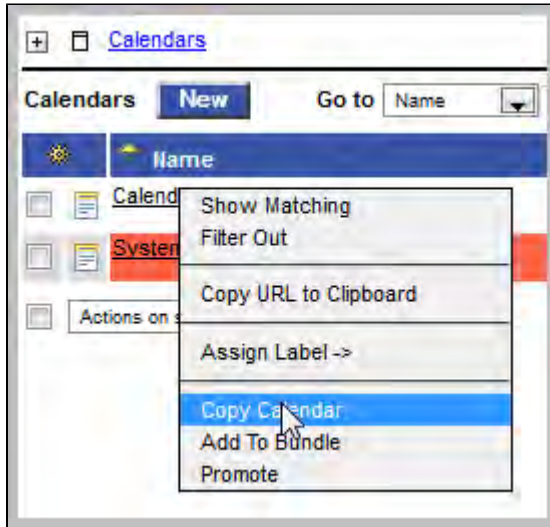
The Copy Calendar option allows you to make a complete copy of a calendar, including all of its associated records, such as variables and notes. It does not copy referenced records, such as virtual resources, but retains the relationship to these records for the copied calendar.

Copying One or More Calendars on the Calendar List

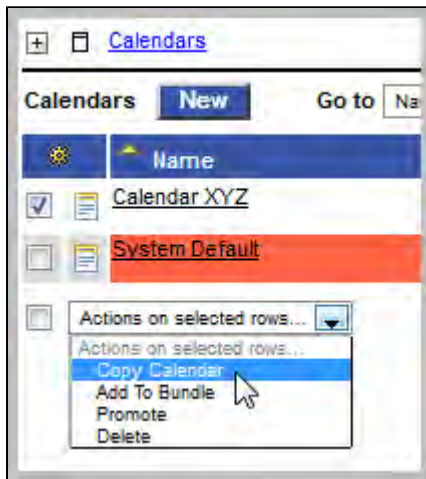
Step 1	Display the calendar list by selecting Calendars from the navigation pane.
Step 2	Locate the calendars you want to copy (see Searching for Records).

Step 3 Two methods are available for copying calendars:

1. To copy a single calendar, hover over the calendar name and right-click. From the pop-up menu, select **Copy Calendar**.



2. To copy multiple calendars, click the box to the left of each calendar name. From the **Action on selected rows...** drop-down list at the bottom of the page, select **Copy Calendar**.



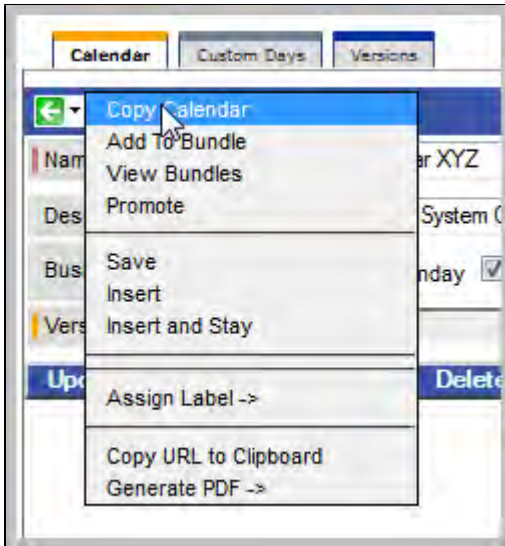
Step 4 The system copies the calendar(s), automatically creating the new name by prepending the original name with "Copy of" (for example, "Copy of Calendar XYZ"), and adds it to the list. If the new name already exists, the system appends a counter to the name, such as "Copy of Calendar XYZ 1", "Copy of Calendar XYZ 2", and so on, until it finds a name that is available.

Step 5 To modify the name or any other information in the calendar, open the new calendar, make your changes, and click **Update**.

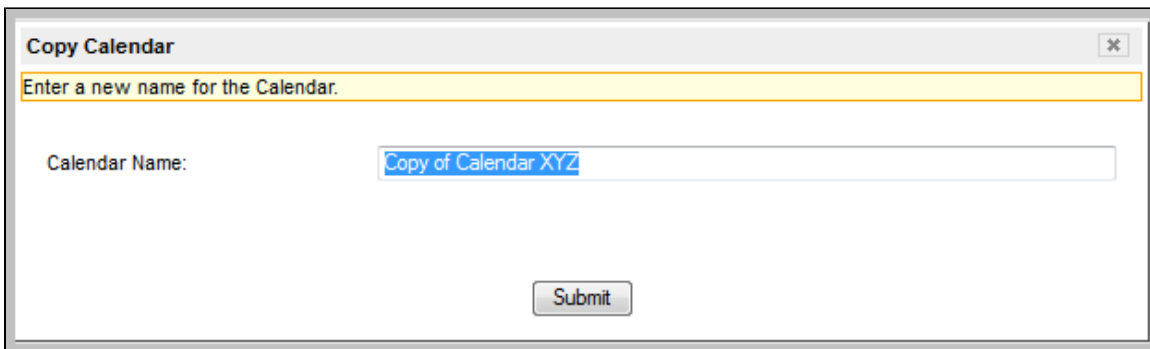
Copying a Calendar on the Calendar Definition Form

Step 1 Open the calendar you want to copy.

Step 2 Hover over the down arrow above **Calendar Name:**. A pop-up menu displays:



Step 3 Click **Copy Calendar**. Another window appears, prompting for a name for the new calendar. The default is the original calendar name, prepended with "Copy of," as shown in the following example:



Step 4 Enter a new name for the calendar and click **Submit**. The system copies the calendar and all its attachments and saves it under the new name. (If the new name already exists, the copy will fail.)

Application Monitoring and Control



Application Monitoring and Control

Overview

Application Monitoring Processing Flow

Application Resources

Application Control Tasks

Application Monitor Trigger



The information on these pages also is located in the [Opswise Automation Center 5.1.1 User Guide.pdf](#).

Application Monitoring and Control Overview

- [Application Monitoring and Control](#)
- [Processing Flow](#)

Application Monitoring and Control

The Opwise Application Monitoring and Control feature of Opwise Automation Center allows you to use it as a network control and monitoring tool. You can use Application Monitoring and Control to start, stop, and query any application running on any machine where you have an Opwise agent installed and running.

You will use the three components of Application Monitoring and Control to monitor your applications:

- The [Application Resource](#) record defines the name and location of the application, along with the necessary control commands. The [list of application records](#) displays a status for each application.
- Three [Application Control tasks](#) are automatically generated when you save an Application Resource record to the Opwise database: one each for executing a start, stop, and query command against the application. You can use these control tasks to schedule the Start, Stop, and Query commands in Workflows and triggers. You can also create customized Application Control tasks where necessary.

**Note**

You can manually run an Application Control task to execute a command specified in an Application resource record, but it is simpler to just execute the command from the Applications list or Application resource record.

- Optional [Application Monitor triggers](#) allow you to launch other tasks based on the status of the Application Resource record being monitored.

Processing Flow

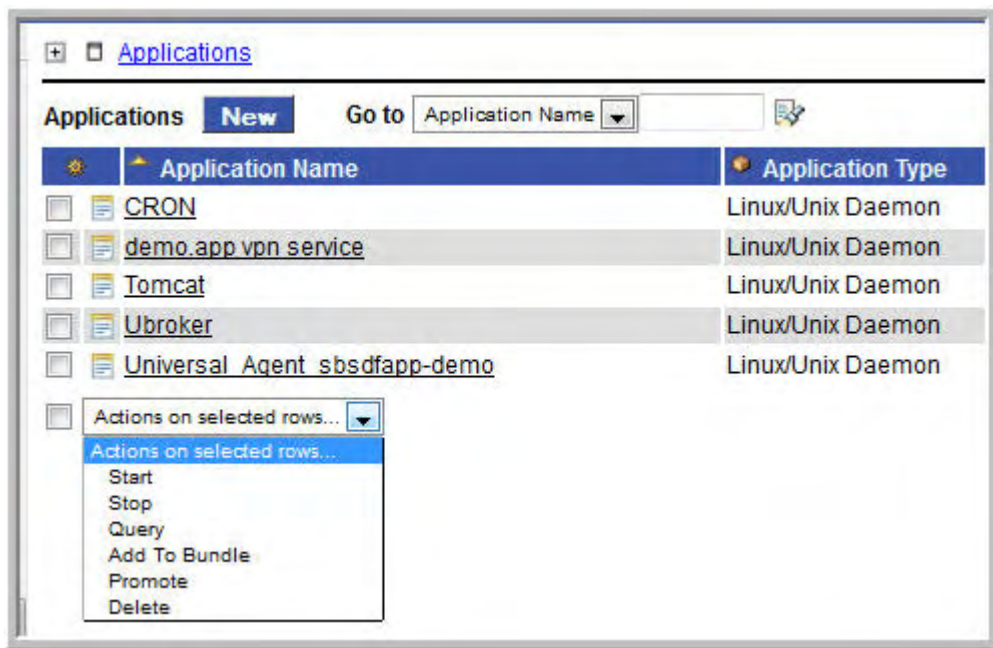
The following steps show a sample process flow for the manual monitoring (that is, not via a trigger or Workflow) of an application:

- | | |
|---------------|--|
| Step 1 | Use the Application Resource screen to create an Application resource record, specifying the name of an application and the start, stop, and query commands to control it. The Controller will automatically create three Application Control tasks that you can use in Workflows and triggers for starting, stopping, and querying the application. |
|---------------|--|

Step 2 Start the application using either of the following methods:

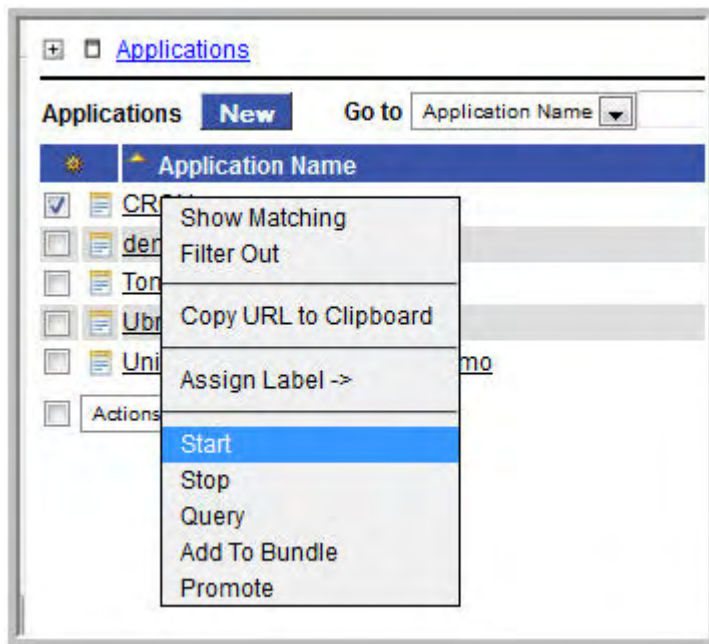
- **Method 1**

1. Click the box to the left of the Application name to select it.
2. From the **Actions on selected rows...** menu, select **Start**.



- **Method 2**

1. To start a single application, right-click the Application Name.
2. Select **Start** from the pop-up menu.



Step 3 Opwise executes the Start command, as provided by the user in the Application Resource definition. It puts the Application into Starting status, and saves the Start Time. The Start command has two functions:

1. It starts the application.
2. It starts the query process that monitors the application.

Step 4 After 30 seconds, Opswise automatically executes a Query command, as provided by the user in the Application resource definition, to determine the status of the application. Opswise continues executing the Query command every 120 seconds thereafter until the user stops the monitoring by issuing a Stop command from Opswise.

Start Command:	sudo /etc/init.d/crond start
Stop Command:	sudo /etc/init.d/crond stop
Query Command:	sudo /etc/init.d/crond status

Step 5 The purpose of the Query is to determine whether or not the Application is Active. Opswise uses the specifications provided by the user the Exit Code Processing fields (example shown below) to make this determination.

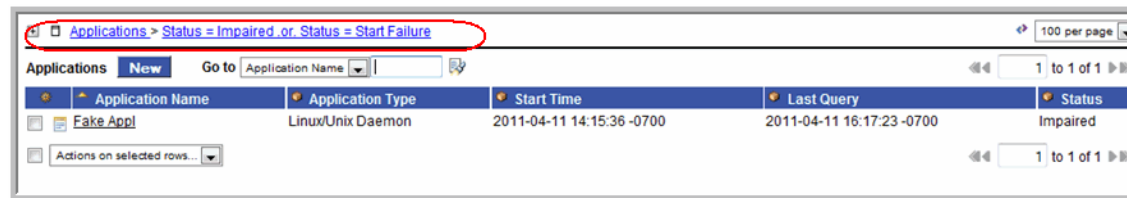
Query Exit Code Processing:	Success Output Contains	Output Type:	Standard Output
Scan Output For:	running		

- If the response from the application indicates a successful startup, Opswise puts the Application in Active status.
- If the response indicates the Application hasn't started, Opswise continues executing the Query (keeping track in the Startup Query Attempts field) until it reaches the maximum attempts specified by the user in the Startup Query Maximum field. If the maximum number is reached before achieving an Active status, Opswise puts the Application into Impaired status. However, Opswise continues monitoring. If the appropriate exit code parameters are eventually returned, the application will go to Active status. The purpose of the Startup Query Attempts field is to avoid having the application go right into Impaired status if it takes awhile to start.

Opswise writes any Exit Code captured by the Query in the Query Exit Code field.

Step 6 After starting the application, Opswise continues monitoring by sending out the Query commands every 120 seconds.

- If Opswise detects a problem based on the Exit Code parameters, it puts the Application into Impaired status. If this occurs, you have several options for handling the problem, with increasing levels of automation:
 1. The Application list displays the status of all Applications being monitored. You could [create a filter](#) for the Application list that displays only those with status of Impaired (or other), as shown in the example below. When you see a problem, troubleshoot the issue and restart the Application from outside Opswise.



2. Set up an [Application Monitor trigger](#) that monitors this application for Impaired and other problem statuses. When the trigger is satisfied, it launches an email task that sends emails to support personnel, notifying them of the problem. Several [built-in variables](#) are supported that allow you to pass required data into the email message: the application name, type, and status.
 3. You also could create a [workflow](#) launched by an Application Monitor trigger looking for Impaired or other problem statuses. The workflow can include Application Control tasks that attempt to resolve the problem by stopping and then re-starting the application. You could also include any other tasks that are specific to troubleshooting the application.
- If Opswise fails to get a response to a Query for three minutes, it puts the Application into Query Overdue status, where is the known query status. This can be any of the following: Starting/Query Overdue, Active/Query Overdue, and Impaired/Query Overdue. For example, you may see this status if the agent went down or there was some other problem on the machine unrelated to the application itself. If this occurs, you should troubleshoot the issue. When you have fixed the problem, the continued queries from Opswise will then return an Active status for the application.

Step 7 To stop monitoring an Application, issue the Stop command against it. Opswise stops the Application and puts it into Inactive status, which means it is no longer monitoring.

Applications

- [Overview](#)
- [Built-In Variables](#)
- [Creating a New Application Resource Record](#)
- [Application Field Descriptions](#)

Overview

Application Resource records are the core component of the Opwise Application Monitoring and Control function. These records define the names of the Applications being monitored, the name and location of the machines where they are running, and the start, stop, and query commands needed to perform the monitoring and control functions.

Shown below is a sample Applications resource list, which displays all the Applications you have set up to be monitored. You must manually refresh this screen (click **Applications** in the navigation pane) to fetch the latest status information.

Application Name	Application Type	Start Time	Last Query	Status
CRONNY	Linux/Unix Daemon	2011-03-30 09:35:51 -0700	2011-03-30 15:17:00 -0700	Active
TomcatSF-01	Linux/Unix Daemon			Inactive
TomcatSF-02	Linux/Unix Daemon			Inactive

You can also use Application resources and their associated [Application Control tasks](#) to start, stop, and query applications as part of your scheduling processes. You can execute Application Control tasks as you would execute any other task and include them in workflows where applicable. In addition, you can define [Application Monitor triggers](#) to automatically launch one or more tasks of any type, depending on the status of one or more Application resources. For example, you might set up an Application Monitor trigger that sends an email to Windows tech support personnel if any Windows application goes to Impaired or Inactive status.

In order for Opwise to access the application, the application must be installed on a machine where an Opwise Windows, Linux/Unix, or z/OS agent is running.

If you set up Opwise to monitor your applications, you should always start and stop the applications from within Opwise. If you stop an application outside of Opwise, you must also restart it from outside of Opwise. If Opwise detects a problem with an application (the application goes to Impaired status), you should troubleshoot the problem and restart the application outside of Opwise. Opwise will continue monitoring and when it detects that the application is back up, it will put the application back into Active status.

Built-In Variables

[Application Monitor built-in variables](#) are provided to pass information about the Application being monitored into the task or tasks being launched by the trigger. You can pass the information into the launched tasks by including the variables in a text field in the task definition.

Creating a New Application Resource Record

Step 1 From the navigation pane, select **Automation Center Resources > Applications**.

Step 2 Click **New**. The Application screen displays.

Step 3 Using the field descriptions provided below as a guide, complete the fields as needed.

Step 4 Click the **Submit** button to save the record and return to the menu, or right-click on the title bar and select **Save** to save the record and remain on the current display. When you save the new Application Resource record, Opwise also automatically creates three related [Application Control Tasks](#), one each for starting, stopping, and querying the application.

Step 5 If appropriate, repeat these steps for any additional Applications you want to add.

Application Field Descriptions

Field Name	Description
Application Name	Required. Name used within Opwise to identify this resource. Up to 40 alphanumeric. It is the user's responsibility to develop a workable naming scheme for resources.
Application Type	User-defined. The type of application. Options: <ul style="list-style-type: none"> • Windows Service • Linux/Unix Daemon • z/OS Started Task
Credentials	Optional. The login credentials that Opwise will use to access the remote machine. For z/OS application resources, make sure the credentials are in upper case.
Agent	Required. The name of the Windows, Linux/Unix, or z/OS agent resource that describes the machine where the application will run.
Run as sudo	Optional; Linux/Unix only. Run the command as Sudo (superuser do).
Member of Business Services	User-defined. Allows you to select one or more Business Services that this record definition belongs to. Click on the lock icon to unlock the field and select Business Services .

Startup Query Maximum	User-defined. The default is 1. Allows you to specify the maximum number of times that Opswise should query for Active status during start-up before it puts the application into Impaired status. For applications that take awhile to start, you should specify a higher number. When you issue a Start command, Opswise issues the Start, waits 30 seconds, then executes the Query command. It continues executing the Query command every 120 seconds thereafter (until you issue a Stop command). If the start-up takes longer than the maximum queries specified, the application goes to Impaired status. However, Opswise continues to query. If the required exit codes are eventually returned, the application then goes to Active status.
Startup Query Attempts	System-supplied. The number of queries that were executed before the Application went into Active or Impaired status.
Query Exit Code	System-supplied. The most recent exit code returned by the application in response to a query.
Runtime Directory	Optional. The directory where the application executes. Variables supported.
Start Command	Required. The command used to start the application. This can be any process or command that starts the application. If you try to start an application monitor that is already started, you will see the message: Application already monitored with <status> status.
Stop Command	Required. The command used to stop the application. This can be any process or command that stops the application.
Query Command	Required. The command used to query the application. This can be any process or command that queries the application. You must first start the application monitor from Opswise before you can query the application.
Query Exit Code Processing	Specifies how Opswise should determine whether or not the application is running. Options: <ul style="list-style-type: none"> • Success Exitcode Range - Application goes to or remains in Active status if its exitcode falls within the range specified in the Query Exit Codes field (see below). Otherwise it has Impaired status. • Failure Exitcode Range - Application goes to or remains in Impaired status if its exitcode falls within the range specified in the Exit Codes field (see below). Otherwise it has Active status. • Success Output Contains - Application goes to or remains in Active status if its output contains the text specified in the Scan Output For field (see below). Otherwise it has Impaired status. • Failure Output Contains - Application goes to or remains in Impaired status if its output contains the text specified in the Scan Output For field (see below). Otherwise it has Active status.
Query Exit Codes	Required if Query Exit Code Processing = Success Exitcode Range or Failure Exitcode Range. Specifies the range. Format: Numeric. Use commas to list a series of exit codes; use hyphens to specify a range. Example: 1,5, 22-30.
Output Type	Required if Query Exit Code Processing = Success Output Contains or Failure Output Contains. Specifies the type of output. Options: Standard Output (STDOUT) Standard Error (STDERR) File
Scan Output For	Required if Query Exit Code Processing = Success Output Contains or Failure Output Contains. Specifies the string that Opswise should scan for in the output.
Output File	Required if Output Type=File. The path and name of the file.
Status	System-supplied. Indicates the current status of the application. One of the following: <ul style="list-style-type: none"> • Inactive - Application is not being monitored by Opswise. • Start Failure - Application failed to start. This may occur, for example, if you have problems with credentials or the start command itself is incorrect. When this occurs, Opswise is not monitoring the application. You should troubleshoot the problem and restart the application from Opswise. • Starting - Start command has been executed. • Active - Application has successfully started and is running, based on the parameters specified in the Exit Code processing fields. • Impaired - An application that is being monitored returned a response that, based on the specified exit code parameters, indicates it is not running. If this occurs, you should troubleshoot the problem and restart the application from outside Opswise. Unless you issue a stop command, Opswise continues monitoring during this process. When the application comes back up, the query process will recognize this and return the application to Active status.
Status Description	System-supplied. A more detailed status message describing why a status change occurred, in the format: "Query exit code <in-not in> <success-failure> exit code range. Query <success-failure> output not found."
Start Time	System-supplied. Date and time that the application was last started by Opswise.
Last Query	System-supplied. Date and time of the last query response received from the application.
Version	Version number of the current record, which is incremented by Opswise every time a user updates a record. Click on the Versions tab to view previous versions. For details, see Record Versioning .

Application Control Tasks tab	Lists all Application Control tasks associated with this Application resource.
Application Control Task Instances tab	Lists all Application Control task instances associated with this Application resource.
Versions tab	Stores copies of all previous versions of the current record. See Record Versioning .
Update button	Updates this record with any changes.
Start button	Executes the Start command associated with this Application resource and begins querying.
Query button	Executes the Query command associated with this Application resource. This allows you to get immediate status of the application instead of waiting for the next automated query.
Stop button	Executes the Stop command associated with this Application resource. Opwise stops the application and stops querying (monitoring).
Delete button	Deletes this Application Resource record.

Application Control Tasks

- [Overview](#)
- [Built-In Variables](#)
- [Creating a New Application Control Task](#)
- [Application Control Task Field Descriptions](#)
- [Specifying When a Task Runs](#)
- [Monitoring Task Execution](#)

Overview

Application Control tasks allow you to execute a start, stop, or query command against an application in the Opwise network. Three Application Control tasks are created automatically when you define and save an [Application Resource record](#) – one each for starting, stopping, and querying the application.

The following shows three Application Control tasks created when an Application resource record called "Appl Tomcat" was saved to the Opwise database:

Task Name	Type	Summary	Updated
Appl Tomcat #QUERY#	Application Control		2011-03-17 07:53:32 -0700
Appl Tomcat #START#	Application Control		2011-03-17 07:26:05 -0700
Appl Tomcat #STOP#	Application Control		2011-03-17 07:26:05 -0700

Each of these auto-created records is stored as a separate task record and can be executed independently or added to a workflow like any other task.

The following shows the task record for the Appl Tomcat#START# task that was automatically created when the Appl Tomcat Application resource record was submitted to the database. Note that many of the fields are protected and the Generated field at the bottom is selected, indicating that this task was generated automatically. You cannot delete an auto-generated Application Control task.

Application Control Task | Variables | Actions | Task Virtual Resources | Mutually Exclusive Tasks | Task Instances | Triggers | Notes | Versions

Application Control Task = Required field Submit

Task Name: Tomcat#START# Member of Business Services:

Version: 1 Hold on Start:

Task Description:

Application: Tomcat Command: Query

Generated:

Late Start:

Late Finish:

Early Finish:

Virtual Resource Priority: 10 Hold Resources on Failure:

Submit

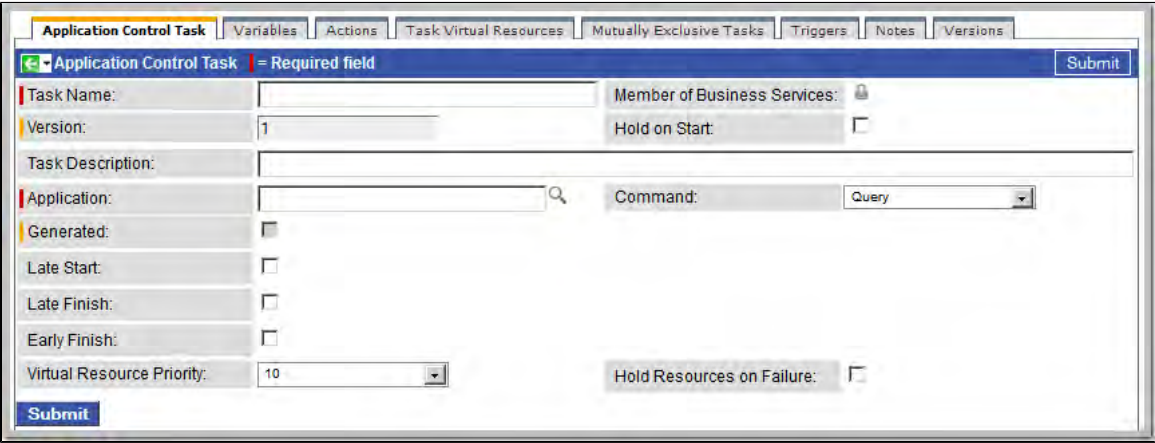
Built-In Variables

Several built-in variables are provided to pass information about the Application being monitored into the task or tasks being launched by the trigger. You can pass the information into the launched tasks by including the variables in a text field in the task definition. See [Application Monitor Variables](#) for details.

Creating a New Application Control Task

You can also manually create Application Control tasks. You might want to do so if you need to set different variables or conditions for different

uses of the same task.

Step 1	From the Navigation Pane, select Application Control Tasks . The Application Control Tasks List screen displays.
Step 2	Click New . The Application Control Task screen displays.
	
Step 3	Using the field descriptions provided below as a guide, complete the fields as needed.
Step 4	Click the Submit button to save the record and return to the menu, or, right-click on the title bar and select Save to save the record and remain on the current display.
Step 5	If appropriate, repeat these steps for any additional tasks you want to add.

Application Control Task Field Descriptions

The table below describes the fields, buttons, and tabs on the Application Control task definition and task instance screens. Most fields appear on both screens; however they do not always appear at the same spot. In the latter case, the table provides a field description at the location found on the task definition screen. Some fields appear only on one of the screens.

Field Name	Description
Task/Instance Name	Required. Name used within Opwise to identify this task. Up to 40 alphanumeric; variables supported. It is the user's responsibility to develop a workable naming scheme for tasks.
Summary	User-supplied description of this record.
Application	Required; protected if auto-generated. The name of the Application resource defined using the Application resource screen. The Application resource defines where the software application is running; it also defines the start, stop, and query commands for the application. Type in a name, or click the magnifying glass to browse to an existing Application resource record or create a new one.
Command	Required; protected if auto-generated. The command this task is executing against the software application. One of the following: <ul style="list-style-type: none"> • Query • Start • Stop
Hold on Start	If enabled, when the task is launched it appears in the Activity display with a status of Held. The task runs when the user releases it.

Member of Business Services	User-defined. Allows you to select one or more Business Services that this record definition belongs to. Click the lock icon to unlock the field and select Business Services .
Version	Task definition only; system-supplied. The version number of the current record, which is incremented by Opwise every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Late Start	If enabled, and if the task instance starts after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late start (see Late Start Type). To determine whether a task instance started late, open the task instance and locate the Started Late field; the field is checked if the instance started after the specified time. This field only appears on the task instance if the user specified a Late Start in the task definition.
Late Start Type	Required if Late Start is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it starts after the specified time. • Duration - Flag the task if it starts a certain amount of time after the programmed start time. The task must have a specific start time.
Late Start Time	Time after which the task start time is considered late. Use hh:mm, 24-hour time
Late Start Duration	Duration (amount of relative time) after which the task is considered to have started late. For a task within a workflow, the duration is the period between the time the workflow starts and the time the task itself starts. For example, a task might have a Late Start Duration of 60 minutes. If the workflow starts at 9:00 a.m. but the task itself does not start until 10:30, the task has started late. For a task that is not within a workflow, Late Start Duration has meaning only if the task has been held upon starting. For example, if a task has a Late Start Duration of 60 minutes and the Hold on Start field is enabled, if the task is not released from hold within the amount of time specified in the Late Start Duration field, the task has started late.
Started Late	Task instance only; system-supplied. This field is flagged if the task started later than the time specified in the Late Start fields.
Late Finish	If enabled, and if the task instance finishes after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late finish (see Late Finish Type). To determine whether a task instance finished late, open the task instance and locate the Finished Late field; the field is checked if the instance finished after the specified time or lasted longer than expected. This field only appears on the task instance if the user specified a Late Finish in the task definition.
Late Finish Type	Required if Late Finish is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it finishes after the specified time (see Late Finish Time). • Duration - Flag the task if it finishes a certain amount of time after the programmed finish time (see Late Finish Duration). The task must have a specific finish time.
Late Finish Time	If Late Finish Type is Time, use this to specify the time after which the task finish time is considered late. Use hh:mm, 24-hour time.
Late Finish Duration	If Late Finish Type is Duration, use this to specify the longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Finished Late	Task instance only; system-supplied. This field is flagged if the task finished later than the time or duration specified in the Late Finish fields.

Early Finish	If enabled, and if the task instance finishes before the time or period specified, the task instance is flagged as early. You can specify a time or duration to determine an early finish (see Early Finish Type). To determine whether a task instance finished early, open the task instance and locate the Finished Early field; the field is checked if the instance finished before the specified time or did not last as long as expected. This field only appears on the task instance if the user added Early Finish specifications to the task definition.
Early Finish Type	Required if Early Finish is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it finishes before the specified time (see Early Finish Time). • Duration - Flag the task if it finishes a certain amount of time before the programmed finish time (see Early Finish Duration). The task must have a specific finish time.
Early Finish Time	If Early Finish Type is Time, use this to specify the time before which the task finish time is considered early. That is, enter a time at which the task should still be running. Use hh:mm, 24-hour time.
Early Finish Duration	If Early Finish Type is Duration, use this to specify the shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Finished Early	Task instance only; system-supplied. This field is flagged if the task finished earlier than the time specified in the Early Finish fields.
First Time Ran	Task definition only; system-supplied. Displays after the first time the task runs. The date and time this task first ran.
Lowest Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. The shortest amount of time this task has taken to run.
Last Time Ran	Task definition only; system-supplied. Displays after the first time the task runs. The most recent date and time the task ran.
Average Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. Shows the average amount of time this task takes to run.
Number of Instances	Task definition only; system-supplied. Displays after the first time the task runs. Shows the number of times this task has run.
Highest Instance Time	Task definition only; system-supplied. Displays after the first time the task runs. The longest amount of time this task has taken to run.
Virtual Resource Priority	Priority for acquiring a resource when two or more tasks are waiting for the resource. This priority applies to all resources required by the task. Options: 1 (high) - 20 (low). Default is 10.
Hold Resources on Failure	If enabled, the task instance will continue to hold Renewable resources if the task instance fails. Renewable resources will be returned only if the task instance status is either Complete, Finished, or Skipped.
Generated	System-supplied; protected. If selected, indicates that this Application Control task was generated automatically when the Application resource record was submitted.

Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Launch Task button	Manually launches the task.
View Instances button	Displays a list of task instances for which there has been a status change or a modification to the task instance record within the last 30 days (an Updated on Last 30 Days filter has been pre-selected for the list).
Delete button	Deletes the current record.
View Parent button	Task instance only; Displays this task's parent task (workflow), if any.
Show Details button	Task instance only; displays detailed information about this task instance.
Skip button	Task instance only; for tasks loaded into the schedule that haven't run yet. Allows you to tell Opwise to skip this task. See Skipping a Task .
Hold button	Task instance only; see Putting a Task on Hold .
Force Finish button	Task instance only; see Force Finishing a Task .
Output tab	Task instance only. Displays output generated from the process, if any, based on specifications provided by the user in the Automatic Output Retrieval fields in the task definition.
Variables tab	Displays all variables associated with this record.
Actions tab	<p>Allows you to specify actions that Opwise will take automatically based on events that occur during the execution of this task. Events are task instance status, exit codes, late start, late finish, and early finish. Actions are:</p> <ul style="list-style-type: none"> • Abort Action - Abort the task if certain events occur. For details, see Setting Abort Actions. • Email Notification - Send an email if certain events occur. For details, see Creating Email Notifications. • Set Variable - Used in workflows to set a variable based on the occurrence of certain events. For details, see Setting Variables within a Workflow. • SNMP Notification - Send an email if certain events occur. For details, see Creating SNMP Notifications. • System Operation - Run an Opwise system operation based on specified conditions. For details, see Setting System Operations.
Task Virtual Resources tab	Lists Virtual Resources to which this task is assigned.
Mutually Exclusive Tasks tab	Displays all tasks that have been set to be mutually exclusive of this task.

Triggers tab	Displays a list of all triggers that have been defined to launch this task. Also allows you to add new triggers. If you add a new trigger from this location, Opswise automatically constructs a default trigger name as follows: <current task name>#TRIGGER#. You can change the default name if desired. For instructions on creating triggers, see Creating Triggers .
Notes tab	Displays all notes associated with this task.
Versions Tab	Stores copies of all previous versions of the current record. See Record Versioning .

Specifying When a Task Runs

You can run the task as part of a [workflow](#), [specify triggers](#) that run the task automatically based on times or events, or [run the task manually](#).

Monitoring Task Execution

You can monitor all system activity from the [Activity screen](#) and can view activity history from the [Activity History screen](#).

Application Monitor Trigger

- [Overview](#)
- [Built-In Variables](#)
- [Creating a New Application Monitor Trigger](#)
- [Application Monitor Trigger Field Descriptions](#)

Overview

The Application Monitor Trigger allows you to trigger one or more tasks based on the status of:

- A specific [application resource](#).
- One or more [application resources](#), based on selection criteria you supply.

You can launch any number of tasks when the conditions in the trigger are satisfied.

If you specify Application Monitor Condition = ALL, and select all Application types, the trigger monitors all Application resource records you have defined. Any time any one of them goes to any of the Statuses you specified in the Status(es) fields, the trigger launches the task(s) specified in the Task(s) field. For example, you might use this trigger to send an email notification to technical support if any of the monitored applications goes into a failure status.

Built-In Variables

[Application Monitor built-in variables](#) are provided to pass information about the Application being monitored into the task(s) being launched by the trigger. You can pass the information into the launched tasks by including the variables in a text field in the task definition.

Creating a New Application Monitor Trigger

Step 1	From the navigation pane, select Automation Center > Triggers > Application Monitor Triggers . The Application Monitor Triggers List screen displays.
Step 2	Click New . The Application Monitor Trigger Definition screen displays. <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> </div>
Step 3	Using the field descriptions provided below as a guide, complete the fields as needed.
Step 4	Click the Submit button to save the record and return to the menu, or right-click the title bar and select Save to save the record and remain on the current display.
Step 5	If appropriate, repeat these steps for any additional triggers you want to add.
Step 6	Enable the trigger(s). Once the trigger is enabled, it monitors the Application resource(s) specified in the Monitoring Type fields.

Application Monitor Trigger Field Descriptions

Field Name	Description
Trigger Name	Required. Name used within Opwise to identify this trigger. It can contain a maximum of 40 alphanumeric characters. It is the user's responsibility to develop a workable naming scheme for triggers.
Enabled	User-defined. Whether or not the Enabled field is checked. The user enables and disables the trigger by clicking the Enable/Disable Trigger buttons. Only enabled triggers are processed by Opwise.
Task(s)	Required. Name of the task(s) being triggered when this trigger is satisfied. When selecting tasks from the definition screen, click on the lock icon to unlock the field and select tasks .
Enabled By	System-supplied. Displays the ID of the user who most recently enabled this trigger.
Member of Business Services	User-defined. Allows you to select one or more Business Services that this record definition belongs to. Click the lock icon to unlock the field and select Business Services .
Calendar	If Special Restriction is selected, the calendar defines the Holidays or Non Business days. Enter a calendar name or click the magnifying glass icon either to browse for an existing calendar or to add a new calendar. To display details about the calendar specified in this field, hover over the paper icon.
Skip Count	User-defined. Allows you to specify that Opwise should skip the next <i>N</i> times this task is triggered.
Version	System-supplied. The version number of the current record, which is incremented by the system every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Skip Trigger if Active	User-defined. Allows you to specify that Opwise should skip the next run of the specified task(s) if the previous run has not gone to a Complete status (that is, it is still active).
Description	User-defined. Copied from the Description field in the trigger.
Status(es)	System-supplied. The application status being monitored for. One or more of the following: <ul style="list-style-type: none"> • Inactive - The initial state of the Application. The Application is stopped and unmonitored. • Start Failure - The agent experienced a failure while attempting to execute the Start command. • Starting - The Start command was executed and Opwise is waiting for Query command response. • Active - The Query command response is reporting that the Application is Active. • Impaired - The Query command response is reporting that the Application is experiencing a problem and is possibly down. • Query Overdue - The agent is late sending Opwise an updated Query command response.
Monitoring Type	Indicates whether you are monitoring one specific Application resource or want to provide selection parameters to monitor multiple Application resources. See Applications for information about setting up Application resources. Options: <ul style="list-style-type: none"> • Specific Application - Use the Application field below to browse for and select the Application resource you want to monitor. • General Applications - Use the Application Monitor Condition and Application Type(s) fields to provide parameters for selecting which Application resources you want to monitor.

Application	Required if Monitoring Type = Specific Application. Name of a specific application resource to be monitored.
Application Monitor Condition	<p>If Monitoring Type = General Application(s), allows you to specify selection parameters, as follows:</p> <ul style="list-style-type: none"> • ALL - Monitor all Application resources. • Starts With - Monitor all Application resources whose name starts with the string you provide in the Condition Value field. • Contains - Monitor all Application resources whose name contains the string you provide in the Condition Value field. • Ends With - Monitor all Application resources whose name ends with the string you provide in the Condition Value field.
Condition Value	If Application Monitor Condition = Starts With, Contains, or Ends With, use this field to specify the search string.
Application Type(s)	<p>If Monitoring Type = General Application(s), type(s) of applications to monitor. Options:</p> <ul style="list-style-type: none"> • Windows Service • Linux/Unix Daemon • z/OS Started Task
Special Restriction	<p>Enable this field in order to specify additional parameters that tell Opwise how to handle exceptions such as when the trigger is satisfied on a holiday or non-business day. You can specify Simple and/or Complex Restrictions (see field descriptions below for details). For example, you can specify a Simple Restriction that disables the trigger if it is satisfied on a holiday identified in the calendar and/or a Complex Restriction that disables the trigger on the last business day of every month.</p>
Simple Restriction	If enabled, allows you to specify an action (see Action field, below) such as Do Not Trigger on a non-business day or holiday (see Situation field, below). For example, do not trigger on a non business day.
Situation	<p>If Simple Restriction is enabled, allows you to select the situation that causes the system to initiate the action specified in the Action field (see Action field below).</p> <p>Options:</p> <ul style="list-style-type: none"> • On Non Business Day • On Holiday
Action	<p>If Special Restriction is enabled, allows you to select an action to take on a non business day or holiday (see Situation field above).</p> <p>Options:</p> <ul style="list-style-type: none"> • Do Not Trigger • Next Day (run on the next day) • Next Business Day (run on the next business day, as defined in the calendar) • Previous Day (run on the previous day) • Previous Business Day (run on the previous business day, as defined in the calendar)
Complex Restriction	<p>If enabled, allows you to specify a set of parameters that determine one or more situations when this trigger should not be satisfied. Used in conjunction with the following fields: Restriction Mode, Restriction Adjective, Restriction Noun, Restriction Qualifier (see details below). For example, you may specify that you do not want to satisfy this trigger on the last business day of the year or the first day of each month.</p>
Restriction Mode	<p>If both Simple and Complex Restriction are enabled, specifies whether you want to use both restriction types (AND) or one or the other (OR).</p> <p>Options:</p> <ul style="list-style-type: none"> • And • Or

<p>Restriction Adjective</p>	<p>If Complex Restriction is enabled, the type of selection.</p> <p>Options:</p> <ul style="list-style-type: none"> • 1st • 2nd • 3rd • 4th • Last <p>Example: The last business day of the month.</p>
<p>Restriction Noun</p>	<p>If Complex Restriction is enabled, the day you want to select.</p> <p>Options:</p> <ul style="list-style-type: none"> • Sunday through Saturday • Day • Business Day • Custom day (see Creating Custom Days) <p>Example: The last business day of the month.</p>
<p>Restriction Qualifier</p>	<p>If Complex Restriction is enabled, the period you are selecting from.</p> <p>Options:</p> <ul style="list-style-type: none"> • Month • Year • January through December • Custom period (see Creating Custom Days) <p>Example: The last quarter of the year.</p>
<p>Submit button</p>	<p>Submits the new record to the database.</p>
<p>Update button</p>	<p>Saves updates to the record.</p>
<p>Enable Trigger button</p>	<p>Activates this trigger and writes your UserID to the Enabled By field.</p>
<p>Disable Trigger button</p>	<p>Deactivates this trigger.</p>
<p>Trigger Now button</p>	<p>Immediately triggers all the tasks specified in this trigger.</p>
<p>Delete button</p>	<p>Deletes the current record.</p>
<p>Variables tab</p>	<p>Displays all variables associated with this record.</p>
<p>Versions Tab</p>	<p>Stores copies of all previous versions of the current record. See Record Versioning.</p>

Reports



Reports Information

[*Overview*](#)

[*Report Permissions*](#)

[*Viewing a Report*](#)

[*Creating a Report*](#)



Report Actions

[*Running a Report*](#)

[*Updating a Report*](#)

[*Saving a Report*](#)

[*Inserting a Report*](#)

[*Deleting a Report*](#)

[*Publishing a Report*](#)

[*Printing a Report*](#)

[*Refreshing a Report*](#)



Gauges

[*Creating a Gauge from a Report*](#)



Scheduling

[*Scheduling Automatic Refresh*](#)

[*Report Scheduling Permissions*](#)

[*Setting Up the Email Server*](#)

[*Scheduling the Report*](#)



Reports Tables

[*Opswise Report Tables*](#)

[*Opswise Activity Tables*](#)



The information on these pages also is located in the [Opswise Automation Center 5.1.1 User Guide.pdf](#).

Opswise Automation Center Reports

- [Overview](#)
- [Report Permissions](#)
- [Viewing an Existing Report](#)
- [Report Field Descriptions](#)
- [Predefined Reports Categories](#)
- [Creating a New Report](#)
- [Performing an Action on a Report](#)
- [Creating a Gauge from a Report](#)
- [Scheduling Automatic Report Distribution](#)
 - [Report Scheduling Permissions](#)
 - [Setting Up the Email Server](#)
 - [Scheduling the Report](#)
 - [Scheduled Email of Report Field Descriptions](#)
- [Opswise Report Tables](#)
- [Opswise Activity Table \(ops_exec\)](#)

Overview

Opswise provides a number of [predefined reports](#). These reports are identified in the **Global reports** section of the [Reports List](#) screen.

You also can [create](#), [save](#), and [run](#) your own reports. If you create a report that is visible for everyone to see, it also is listed in the **Global reports** section.

Every report, whether pre-defined or user-defined, contains data from a single [Opswise Report table](#). When you create a report, you select the table containing the data that you want to include in the report. The Report table that you select for a report determines under which category in the **Global reports** section it is listed. If a category does not exist for the new report, Opswise creates one.

The [Activity Screen](#) uses Activity reports - a category of [Global reports](#) - to define the task instances that it displays. All Activity reports contain data from the [Activity table](#) (`ops_exec`). If you want to [create a new Activity report](#), you must select the Activity table for that report. User-defined Activity reports, along with the pre-defined Activity reports, automatically are listed on the drop-down menu of reports on the Activity screen.

All user-defined reports, including Activity reports, automatically are listed on the Reports List screen.

Report Permissions

Access to information in reports depends on the permissions assigned to the user and whether the:

- Report is generated and displayed on the Reports screen (or created as a [gauge](#)).
- Report is [published](#) and accessed through the published URL.

All unpublished reports, whether list reports or non-list reports (such as pie charts and calendars), and published list reports display only those records that the user has permission to read.

Published non-list reports display all records in the report, regardless of the user permissions (see [Roles and Permissions](#) for information).

For information on permissions associated with scheduling the email of reports, see [Report Scheduling Permissions](#), below.

Viewing an Existing Report

Step 1 From the navigation pane, select **Automation Center > Reports**. The Reports List screen displays a list of all existing reports (predefined and user-created).

The **Global reports** section identifies all of the predefined reports. Each [category of predefined reports](#) is based on the name of the

Opswise Reports table that is used to build the reports in that category.

Reports

My Saved reports New

No reports saved for Administrator

My Groups' reports

No reports saved for any of Administrator 's groups

Global reports

Agent

Agent Connection Status

All Agents

Application

Application Status

Connector

All Connectors

Activity

Active Task Instances

Active Workflow Task Instances

Active/Late Task Instances

All Task Instances by Status

Cancelled Task Instances

Held Task Instances

In Doubt Task Instances

Queued Task Instances

Running Task Instances

Task Instances Due to Finish in the Next 3 Hours

Task Instances Due to Finish in the Next Hour

Today's Failed Task Instances by Status

Today's Successful Task Instances

Today's Task Instances by Created Time

Today's Task Instances by Type

Undeliverable Task Instances

Unsuccessful Task Instances

Waiting For Resources Task Instances

Waiting Task Instances

Workflow Task Instances

Workflow Task Instances with Problems

History

Duration

Outstanding Request

Virtual Resources - Outstanding Requests By Task

Virtual Resources - Outstanding Requests By Virtual Resource

Currently In Use By

Virtual Resources - Currently In Use By Task

Virtual Resources - Currently In Use By Virtual Resource

Forecast

Forecast - All - Calendar

Forecast - All - List

Forecast - All - List with Agent

Forecast - All - List with Agent Cluster

Forecast - All - List with Run Criteria Evaluation

Forecast - All - List with Target Server

Forecast - All - List without Skipped

Forecast - Calendar

Forecast - Daytime

Forecast - Overnight

Forecast - Today

Forecast - Weekend

User

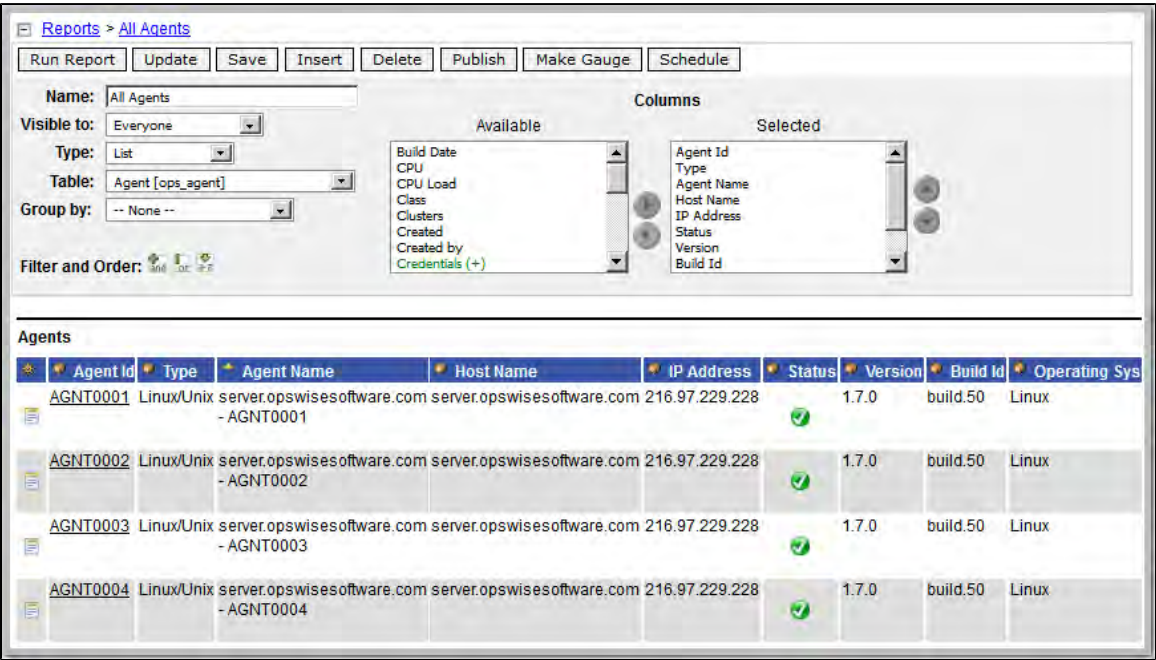
Users

Group Member

Group Membership

User Role

Step 2 Click the report that you want to view. The Report screen for that report displays (the following example shows the All Agents report).



The screenshot shows the 'All Agents' report configuration interface. At the top, there are buttons for 'Run Report', 'Update', 'Save', 'Insert', 'Delete', 'Publish', 'Make Gauge', and 'Schedule'. Below these are configuration fields: 'Name' (All Agents), 'Visible to' (Everyone), 'Type' (List), 'Table' (Agent [ops_agent]), and 'Group by' (-- None --). A 'Columns' section shows 'Available' and 'Selected' columns. The 'Available' column includes Build Date, CPU, CPU Load, Class, Clusters, Created, Created by, and Credentials (+). The 'Selected' column includes Agent Id, Type, Agent Name, Host Name, IP Address, Status, Version, Build Id, and Operating Sys. Below the configuration is a table of agents:

Agent Id	Type	Agent Name	Host Name	IP Address	Status	Version	Build Id	Operating Sys
AGNT0001	Linux/Unix	server.opswisesoftware.com - AGNT0001	server.opswisesoftware.com	216.97.229.228	✓	1.7.0	build.50	Linux
AGNT0002	Linux/Unix	server.opswisesoftware.com - AGNT0002	server.opswisesoftware.com	216.97.229.228	✓	1.7.0	build.50	Linux
AGNT0003	Linux/Unix	server.opswisesoftware.com - AGNT0003	server.opswisesoftware.com	216.97.229.228	✓	1.7.0	build.50	Linux
AGNT0004	Linux/Unix	server.opswisesoftware.com - AGNT0004	server.opswisesoftware.com	216.97.229.228	✓	1.7.0	build.50	Linux

- The upper section of the page displays the fields used to define the report (see [#Report Field Descriptions], below).
- The lower section of the page displays the current report as defined by the fields in the upper section.

The tabs at the top of the upper section let you perform specific actions on the report (see [Performing an Action on a Report](#)).

Report Field Descriptions

Field Name	Description
Name	Name of this report.
Visible to:	Defines who has access to this report. Options: <ul style="list-style-type: none"> • Me - Only the user who created the report can access it. • Everyone - All users. To assign the "Visible to" field to Everyone, you must have the report_global role. • Other group names - All users that are a member of the specified Security group. To assign the "Visible to" field to a Security group, you must have the report_group role.
Type	Format of the report. The option you select in this field determines what additional fields display in the second column of the report screen. Options: <ul style="list-style-type: none"> • Pie chart - Circular graph, useful when comparing sections. • Bar chart - Chart of color-coded bars. Can be displayed horizontally or vertically. • List - List of items that match the report parameters. • Box chart - Graphical representation of groups of numerical data via five-number summaries. • Calendar - For date-based events: displays a calendar showing events that match the report parameters. • Control chart - Chart used to determine whether a given process is in a state of statistical control. • Histogram - Used in statistics, a histogram uses bars to show the distribution of data. • Horizontal bar - See Bar Chart, above. • Line chart - Graph showing information as a series of data points. • Pareto chart - Combination chart showing bars (individual values) and a line graph (cumulative totals) • Trend chart - Used to show trends in data over time. • Trendbox chart - Box chart trended by a secondary value over a period. • Vertical bar - See Bar Chart, above.
Table	Opwise table being used to create the report.

Group by	Field within the table that will be used to sort the report.
Stacked Field	For bar charts: allows you to specify a second field for displaying additional color-coded information. For example, you might display a bar chart showing today's system activity sorted by agent, additionally sorted by color-coded status.
Sum Field	For bar charts, control charts, histograms, line charts, trend charts, and trendbox charts: allows you to select a field into which task instance data will be summarized.
Chart size	For bar charts, Pareto charts, and trend charts, allows you to specify the chart size.
Other threshold	For bar charts, Pareto charts, and trend charts: allows you to limit the number of bars in your chart.
Columns (Available and Selected)	For lists of data: specifies fields from the table that should appear as columns in the list report. if you leave the Selected section blank, Opswise uses a hard-coded default selection.
Measured Field	For box charts: allows you to choose a field from the record that is used as one measurement of the data.
Calendar Field	For Calendars: allows you to choose a field from the table to be displayed on one or more calendar dates.
Trend Field	For control charts, histograms, line charts, trend charts, and trendbox charts: allows you to specify the field being monitored and the period of the trend.
Filter and Order	Allows you to create selection parameters for the report.
Run Report button	Generates the report and displays the results on the current screen. (Clicking Run Report does not save any new data entered for the report.)
Update button	For existing reports: Saves changes made to the report and generates the report, but does not display the report; the Reports List screen re-displays. (Compare with Save button.)
Save button	Saves (for new reports: adds it to the Reports list), generates, and displays the report.
Insert button	Copies the current report and adds it to your Reports List. Make sure you enter a new name in the Name field before clicking this button or it will overwrite the current report.
Delete button	Deletes the current report.
View Published URL button	For published reports only: Displays the URL of the published report.
Unpublish button	For published reports only: Deletes the URL of the published report.
Publish button	Creates a URL for the current report, which you can copy and paste into your browser.
Make Gauge button	Creates a gauge using the report parameters. You can then import the gauge into your home page or dashboard.
Schedule button	Opens a page that allows you to schedule the generation and distribution of this report .

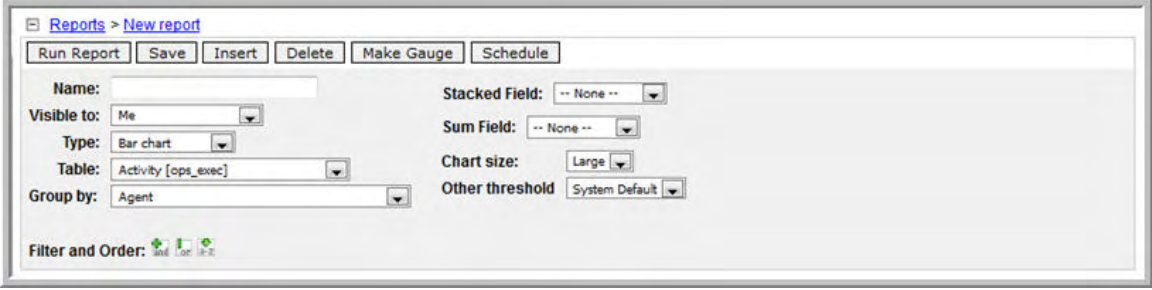
Predefined Reports Categories

The following table identifies the [Opswise report table](#) that is used to build the reports in each category of [predefined reports](#).

Category	Opswise Report Table
Agent	ops_agent
Application	ops_application
Connector	ops_connector
Activity	ops_exec
History	ops_history
Outstanding Request	ops_resource_order
Currently In Use By	ops_resource_usage
Forecast	ops_trigger_forecast
User	sys_user

Group Member	sys_user_grmember
User Role	sys_user_has_role


Creating a New Report



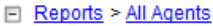
Step 1	From the navigation pane, select Automation Center > Reports . The Report List screen displays.
Step 2	Click New . The New Report screen displays. 
Step 3	Using the report field descriptions as a guide, define the report as desired.
Step 4	Select an action to perform on the report. Depending on the action that you select, the report may be saved on the Reports list (see Performing an Action on a Report , below).

Performing an Action on a Report

The following table identifies the basic actions that you can perform on any report that you create or select from the [Reports List](#) screen.

(For additional actions, see [Creating a Gauge from a Report](#) and [Scheduling Automatic Report Distribution](#).)

Running a Report	<ol style="list-style-type: none"> Using the report field descriptions as a guide, change any report definition fields in the upper section of the screen. Click Run Report. Opswise generates a report based on the report definition and displays the report, but does not save changes made to the report definition. If the report is new, it is not saved and does not appear on the Reports list.
Updating a Report	<ol style="list-style-type: none"> Using the report field descriptions as a guide, change any report definition fields in the upper section of the screen. Click Update. Opswise saves changes made to the report definition and generates a report based on the report definition. The report is not displayed; instead, the Reports List screen re-displays. If the report is new, it is saved and appears on the Reports list.
Saving a Report	<ol style="list-style-type: none"> Using the report field descriptions as a guide, change any report definition fields in the upper section of the screen. Click Save. Opswise saves changes made to the report definition, generates a report based on the report definition, and displays the report. If the report is new, it is saved and appears on the Reports list.
Inserting a Report	<ol style="list-style-type: none"> Enter a new Name for the report. As desired, and using the report field descriptions as a guide, change any other report definition fields in the upper section of the screen. Click Insert. Opswise creates a copy of the report and inserts it on the Reports List. If the report is new, it is saved and appears on the Reports list.
Deleting a Report	<div style="background-color: #ffe6e6; padding: 10px; margin-bottom: 10px;">  Warning Do not delete any of the predefined reports. Opswise utilizes some of these reports in other areas. </div> <ul style="list-style-type: none"> Click Delete. Opswise removes the report from the Reports List.

Publishing a Report	<div data-bbox="402 157 1495 262" style="background-color: #ffffcc; padding: 5px;">  Note This action is available for existing reports only. </div> <ol style="list-style-type: none"> Click Publish. The following then appears at the top of the page: <ul style="list-style-type: none"> URL of the report generated by Opwise. Two Published Report buttons: View Published URL and Unpublish (for a description of these buttons Descriptions, above). <div data-bbox="521 411 1495 577" style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p> Use the following URL for the public report: http://server.opswissoftware.com:8080/opwise/sys_report_display.do?sysparm_report_id=3ccdf728c0a8026701f04af69231a124</p> <p> Reports > All Agents</p> <p> <input type="button" value="Run Report"/> <input type="button" value="Update"/> <input type="button" value="Save"/> <input type="button" value="Insert"/> <input type="button" value="Delete"/> <input type="button" value="View Published URL"/> <input type="button" value="Unpublish"/> <input type="button" value="Make Gauge"/> </p> </div> You then can: <ul style="list-style-type: none"> Copy and send the URL via email or other method. Open a new tab in your browser and paste the URL. You then can bookmark the page or save it as a text file.
Printing a Report	<ol style="list-style-type: none"> Click the printer icon in the top right corner of the title bar. A preview screen generates. Click Click to Print to print the report.
Refreshing a Report	<ul style="list-style-type: none"> Click Run Report, Update, Save, or Insert to refresh the data on a currently displayed report. For example, if the report is being displayed and a new Agent is added to the Opwise system, you can click Run Report to see that new report.

Creating a Gauge from a Report

A gauge is a "live" report, which you can display on your home page and/or dashboard, whose information is updated automatically according to the refresh setting on each of those pages.

You can create a gauge from any report and then add the gauge to your home page and/or dashboard:

Step 1	Display the report from which you want to create a gauge.
Step 2	Click Make Gauge . The gauge is then listed on the Gauges List screen .
Step 3	See Using the Home Page and Using the Dashboard for information about adding the gauge to your home page and dashboard.



Note

If you delete a report from which a gauge was created, the gauge itself is not deleted as well. To delete the gauge, you must do so from the [Gauges List screen](#).

Scheduling Automatic Report Distribution

You can schedule a report to be run and distributed to an email list automatically on specific dates and times.

The data contained in a scheduled report contains only those records that the user who scheduled the report has permission to read.

Report Scheduling Permissions

Only users assigned the **report_scheduler** role (see [Assigning Permissions to Users or Groups](#)) are provided with the **Scheduled Report Emails** link in the navigation pane and the **Schedule** button on any Reports page, which enables them to [create scheduled reports](#) and update or

delete existing scheduled reports.

Scheduled reports are run as the user that scheduled the report, as indicated in the **Run as** field on the Scheduled Email of Reports screen. However, any updates to a report schedule definition by a user assigned the **report_schedule** role will cause the **Run_as** field to inherit the user's **sys_id**. From then on, the scheduled report will run under the privileges of that user.

Setting Up the Email Server

Before you can schedule automatic report distribution, you must first set up the email server that you want to use.

Step 1	From the navigation pane, select Automation Center Administration > Configuration > Report Email Properties .
Step 2	Using the field descriptions provided on the screen, complete the fields as needed and click Save .

Email Properties
Save

Please edit your changes and press save

SMTP Server settings

Outgoing (SMTP) mail server. SMTP requires port 25. Leave this value blank to disable email.

Outgoing mail server address (e.g. automation.center@company.com).

Outgoing (SMTP) mail server password.

Outgoing email display name (e.g. "Automation Center").

Automation Center

Content and Security

Send emails in HTML format

Yes | No

Requires authorization

Yes | No

Issue STARTTLS command

Yes | No

Advanced Settings

Email address to which all emails will be sent. Used for non-production systems for testing purposes.

Save

Scheduling the Report

Step 1 From the navigation pane, either:








1. Select **Automation Center > Reports**, click the name of the report that you want to schedule, and on the Reports page click the **Schedule** button.
2. Select **Automation Center > Scheduled Report Emails**, click **New**, and on the Scheduled Email of Report page browse for the report that you want to schedule in the Report field.

Step 2 Use the field descriptions below for guidance to complete the fields.

Step 3 Click **Submit**.

Scheduled Email of Report Field Descriptions

Field Name	Description
Name	Name of the report schedule.
Active	Indicates whether the schedule is active or not. Only active schedules are executed.
Report	Name of the report being scheduled. Click the magnifying glass to browse for a report.
Every	Specification for when the report should be run and distributed. Options: <ul style="list-style-type: none"> • Day • Week • Month
Users	Opwise users that will receive this report. See Adding Users .
Day (of week)	If Every = Week, specify the day of the week when the report should be run and distributed. Options: Monday through Sunday.
Day (of month)	If Every = Month, specify the numeric day of the month when the report should be run and distributed. Options: 1 through 31.
Time	In conjunction with the Every field, specifies a time when the report should be run and distributed.
Groups	Opwise groups that will receive this report. See Adding Groups .
Email Addresses	Comma-separated list of email addresses for those receiving this report.
Subject	Subject line of the email.

Introductory Message	Message that will go in the body of the email. You can format the message using the standard formatting options provided. You can also specify font color  , background color  , horizontal rule  , web link  , images  , tables  . You can also toggle between wysiwyg and html source  .
Type	Output type for the report. Options: PDF landscape, PDF, Excel, CSV (comma-separated values).
Include detail	For PDF list reports, includes any additional details available for the record type(s).

Opwise Report Tables

The following table identifies and describes the tables that you can use when creating reports.

Table Name	Table ID	Description
Abort Action	ops_abort_action	Contains details about Abort actions .
Abort Action	ops_abort_action_v	Contains details about previous versions of Abort actions . New versions of Abort Action records are created when a task record is updated.
Action	ops_notification	Contains details about all actions on the Actions menu (Abort actions , Email notifications , SNMP notifications , and Set Variable actions).
Action	ops_notification_v	Contains details about previous versions of all actions on the Actions menu (Abort actions , Email notifications , SNMP notifications , and Set Variable actions). New versions of Action records are created when a task record is updated.
Activity	ops_exec	Task instance activity (running tasks).
Agent	ops_agent	Displays a list of Agents .
Agent Cluster	ops_agent_cluster	Contains details about Agent Clusters .
Agent Cluster Version	ops_agent_cluster_v	Contains details about previous versions of Agent Clusters .
Agent Clusters	ops_bundle_agent_cluster_join	Shows relationship information between Agent Clusters and Bundles , that is, which agent clusters belong to which bundles.
Agent Mapping	ops_agent_mapping	Shows all the agents connected to one or more Promotion Targets (as retrieved using the Refresh Target Agents button).
Agents In Cluster	ops_unix_agent_cluster_join	Shows relationship information between Unix agents and Unix agent clusters , that is, which agents belong to which clusters.
Agents In Cluster	ops_unix_agent_cluster_join_v	Shows previous versions of relationship information between Unix agents and Unix agent clusters .
Agents In Cluster	ops_windows_agent_cluster_join	Shows relationship information between Windows agents and Windows agent clusters , that is, which agents belong to which clusters.
Agents In Cluster	ops_windows_agent_cluster_join_v	Shows previous versions of relationship information between Windows agents and Windows agent clusters .
Application	ops_application	Shows a list of Application resources .
Application Control Task	ops_task_application_control	Contains details about Application Control tasks .
Application Control Task Instance	ops_exec_application_control	Contains details about Application Control task instances .
Application Control Task Version	ops_task_application_control_v	Shows previous versions of Application Control tasks .
Application Monitor Trigger	ops_trigger_appl_monitor	Contains details about Application Monitor triggers .

Application Monitor Trigger Version	ops_trigger_appl_monitor_v	Contains details about previous versions of Application Monitor triggers.
Application Version	ops_application_v	Contains details about previous versions of Application resources.
Applications	ops_bundle_application_join	Shows relationship information between Application resources and Bundles , that is, which Application resources belong to which bundles.
Audit Record	ops_audit	Contains details of events being written to the Audit history .
Backup	ops_backup	Contains Backup and Purge records.
Bundle	ops_bundle	Contains all Bundles records.
Business Service	ops_generic_group	Contains details about Business Services .
Business Service Version	ops_generic_group_v	Contains previous versions of Business Service records.
Business Services	ops_bundle_generic_group_join	Contains relationship information between Business Services and Bundles ; that is, which Business Services belong to which Bundles.
Calendar	ops_calendar	Contains details about Calendar records .
Calendar Custom Days	ops_cal_cust_join	Contains details about which Custom Days are associated with which Calendar records .
Calendar Custom Days	ops_cal_cust_join_v	Contains previous versions of the association between Custom Days and Calendar records .
Calendar Version	ops_calendar_v	Contains previous versions of Calendar records .
Calendars	ops_bundle_calendar_join	Contains relationship information between Calendars and Bundles ; that is, which Calendars belong to which Bundles.
Cluster Nodes	ops_cluster_node	Provides details about cluster nodes (Controllers).
Cluster Notification	ops_cluster_notification	Contains Email and SNMP notification records associated with the cluster node.
Connector	ops_connector	Contains Connector records.
Connector Notification	ops_connector_notifications	Contains notification records associated with Connectors .
Credentials	ops_bundle_credentials_join	Contains relationship information between Credentials and Bundles , that is, which Credential records belong to which bundles.
Credentials	ops_credentials	Login credentials used by Opwise to access remote machines.
Credentials Version	ops_credentials_v	Contains previous versions of Credentials records.
Cron Trigger	ops_trigger_cron	Contains details about Cron trigger records.
Cron Trigger Version	ops_trigger_cron_v	Contains previous versions of Cron trigger records.
Currently In Use By	ops_resource_usage	Contains details about Virtual resource usage, as displayed in the Currently In Use By tab.
Custom Day Version	ops_custom_day	Contains previous versions of Custom Days records.
Custom Days	ops_custom_day	Contains details about defined Custom Days .
Custom Days	ops_bundle_custom_day_join	Contains relationship information between Custom Days and Bundles ; that is, which Custom Days belong to which Bundles.
Database Connection	ops_database_connection	Contains details about Database Connections defined in the Opwise database.

Database Connection Version	ops_database_connection_v	Contains previous versions of Database Connections records.
Database Connections	ops_bundle_db_cntn_join	Contains information about the relationship between Database Connections and Bundles , that is, which Database Connections belong to which Bundles.
Email Connection	ops_connect_email	Contains details about Email Connection resources.
Email Connection Version	ops_email_connection_v	Contains previous versions of Email Connection records.
Email Connections	ops_bundle_email_cntn_join	Contains information about the relationship between Email Connection and Bundles , that is, which Email Connections belong to which Bundles.
Email Notification	ops_email_cluster_notification	Contains Email Notification records associated with a Cluster Node .
Email Notification	ops_email_conn_notification	Contains Email Notification records associated with a Connector .
Email Notification	ops_email_notification	Contains details about Email Notifications associated with tasks.
Email Notification	ops_email_notification_v	Contains previous versions of Email Notifications associated with tasks. Note that a new version is created only when the task is updated.
Email Task	ops_task_email	Contains details about Email tasks .
Email Task Instance	ops_exec_email	Contains details about Email task instances .
Email Task Version	ops_task_email_v	Contains previous versions of Email task records.
Email Template	ops_email_template	Contains details about Email templates .
Email Template Version	ops_email_template_v	Contains previous versions of Email templates records.
Email Templates	ops_bundle_email_tmplt_join	Contains relationship information between Email templates and Bundles , that is, which Email Templates belong to which Bundles.
File Monitor	ops_task_file_monitor	Contains details about File Monitor tasks.
File Monitor Instance	ops_exec_file_monitor	Contains details about File Monitor task instances.
File Monitor Trigger	ops_trigger_fm	Contains details about File Monitor triggers.
File Monitor Trigger Version	ops_trigger_fm_v	Contains previous versions of File Monitor trigger records.
File Monitor Version	ops_task_file_monitor_v	Contains previous versions of File Monitor task records.
File Transfer Task	ops_task_ftp	Contains details about File Transfer tasks.
File Transfer Task Instance	ops_exec_ftp	Contains details about File Transfer task instances.
File Transfer Task Version	ops_task_ftp_v	Contains previous versions of File transfer task records.
Forecasts	ops_trigger_forecast	Contains details about trigger forecasts .
FTP File Monitor	ops_task_ftp_file_monitor	Contains details about FTP File Monitor tasks.

FTP File Monitor Instance	ops_exec_ftp_file_monitor	Contains details about FTP File Monitor task instances.
FTP File Monitor Version	ops_task_ftp_file_monitor_v	Contains previous versions of FTP File Monitor task records.
Group	sys_user_group)	Contains details about Opwise Security Groups .
Group Member	sys_user_grmember	Contains relationship information between Opwise Security Groups and Opwise Users ; that is, which Users belong to which Groups .
Group Role	sys_group_has_role	Contains relationship information between Opwise Security Groups and Roles ; that is, which Groups have been assigned which Roles .
History	ops_history	Contains a history of task activity.
Indesca Agent	ops_indesca_agent	Contains details about Indesca agents .
Indesca Agent Mapping	ops_agent_mapping_indesca	Shows the Agent mapping specifications between local Indesca agents and Indesca agents on a Promotion Target (as retrieved using the Refresh Target Agents button).
Indesca Task	ops_task_indesca	Contains details about Indesca tasks .
Indesca Task Instance	ops_exec_indesca	Contains details about Indesca task instances .
Indesca Task Version	ops_task_indesca_v	Contains previous versions of Indesca task records.
JobStep Files Data	ops_exec_zos_files	Contains details about jobsteps in a z/OS task.
Label	label	Contains details about currently displayed navigation pane labels .
Label entry	label_entry	Contains details about all navigation pane labels , including those that have been deleted.
Label history	label_history	Contains the event history for navigation pane label records, including creation, updates, and deletions.
Linux/Unix Agent	ops_unix_agent	Contains details about Linux/Unix agent resources .
Linux/Unix Agent Cluster	ops_unix_agent_cluster	Contains details about Linux/Unix agent clusters .
Linux/Unix Agent Cluster Version	ops_unix_agent_cluster_v	Contains previous versions of Linux/Unix cluster records.
Linux/Unix Agent Mapping	ops_agent_mapping_unix	Shows the mapping specifications between local Linux/Unix agents and Linux/Unix agents on a Promotion Target (as retrieved using the Refresh Target Agents button).
Linux/Unix Task	ops_task_unix	Contains details about Linux/Unix tasks .
Linux/Unix Task Instance	ops_exec_unix	Contains details about Linux/Unix task instances .
Linux/Unix Task Version	ops_task_unix_v	Contains previous versions of Linux/Unix task records.
Manual Task	ops_task_manual	Contains details about Manual tasks .
Manual Task Instance	ops_exec_manual	Contains details about Manual task instances .
Manual Task Version	ops_task_manual_v	Contains previous versions of Manual task records.
Manual Trigger	ops_trigger_manual	Contains Manual trigger records.
Manual Trigger Version	ops_trigger_manual_v	Contains previous versions of Manual trigger records.

Mutually Exclusive Tasks	ops_tasks_to_exclusive	Contains relationship information between tasks and mutually exclusive tasks; that is, which tasks are mutually exclusive with each other.
Mutually Exclusive Tasks	ops_tasks_to_exclusive_v	Contains previous versions of relationship information between tasks and mutually exclusive tasks.
Note	ops_note	Contains details about Notes attached to Opwise records.
Note	ops_note_v	Contains previous versions of Notes records.
Opwise Permissions	ops_permission	Contains details about Opwise Permissions assigned to Opwise Users and Opwise Security Groups .
Opwise Properties	ops_config	Contains Opwise Configuration Properties .
Output	ops_exec_output	Contains any output (such as STDOUT) attached to task instances.
Outstanding Exclusive Request	ops_exclusive_order	Contains any outstanding requests to run exclusively by a task instance.
Outstanding Request	ops_resource_order	Contains any outstanding requests for a resource by a task instance.
Promotion History	ops_promotion_history	Contains a list of Bundles that have been promoted into the current database.
Promotion History Item	ops_promotion_history_item	Contains a list of records that have been promoted into the current database. If a record has been promoted more than once, each version is listed separately.
Promotion Target	ops_bundle_target	Contains Promotion Target records.
Restart Confirmation	ops_exec_zos_confirm	Contains details about any restart confirmations performed on z/OS tasks.
Restartable JobSteps	ops_exec_zos_jobstepsui	Contains details about restartable job steps in a z/OS task.
SAP Connection	ops_sap_connection	Contains SAP Connection records.
SAP Connection Version	ops_sap_connection_v	Contains previous versions of SAP Connection records.
SAP Connections	ops_bundle_sap_cntn_join	Contains relationship information between SAP Connection records and Bundles ; that is, which SAP Connection records are in which Bundles .
SAP Task	ops_task_sap	Contains SAP task records.
SAP Task Instance	ops_exec_sap	Contains SAP task instance records.
SAP Task Version	ops_task_sap_v	Contains previous versions of SAP task records.
Script	ops_script	Contains Script records.
Script Version	ops_script_v	Contains previous versions of Script records.
Scripts	ops_bundle_script_join	Contains relationship information between Script records and Bundles ; that is, which Scripts belong to which Bundles .
Set Variable	ops_variable_action	Contains details about Set Variable actions.
Set Variable	ops_variable_action_v	Contains previous versions of Set Variable actions.
Sleep Task	ops_task_sleep	Contains details about Sleep tasks.
Sleep Task Instance	ops_exec_sleep	Contains details about Sleep task instances.

Sleep Task Version	ops_task_sleep_v	Contains previous versions of Sleep tasks records.
SNMP Manager	ops_snmp_connection	Contains SNMP Manager records.
SNMP Manager Version	ops_snmp_connection_v	Contains previous versions of SNMP Manager records.
SNMP Managers	ops_bundle_snmp_cntn_join	Contains relationship information between SNMP Manager records and Bundles ; that is, which SNMP Managers belong to which Bundles .
SNMP Notification	ops_snmp_cluster_notification	Contains SNMP notifications defined for Cluster Nodes .
SNMP Notification	ops_snmp_conn_notification	Contains SNMP notifications defined for Connectors .
SNMP Notification	ops_snmp_notification	Contains SNMP notifications defined for Tasks .
SNMP Notification	ops_snmp_notification_v	Contains previous versions of SNMP notifications defined for Tasks . Note that a new version is created only when the task is updated.
SQL Result Set	ops_sql_results	Contains output from SQL tasks .
SQL Task	ops_task_sql	Contains details about SQL tasks .
SQL Task Instance	ops_exec_sql	Contains details about SQL task instances .
SQL Task Version	ops_task_sql_v	Contains previous versions of SQL tasks records.
SQL Warning Set	ops_sql_warnings	Contains warnings returned by executed SQL statements .
Step Condition	ops_exec_zos_stepcond	Contains details about z/OS task instance step conditions
Step Condition	ops_task_zos_stepcond	Contains details about z/OS task step conditions
Step Condition	ops_task_zos_stepcond_v	Contains previous versions of z/OS task step conditions
Stored Procedure Parameters	ops_stored_proc_param	Contains Parameter records associated with Stored Procedure tasks .
Stored Procedure Parameters	ops_stored_proc_param_v	Contains previous versions of Parameter records associated with Stored Procedure tasks . Note that versions are only created when the task is updated.
Stored Procedure Task	ops_task_stored_proc	Contains details about Stored Procedure tasks .
Stored Procedure Task Instance	ops_exec_stored_proc	Contains details about Stored Procedure task instances .
Stored Procedure Task Version	ops_task_stored_proc_v	Contains previous versions of Stored Procedure tasks records.
System Monitor	ops_task_system_monitor	Contains System Monitor task records.
System Monitor Task Instance	ops_exec_system_monitor	Contains System Monitor task task instance records.
System Monitor Version	ops_task_system_monitor_v	Contains previous versions of System Monitor task records.

Task	ops_task	Contains details about tasks of every type , along with associated Task Instance information.
Task Instance Run Criteria	ops_exec_run_criteria	Contains run criteria information for task instances within a Workflow .
Task Instance Virtual Resources	ops_exec_to_resource	Contains relationship information between Virtual resources and task instances; that is, which task instances are assigned to which Virtual Resources.
Task Monitor	ops_task_monitor	Contains details about Task Monitor tasks .
Task Monitor Instance	ops_exec_monitor	Contains details about Task Monitor task instances .
Task Monitor Trigger	ops_trigger_tm	Contains details about Task Monitor triggers .
Task Monitor Trigger Version	ops_trigger_tm_v	Contains previous versions of Task Monitor trigger records.
Task Monitor Version	ops_task_monitor_v	Contains previous versions of Task Monitor task records.
Task Run Criteria	ops_task_run_criteria	Contains run criteria information for tasks within Workflows .
Task Run Criteria	ops_task_run_criteria_v	Contains previous versions of run criteria information for tasks within Workflow . Note that new versions are created only when the Workflow task is updated.
Task Version	ops_task_v	Contains previous versions of all task records.
Task Virtual Resources	ops_task_to_resource	Contains relationship information between Virtual resources and tasks; that is, which tasks are assigned to which Virtual Resources.
Task Virtual Resources	ops_task_to_resource_v	Contains previous versions of relationship information between Virtual resources and tasks.
Tasks	ops_bundle_task_join	Contains relationship information between Task records and Bundles ; that is, which Tasks are in which Bundles.
Temporary Trigger	ops_trigger_temp	Contains details about Temporary triggers .
Temporary Trigger Version	ops_trigger_temp_v	Contains previous versions of Temporary trigger records.
Time Trigger	ops_trigger_time	Contains details about Time triggers .
Time Trigger Version	ops_trigger_time_v	Contains previous versions of Time trigger records.
Trigger	ops_trigger	Contains details about triggers of every type .
Trigger Version	ops_trigger_v	Contains previous versions of Trigger records.
Triggers	ops_bundle_trigger_join	Contains relationship information between Trigger records and Bundles ; that is, which Triggers are in which Bundles.
User	sys_user	Contains details about User records .
User Role	sys_user_has_role	Contains details about User and Role records , including which Users have which Roles.
Variable Version	ops_variable_v	Contains previous versions of Global variables.
Variables	ops_bundle_variable_join	Contains relationship information between Global variables and Bundles ; that is, which Global variables belong to which Bundles.
Variables	ops_local_variable	Contains details about task and trigger variables (also called local variables), entered into the Variables tab on a task or trigger record.

Variables	ops_local_variable_v	Contains previous versions of Local variables associated with tasks or triggers. Note that new version records are only created when the task or trigger is updated.
Variables	ops_variable	Contains details about Global variables , entered by selecting Variables from the Navigation pane.
Virtual Resource	ops_virtual_resource	Contains details about Virtual resource records.
Virtual Resource Version	ops_virtual_resource_v	Contains previous versions of Virtual resources .
Virtual Resources	ops_bundle_resource_join	Contains relationship information between Virtual resources and Bundles ; that is, which Virtual resources belong to which Bundles.
Windows Agent	ops_windows_agent	Contains details about Windows agents .
Windows Agent Cluster	ops_windows_agent_cluster	Contains details about Windows agent clusters .
Windows Agent Cluster Version	ops_windows_agent_cluster_v	Contains previous versions of Windows Agent Cluster records.
Windows Agent Mapping	ops_agent_mapping_windows	Shows the mapping specifications between local Windows agents and Windows agents on a Promotion Target (as retrieved using the Refresh Target Agents button).
Windows Task	ops_task_windows	Contains details about Windows tasks .
Windows Task Instance	ops_exec_windows	Contains details about Windows task instances .
Windows Task Version	ops_task_windows_v	Contains previous versions of Windows task records.
Workflow Task	ops_task_workflow	Contains details about Workflow tasks .
Workflow Task Edges	ops_task_workflow_edge	Contains information about the conditions specified between tasks in workflows .
Workflow Task Edges	ops_task_workflow_edge_v	Contains previous versions of information about the conditions specified among tasks in workflows . Note that new version records are only created when the Workflow task is updated.
Workflow Task Instance	ops_exec_workflow	Contains details about Workflow task instances .
Workflow Task Instance Edges	ops_exec_workflow_edge	Contains information about the conditions specified between task instances within workflows .
Workflow Task Instance Vertices	ops_exec_workflow_vertex	Contains relationship information between workflows instances and task instances; that is, which tasks are running in which workflows .
Workflow Task Version	ops_task_workflow_v	Contains previous versions of workflow task records.
Workflow Task Vertices	ops_task_workflow_vertex	Contains relationship information between tasks and workflows; that is, which tasks are in which workflows .
Workflow Task Vertices	ops_task_workflow_vertex_v	Contains previous versions of the relationship between tasks and workflows. Note that new version records are created only when the workflow task is updated.
z/OS Agent	ops_zos_agent	Contains details about z/OS agents .
z/OS Agent Mapping	ops_agent_mapping_zos	Shows the mapping specifications between local z/OS agents and z/OS agents on a Promotion Target (as retrieved using the Refresh Target Agents button).
z/OS Restartable JobSteps ND	ops_exec_zos_jobsteps	Contains details about restartable job steps in a z/OS task .
z/OS Task	ops_task_zos	Contains details about z/OS tasks .

z/OS Task Instance	ops_exec_zos	Contains details about z/OS task instances.
z/OS Task Version	ops_task_zos_v	Contains previous versions of z/OS task records.

Opwise Activity Table (ops_exec)

When you create an Activity report, you select data only from the Opwise Activity table (`ops_exec`), which contains all available data about executed task instances.

Field Name	Description
Agent	For agent-based tasks, the name of the agent.
Agent Acquired	System-supplied; For internal processing only.
Agent Cluster Acquired	System-supplied; For internal processing only.
Attempt	A counter that keeps track of the number of times this task instance was attempted.
Average Estimated End Time	System-supplied.
CPU Time	System-supplied; Amount of CPU time the task took to run.
Calendar	If Special Restriction is selected, the calendar defines the Holidays or Non Business days. Enter a calendar name or click the magnifying glass icon either to browse for an existing calendar or to add a new calendar. To display details about the calendar specified in this field, hover over the paper icon.
Class	Type of task instance, such as Sleep task instance or Workflow task instance.
Created	Date and time when the task instance was created.
Created by	User ID of the user who created the task.
Credentials(credentials)	Credentials under which an Agent runs this task. These credentials override any credentials provided in the Agent resource definition for any Agent running this task.
Credentials(credentials_var)	The variable specified in the login credentials field, if enabled.
Credentials Variable	Optional. If enabled, the Credentials field (see above) converts from a reference field (where you browse and select a record) into a text field that allows you to enter a variable. Use the format: <code>\${variable name}</code> . The variable must be a supported type as described in Variables and Functions .
Current Retry Count	System-supplied; Displays, only for a running task instance, the current number of times that Opwise has retried the task after it first went to failure status.
Duration	System-supplied; Amount of time the task took to run.
Duration in Seconds	The amount of time, in seconds, the task instance took to run.

Early Finish	If enabled, and if the task instance finishes before the time or period specified, the task instance is flagged as early. You can specify a time or duration to determine an early finish (see Early Finish Type). To determine whether a task instance finished early, open the task instance and locate the Finished Early field; the field is checked if the instance finished before the specified time or did not last as long as expected. This field only appears on the task instance if the user added Early Finish specifications to the task definition.
Early Finish Duration	If Early Finish Type is Duration, use this to specify the shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Early Finish Time	If Early Finish Type is Time, use this to specify the time before which the task finish time is considered early. That is, enter a time at which the task should still be running. Use hh:mm, 24-hour time.
Early Finish Type	Required if Early Finish is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it finishes before the specified time (see Early Finish Time). • Duration - Flag the task if it finishes a certain amount of time before the programmed finish time (see Early Finish Duration). The task must have a specific finish time.
End Time	System-supplied; Date and time the task instance completed
Exclusive State	Current status of the exclusive request being used by a task instance. Opswise uses the same process each time it launches a task with exclusive requirements and goes through the same series of states: <ul style="list-style-type: none"> • INITIAL - The initial state. This is the default value at launch time. • REQUESTED - When the task requests its exclusive requirements, the exclusive state becomes Requested. • ACQUIRED - When all of the requested exclusive requirements are met and acquired by the server, the exclusive state becomes Acquired. • RETURNED - When the task completes, the server returns the acquired exclusive requirements and the exclusive state becomes Returned. • CLEARED - When the Clear Exclusive command is run, the server cancels or returns the exclusive requirements and the exclusive state becomes Cleared.
Execution User	System-supplied; If the task was launched manually, the ID of the user who launched it.
Exit Code	The exit code, if any, returned by the process.
Finished Early	System-supplied; This field is flagged if the task finished earlier than the time specified in the Early Finish fields.
Finished Late	System-supplied; This field is flagged if the task finished later than the time or duration specified in the Late Finish fields.
Forced Finished	True or False. Indicates whether the task instance was force-finished.
Hold Reason	Information about why the task will be put on hold when it starts.
Hold on Start	If enabled, when the task is launched it appears in the Activity display with a status of Held. The task runs when the user releases it.
IO Other	Total input/output operations for this task.
IO Reads	Total input/output reads for this task.

IO Writes	Total input/output writes for this task.
Instance Name	Required. Name used within Opwise to identify this task. Up to 40 alphanumeric; variables supported. It is the user's responsibility to develop a workable naming scheme for tasks.
Instance Reference Id	System-supplied; Opwise increments this number each time the task is run.
Invoked by	System-supplied; How the task instance was launched. One of the following: <ul style="list-style-type: none"> • Trigger: (Trigger Name) - Instance was launched by the named trigger. • Workflow: (Workflow Name) - Instance was launched by the named workflow. • Manually Launched - Instance was launched by a user. To determine the name of the user: <ul style="list-style-type: none"> • From the Activity or Task Instances screen, click the task instance name to open the record. • The Execution User field identifies the user who launched the task instance.
Late Finish	If enabled, and if the task instance finishes after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late finish (see Late Finish Type). To determine whether a task instance finished late, open the task instance and locate the Finished Late field; the field is checked if the instance finished after the specified time or lasted longer than expected. This field only appears on the task instance if the user specified a Late Finish in the task definition.
Late Finish Duration	If Late Finish Type is Duration, use this to specify the longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Late Finish Time	If Late Finish Type is Time, use this to specify the time after which the task finish time is considered late. Use hh:mm, 24-hour time.
Late Finish Type	Required if Late Finish is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it finishes after the specified time (see Late Finish Time). • Duration - Flag the task if it finishes a certain amount of time after the programmed finish time (see Late Finish Duration). The task must have a specific finish time.
Late Start	If enabled, and if the task instance starts after the time or period specified, the task instance is flagged as late. You can specify a time or duration to determine a late start (see Late Start Type). To determine whether a task instance started late, open the task instance and locate the Started Late field; the field is checked if the instance started after the specified time. This field only appears on the task instance if the user specified a Late Start in the task definition.
Late Start Duration	If Late Start Type is Duration, use this to specify the longest amount of time this task instance can wait before starting. You can specify any combination of hours, minutes, and seconds.
Late Start Time	Time after which the task start time is considered late. Use hh:mm, 24-hour time
Late Start Type	Required if Late Start is enabled. Options are: <ul style="list-style-type: none"> • Time - Flag the task if it starts after the specified time. • Duration - Flag the task if it starts a certain amount of time after the programmed start time. The task must have a specific start time.
Longest Estimated End Time	System-supplied.
Maximum Retries	User-defined. The maximum number of times Opwise should retry this task after it has started and gone to a failed state.

Member of Business Services	User-defined. Allows you to select one or more Business Services that this record definition belongs to. Click the lock icon to unlock the field and select Business Services .
Memory Peak	The peak amount of memory used during the execution of this task instance.
Memory Used	The amount of memory used during the execution of this task instance.
Progress	Indicates the workflow progress in terms of completed tasks: success, finished, or skipped. (A sub-workflow within a workflow counts as one task.) For example, 5/10 indicates that 5 of 10 tasks within the workflow have completed.
Queued Time	System-supplied; the time that the task was queued for processing.
Resources Consumed	System-supplied; For internal processing only.
Resources State	<p>The current status of the resource being used by a task instance. Opwise uses the same process each time it launches a task on a resource and the resource goes through the same series of states:</p> <ol style="list-style-type: none"> 1. INITIAL - The initial state. This is the default value at launch time. 2. REQUESTED - When the task requests the resources it needs, the resource state becomes Requested. 3. ACQUIRED - When all of the requested resources are available and acquired by the server, the resource state becomes Acquired. 4. RETURNED - When the task completes, the server returns the resources it was using, and the resource state becomes Returned.
Retry Indefinitely	User-defined. Enabled or disabled. Indicates whether Opwise should continue trying indefinitely to run this task. If you enable this field, it overrides any value placed in the Maximum Retries field (above).
Retry Interval (Seconds)	User-defined. The number of seconds between each retry.
Run Called	(Internal property.)
Run Criteria Run Time	Indicates that run-time run criteria was specified for the task.
Run Criteria Trigger Time	Indicates that trigger-time run criteria was specified for the task.
Security Name	The task name.
Shortest Estimated End Time	System-supplied.
Start Time	System-supplied; the date and time that the task started.
Started Late	System-supplied; This field is flagged if the task started later than the time specified in the Late Start fields.
Status	System-supplied; Provides additional information, if any, about the status of the task.
Status Description	System-supplied; Provides additional information, if any, about the status of the task.
Sys id	Unique system identifier associated with a task instance.
Task	Required. Name used within Opwise to identify this task. Up to 40 alphanumeric; variables supported. It is the user's responsibility to develop a workable naming scheme for tasks.
Task Description	User-supplied description of this record.
Task Priority	Priority of this task instance, as set by the user via the Set Priority command . Options are: HIGH, MEDIUM, LOW.
Trigger	Required. Name used within Opwise to identify this trigger. It can contain a maximum of 40 alphanumeric characters. It is the user's responsibility to develop a workable naming scheme for triggers.

Type	Type of task instance.
Updated	Date and time this record was last updated.
Updated by	User who last updated this record.
Updates	Number of updates that have been made to the task record.
User Estimated End Time	System-supplied; If the user entered information into the User Estimated Duration field in the task definition, Opwise uses this information to calculate an end time for the task instance, based on the date/time the task instance started.
Vertex Id	Each task within a workflow has a unique vertex ID, which distinguishes it from other tasks of the same name, if any.
Waited for Exclusive	Indicates that the task instance could not run exclusively immediately and went into an Exclusive Wait state.
Waited for Resources	Indicates that the task instance could not get resources immediately and went into a resource wait state.
Workflow Definition ID	ID of the parent workflow task definition.
Workflow Id	Name of the workflow, if appropriate.
Workflow Start Time	Start time of the parent workflow task instance.
Show Related Files	Not supported for reports.

Managing Records



Opwise Records Management

Overview



Record Versioning

Overview

Viewing Old Versions of Records

Restoring Old Versions of Records

Purging Old Versions of Records

Enabling/Disabling Versioning



Exporting and Importing Records

Run an Opwise Export

Run an Opwise Import



Bundling and Promotions

Overview

Creating Bundles

Defining Promotion Targets

Promoting Bundles and

Generating a Bundle Report

Audit Records

Promotion History and



Backing Up and Purging

Overview

Creating a New Backup

Manually Running a Backup

Importing Backed/Purged

Backup Field Descriptions



The information on these pages also is located in the Opwise Automation Center 5.1.1 User Guide.pdf.

Overview of Opwise Records Management

Opwise supports several features that allow you to manage records and control the amount of data in your database:

- [Record Versioning](#)
- [Bundling and Promoting Records](#)
- [Backing Up and Purging Data](#)
- [Exporting and Importing Records](#)

Record Versioning

- Overview
- Viewing Old Versions of Records
- Restoring Old Versions of Records
- Purging Old Versions of Records
 - Purge Specific Versions Manually
 - Purge All Outdated Versions Manually
 - Purge All Outdated Versions Automatically
- Enabling/Disabling Versioning

Overview

Opswise maintains historical copies of most user-created records in the database. These include tasks and their associated records (virtual resources, variables, actions, notes), calendars and their custom day associations, custom days, variables, credentials, virtual resources, scripts (and associated notes), email templates and connections, database connections, SNMP managers, SAP connections, agent clusters, applications, Business Services, and triggers (and associated variables).

These historical copies - old versions of the current records - are read-only.

When you update any of these records, Opswise creates an image of the old version and stores it in the record's **Versions** tab. It also updates the Version field in the current version of the record. For example, if you have updated TaskABC three times, there will be three versions of that task stored in the Versions tab, and the current version will be identified as Version 4.

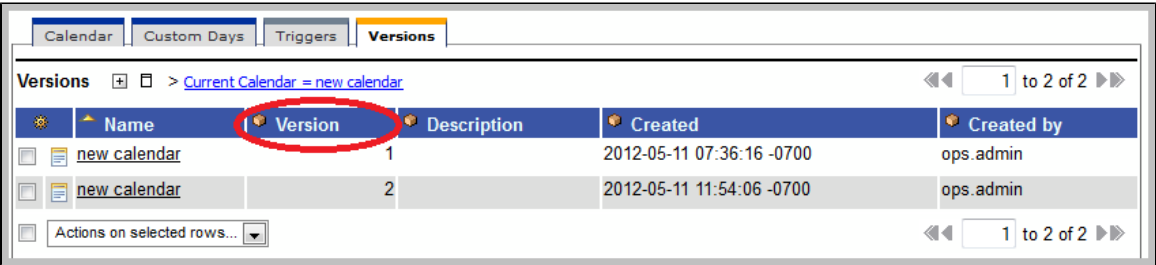
You can view a video about Record Versioning.

Viewing Old Versions of Records

To view old versions:

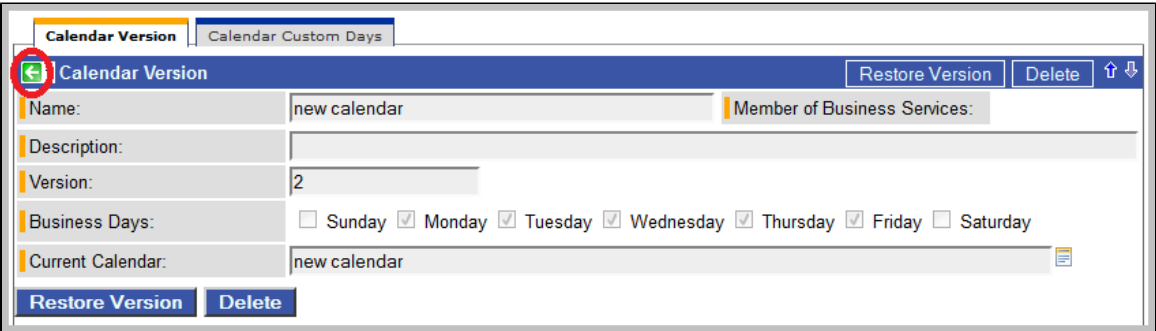
Step 1 Open the record you want to view.

Step 2 Click the **Versions** tab. Opswise displays a list of old versions that exist for this record. You can sort this list in ascending or descending order by clicking the **Version** column label.



Name	Version	Description	Created	Created by
new_calendar	1		2012-05-11 07:36:16 -0700	ops.admin
new_calendar	2		2012-05-11 11:54:06 -0700	ops.admin

Step 3 Click the name of the version you want to view. All of the associated records (tabs) contain the data as it existed when this version was the current version. At the top of the screen, the Name field contains the name of the record as it existed for this version. At the bottom of the screen, the Current <Record Type> field contains the current name of this record.



Calendar Version

Name: new_calendar Member of Business Services:

Description:

Version: 2

Business Days: Sunday Monday Tuesday Wednesday Thursday Friday Saturday

Current Calendar: new_calendar

Restore Version Delete

Step 4	To return to the list of Record Versions, click the green arrow.
---------------	--

Restoring Old Versions of Records

You can restore old versions to the current version. When you restore an old version, the current version will become the newest old version.

Step 1	Display the version of the record you want to restore.
---------------	--

Step 2	Click the Restore Record button.
---------------	---

Purging Old Versions of Records

Opwise provides three methods for purging old versions of records.



Note

When you purge old versions of records, the version number of the current version remains the same.

Purge Specific Versions Manually

Step 1	Display the record definition screen for the record versions you want to purge.
---------------	---

Step 2	Click the Version tab to display a list of old versions of that record.
---------------	--

Step 3	Either:
---------------	---------

1. Select one or more versions to be purged and click **Delete** on the drop-down menu at the bottom of the list.
2. Click the **Name** of the Version to be purged and, on the definition screen for that version, click the **Delete** button.

Purge All Outdated Versions Manually

Step 1	From the navigation pane, select Automation Center Administration -> Properties .
---------------	---

Step 2	Run the purge_versions_exceeding_maximum.js maintenance script to purge versions that exceed the maximum number of records allowed, as defined by the System Default Maximum Versions Opwise system property.
---------------	---

Purge All Outdated Versions Automatically

Step 1	From the navigation pane, select Automation Center Administration -> Properties .
---------------	---

Step 2	Set the Automatically Purge Versions Opwise Controller property to true to automatically purge versions that exceed the maximum number of records allowed, as defined by the System Default Maximum Versions Opwise system property.
---------------	---

Enabling/Disabling Versioning

Two properties are available that allow you to control if and when Opwise automatically creates a new version of a record (and all its associated records):

- The [Automatically Create Versions](#) property (true or false) determines whether modifications to the record itself will cause Opwise to create a new version of the record. The default value is true. If this property is set to false, Opwise does not create versions.
- The [Create Version On Related List Change](#) property (true or false) determines whether changes, deletions, or additions to a related list will cause Opwise to create a new version of the record. For example, if this property is enabled, Opwise will create a new version of the task and all its associated records when the user adds a variable to the task, deletes a Note, or changes an Email Notification. The default value is true. If this property is set to false, and [Automatically Create Versions](#) is set to true, Opwise creates a new version only if the base record is updated.

To change the value of either property:

Step 1	From the navigation pane, select Configuration > Properties .
---------------	---

Step 2 Locate the property you want to change:

The screenshot shows the 'Opswise Properties' configuration page. On the left, a navigation pane lists various configuration categories, with 'Properties' selected and highlighted by a red box. The main content area displays a table of properties. Two rows in the table are highlighted with red boxes: 'Automatically Create Versions' with a value of 'true', and 'Create Version On Related List Change' with a value of 'true'.

Name	Value
Administrator Email Address	
Agent Cache Retention Period In Days	7
Agent Heartbeat Interval In Seconds	120
Agent Prefix	AGNT
Automatically Create Versions	true
Automatically Purge Versions	false
Automatically Skip Conflicting Multi-Origin Paths	false
Broadcast On Hold If Cluster Suspended	true
Compress Bundle Promotion Payload	false
Copy Notes To Task Instances For Reporting	false
Create Version On Related List Change	true

Step 3 Click the property **Name** to open the record and enter **true** or **false** in the Value field.

Step 4 Click the **Update** button.

Exporting and Importing Records

Opswise provides several utilities that allow you to export and import records. Normally you will use these to migrate data during a system upgrade.

See [Run an Opswise Export](#) and [Run an Opswise Import](#) in the migration documentation.

Bundling and Promoting Records

- Overview
 - Bundling and Promoting Process
- Creating Bundles
 - Manually Defining a New Bundle
 - Creating a Bundle Based on Date
 - Adding Record(s) to a Bundle From Record Lists and Forms
 - Displaying a Record's Bundles
- Defining Promotion Targets
 - Promotion Target Field Descriptions
 - Specifying Agent Mapping
- Promoting Bundles and Records to a Target
 - Introduction
 - Promoting a Bundle
 - Promoting One or More Individual Records
 - Promotion Error Messages
- Generating a Bundle Report
- Audit Records
- Promotion History and the Restore Option

Overview

The Opwise Bundling and Promoting features allow you to select and bundle a group of Opwise records and "promote" them from one Opwise server to another. For example, you can use these features when you create your workflows on a development installation then move them to a QA installation for testing. Once you are satisfied with the stability of the workflows, you can promote them to your production system.



Caution

Bundle promotion should be performed only to and from the same versions, or to a higher version, of Opwise Automation Center. Promoting to an older version could result in loss of data or promotion exception.

Bundling and Promoting Process

The general process for bundling and promoting your data from a source server is:

Step 1	On the source server, define one or more Bundle records.
Step 2	On the source server, create a Promotion Target record for each target server.
Step 3	Specify agent mappings between the source and target server.
Step 4	Promote Bundles and/or individual records to the target.

These features use web services calls to communicate when you are promoting Bundles of records from one server to another.

To see a demonstration of how to bundle and promote records, watch the [Bundling and Promoting](#) video.

Creating Bundles

You can create Bundles manually by selecting records yourself or you can specify a date parameter that automatically selects all records added or changed on or after the date. Each procedure is described below.

Manually Defining a New Bundle

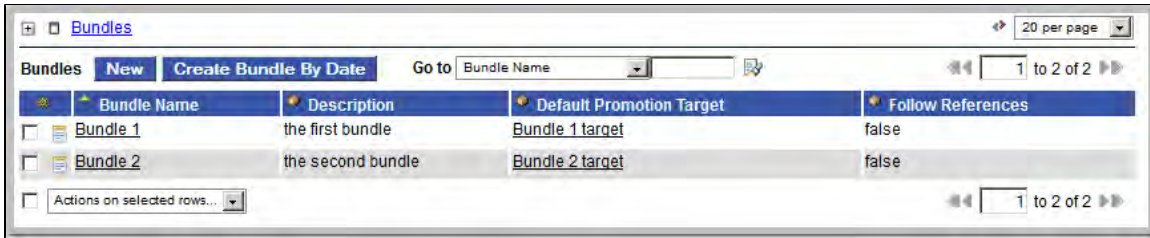


Note

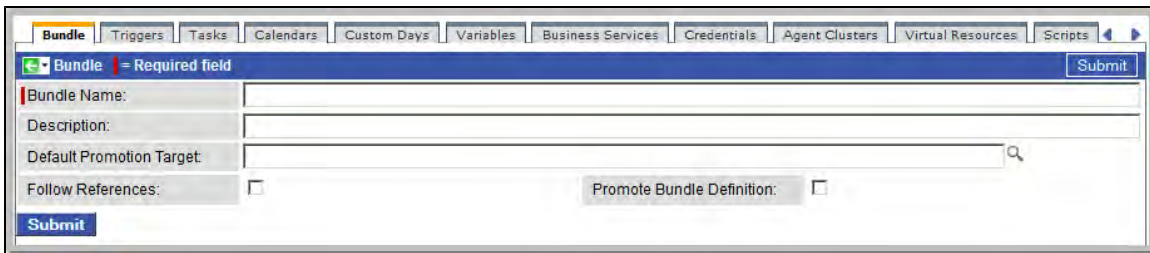
Your userid must have the `ops_bundle_admin` role to use this feature.

The procedure for defining a new Bundle involves creating the Bundle record, selecting which records you want to include, and selecting a target for promotion.

Step 1 From the navigation pane, select **Automation Center Bundles & Promotion > Bundles**. The Bundles List screen displays.



Step 2 Click **New**. The Bundle Definition screen displays.





Step 3 Each Bundle record includes a separate tab for each type of record you can include in the Bundle (the Bundle Definition screen, above, does not show all of these tabs). Use the [field descriptions](#) below as a guide for filling in the fields. Minimally, you must specify a Bundle name. You optionally can specify a default promotion target. If you do not specify a default, you will be prompted for a target when performing a promotion of the bundle.

Step 4 Before selecting records to include in the Bundle, you must first create the Bundle record in the database. Click the down arrow and select **Save** or click **Submit**.

Step 5 To select records for inclusion in this Bundle, select the tab for that record type. (You also can add a record to a bundle directly from a record form, or you can add one or more records directly from a records list. See [Adding Records to a Bundle](#), below).

You can create a new record to add to the Bundle by selecting **New** and filling out the form. Or, you can select from existing records as follows:

1. Click the **Edit** button. Opswise displays all records for which you have read permission.
2. The records listed under Collection are existing records that do not already belong to this Bundle. The records listed under Has (Record Type) List are records that belong to this Bundle.
3. To filter the records listed under Collection:
 - a. Select filter conditions in the --choose field--, --oper--, and --value-- fields. (See [Create a Filter](#) for information about how to construct a filter.)
 - b. If you want to add more filter conditions, click **Add Filter**.
 - c. When you have defined the filter you want, click **Run Filter**. The Collection list now displays only those records that match the filter.
 - d. To remove filter conditions, click the X (Delete) icon that displays to the right of each set of filter conditions, and then click **Run Filter**.
4. To add to or remove records from the Has (Record Type) List:
 - To add a record to the list, double-click the record in the Collection list. Or, use **CTRL-click** to select multiple records and click the right arrow .
 - To remove a record from the list, double-click the name in the Has (Record Type) List. Or, use **CTRL-click** to select multiple records and click the left arrow .


As you click a record, Opswise displays details about it at the bottom of the form.

Step 6 When you are finished, click **Save**.

Step 7 Repeat the above steps in the appropriate tabs for all records you want to add.

Bundle Field Descriptions

Field Name	Description
------------	-------------

Bundle Name	Required. Name for this bundle.
Description	User-supplied description of this record.
Default Promotion Target	Allows you to browse for and select a Promotion Target, which you defined using the Promotion Targets feature.
Follow References	<p>Specification for whether or not to dynamically include items that are referenced by bundled item definitions.</p> <p>For example, if you bundle an Email Task and enable the Follow References option, the Email Connection and/or Email Template referenced by the Email Task will be included in the promotion operation. If you bundle a Trigger and enable the Follow References option, the Task(s) and Calendar referenced by the Trigger will be included in the promotion. Tasks within a bundled Workflow are included in the promotion regardless of the Follow Reference option. Custom Days that are defined within a bundled Calendar are included in the promotion regardless of the Follow Reference option as well.</p> <p>(The information included is similar to the information included in an XML export with references.)</p>
Promote Bundle Definition	<p>Enable this option to promote the bundle definition along with the bundle itself when performing the promotion operation.</p> <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note This option is not supported when promoting to Opwise Automation Center 5.1.1.3 or earlier.</p> </div>
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Bundle Report button	Allows you to generate a report about the current bundle. See Generating a Bundle Report .
Promote Bundle button	Allows you to promote this bundle to a target Opwise server. See Promoting a Bundle to a Target .
Delete button	Deletes the current record.
Record type tabs	Each tab shows the records of that type that belong to this Bundle. The New and Edit keys allow you to add records to the Bundle, as described in the above procedure.

Creating a Bundle Based on Date



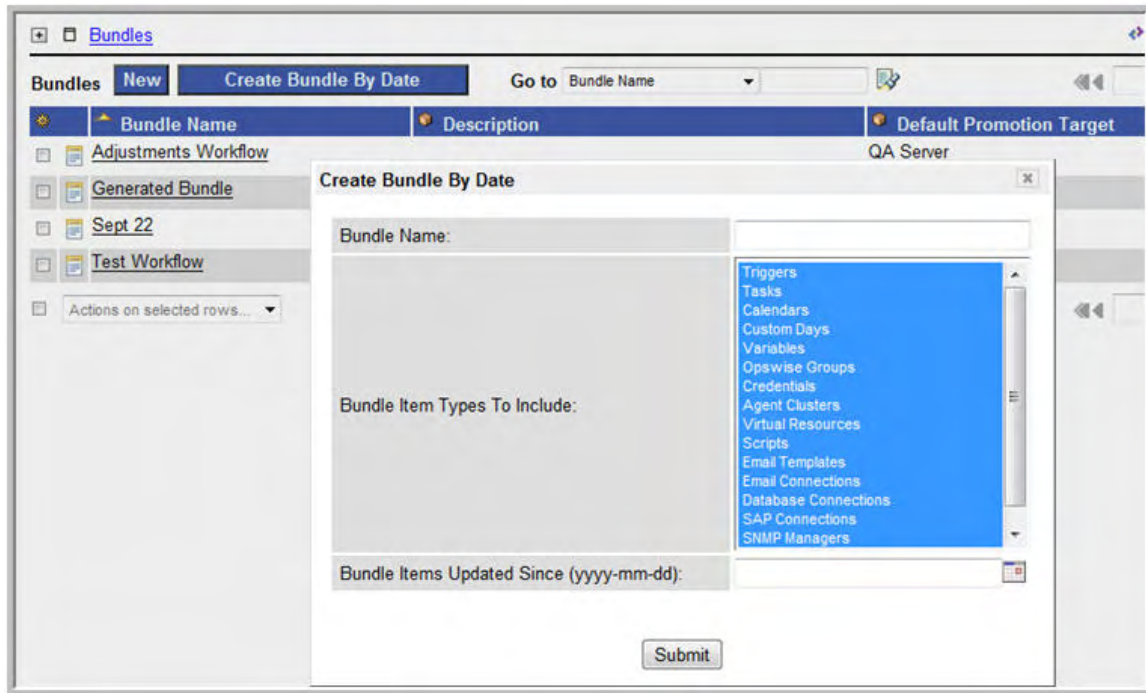
Note

Your userID must have the ops_Bundle_admin [role](#) to use this feature.

This feature allows you to Bundle all records created or updated on the current server on or since a specific date. The procedure involves selecting which records you want to include and specifying the date parameter.

Step 1 From the navigation pane, select **Automation Center Bundles & Promotion > Bundles**.

Step 2 Click the **Create Bundle by Date** button. The Create Bundle by Date pop-up screen displays. By default, all record types are selected, as shown below:



Step 3 You can leave the default all record types selected. Or, to deselect the group of records, click anywhere in the list. Use **CTRL-click** to select multiple record types.

Step 4 To select the date parameter, click the calendar in the lower right corner. A calendar pops up.

Step 5 Click the date you want applied to this Bundle. All records created or updated on or after that date will be included.

Step 6 Click the **Submit** button. If any records qualified for inclusion in the Bundle, the Bundle is created and saved to the database. However, if no records qualified according to the specified date, the Bundle is not saved.

Adding Record(s) to a Bundle From Record Lists and Forms

At any time, you can add record(s) to a Bundle from lists of records or from the record itself, as described below.

To add a single record from a list of records:

Step 1	Right click the record you want to add.
Step 2	Select Add to Bundle . Opswise displays a window, prompting you to select the Bundle from a drop-down list of Bundles.
Step 3	Select the Bundle from the drop-down list and click Submit .

To add multiple records from a list of records:

Step 1	Select the records you want to add by clicking the box to the left of the record name.
Step 2	From the Actions on selected rows menu, select Add to Bundle . Opswise displays a window, prompting you to select the Bundle from a drop-down list of Bundles.
Step 3	Select the Bundle from the drop-down list and click Submit .

To add the current record to a Bundle:

Step 1	Open the record you want to add.
---------------	----------------------------------

Step 2	Click the down-arrow next to the record name or right-click the record name. A menu appears.
Step 3	Select Add To Bundle . Opwise displays a window, prompting you to select the Bundle from a drop-down list of Bundles.
Step 4	Select the Bundle from the drop-down list and click Submit .

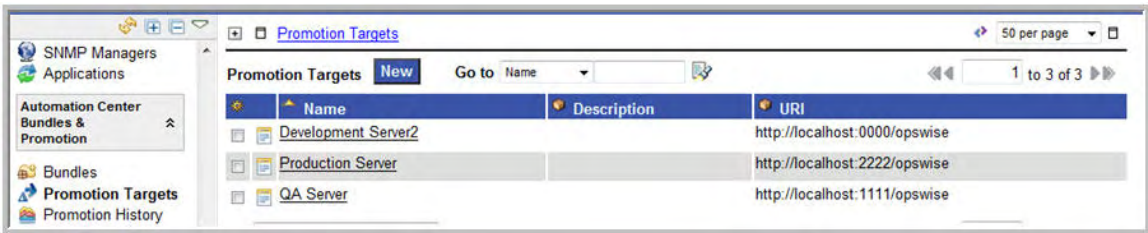
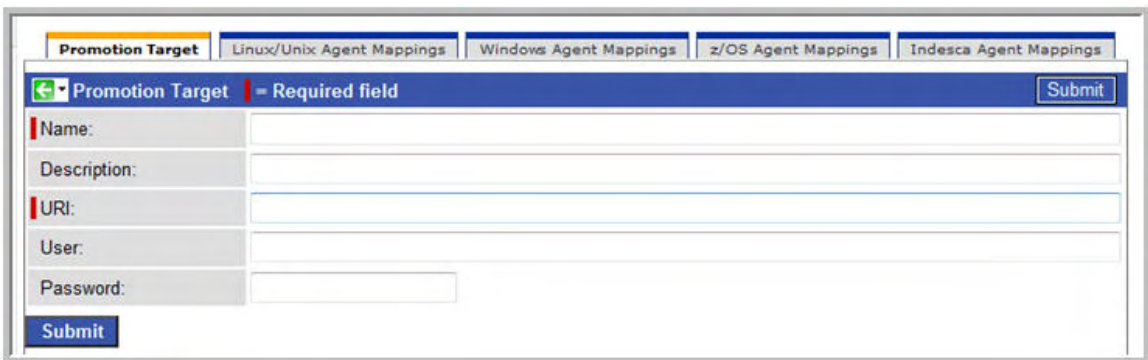
Displaying a Record's Bundles

To find out what Bundles a record belongs to:

Step 1	Open the record.
Step 2	Move your cursor over the down-arrow next to the record name or right-click the record title bar. A menu appears.
Step 3	Select View Bundles . Opwise displays a list of Bundles to which the current record belongs. You can add a new Bundle on the fly, click a Bundle name to view it, or click the browser's back button to return to the record.

Defining Promotion Targets

Before you can promote Bundles or individual records, you must identify and create a Promotion Target record(s) for the target machine(s). The Promotion Target record provides the Uniform Resource Identifier (URI) of the target server, along with the user name and password required to log on to Opwise on the target server.

Step 1	From the navigation pane, select Automation Center Bundles & Promotion > Promotion Targets . Opwise displays a list of existing Promotion Targets.
	
Step 2	To create a new record, click New . Opwise displays a blank Promotion Target form.
	
Step 3	Using the field descriptions below for guidance, complete the form. Minimally, you must provide a record name and URI.
Step 4	Click Submit to save the record.

Promotion Target Field Descriptions

Field Name	Description
Name	Required. Name for this promotion target.
Description	User-supplied description of this record.

URI	Required. Uniform Resource Identifier (URI) used to locate the target server. Similar to a URL.
User	Login ID on the target instance of Opwise.
Password	Login password on the target instance of Opwise.
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Refresh Target Agents button	Accesses the specified Opwise server and fetches all agent records. For details, see Specify Agent Mapping .
Delete button	Deletes the current record.
Agent Mappings tabs	Each tab contains the agent mapping instructions between the source server and the target server. See Specify Agent Mapping .

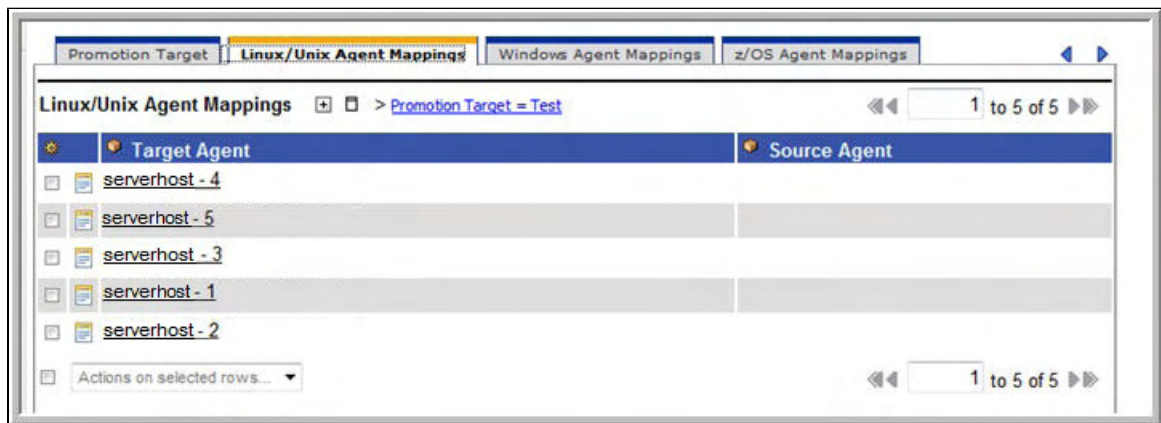
Specifying Agent Mapping

Because your source and target Opwise instances may not have the same agents, you must provide instructions to Opwise on how to map agents on the source machine to agents on the target machine. The process consists of instructing Opwise to fetch the list of agents on the target server and manually identifying how each agent should be mapped.

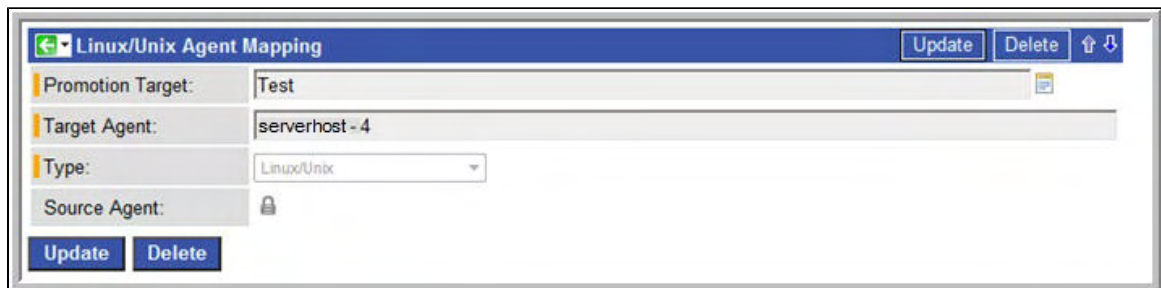
Step 1	Open the Promotion Target record for which you want to specify the mapping.
Step 2	Click Refresh Target Agents . If the Promotion Target record does not provide the user ID and password, Opwise prompts for them. If you want to override the ID and password from the Promotion Target record, click Override User/Password and type in the new information.
Step 3	Click Submit . Opwise logs in to the target machine specified in the URI field, accesses the appropriate tables, sorts the agents into agent types and lists them in the appropriate agent tab in the Promotion Target record.


Step 4 To specify agent mapping, complete the steps below for each tab that contains agents:

1. Click the tab title to display the target agents. The example below shows a list of Unix agents on a target server:



2. To specify mapping, click the name of a Target Agent. A record opens showing more detail, as shown in the example below.



3. In the Source Agent field, click the Lock icon . Click the magnifying glass to browse for and select a record from the list of agents defined on the source machine. Select the source agent that you want to map to the target agent.
4. Repeat the above procedure for each agent listed. For best results, you should make sure all the agents on your source system are mapped to an agent on the target system. You can map as many source agents to a single target agent as needed. Once you have specified the mapping for all your source agents, you can easily promote Bundles or individual records to this Promotion Target. When you promote records (via Bundle or individually) to the target machine, the target agent will replace the source agent.

Refresh Agent Error Messages


If your setup is incorrect, you may see the error message described below.

If you tried to refresh target agents using a non-existent user or invalid password on the Promotion Target:

Error Message	Location
<pre>GET http://NN.NNN.NN.N:8080/opwise/resources/agents/list returned a response status of 401 Unauthorized</pre>	User interface on source machine
<pre>2012-03-29-16:27:17:134 ERROR [http-8080-6] com.sun.jersey.api.client.UniformInterfaceException: GET returned a response status of 401 Unauthorized</pre>	Opwise log on source machine

<pre>2012-03-29-16:27:16:138 ERROR [http-8080-1] *** ERROR *** Login using Basic Authentication failed for: [userID]</pre>	Opswise log on target machine
--	-------------------------------

Promoting Bundles and Records to a Target

 **Note**
 To use this feature, the user logged in to the source machine must have the `ops_promotion_admin` role. Also, the user ID and password specified for the Promotion Target must be a valid user on the target instance with the `ops_promotion_admin` role.

Introduction

Promoting a Bundle means copying all of the records in a Bundle from a source Opswise instance to a target Opswise instance. You also can promote one or more individual records without first bundling them.

For every item in the bundle being promoted and every item being promoted individually, the following associated item data is always included in the promotion.

- If you promote a workflow, all of the tasks in the workflow are also promoted.
- If you promote a task (including a workflow), all variables, virtual resource dependencies, actions, notes, etc. are included in the promotion.
- If you promote an application, its associated start, stop, and query tasks are included in the promotion.
- If you promote a calendar, its associated custom days are included in the promotion.

Promoting a Bundle

Step 1	Select the Bundle you want to promote.
Step 2	Click Promote Bundle . Opswise displays a window, prompting you to select the Promotion Target from a drop-down list. Select the target.
Step 3	The default login ID and password are provided from the Promotion Target record, if specified. If you want to override the default, click Override User/Password and type in the new information.
Step 4	Click Submit . Opswise logs in to the target machine specified in the URI field of the Promotion Target Record and copies the bundled records to the target server. Based on the specified agent mapping, the target agent replaces the source agent where required.

This process creates audit records on the source and target machines. On the target machine, Opswise also creates a Promotion History record. For details, see [Promotion History and the Restore Option](#).

Promoting One or More Individual Records

Opswise also allows you to promote records to a target server without going through the process of creating a Bundle.

Step 1	<p>Select the record(s) you want to promote:</p> <ul style="list-style-type: none"> • To promote a single record from a list of records: <ol style="list-style-type: none"> 1. Right click the record you want to promote. 2. Select Promote. • To promote multiple records from a list of records: <ol style="list-style-type: none"> 1. Select the records you want to promote by clicking the box to the left of the record name. 2. From the Actions on selected rows menu, select Promote. • To promote the current record: <ol style="list-style-type: none"> 1. Open the record you want to promote. 2. Click the down-arrow next to the record name or right-click the record name. A menu appears. 3. Select Promote.
---------------	---

Step 2 The Promote dialog displays, prompting you to select the Promotion Target from a drop-down list. Select the target.

Step 3 The default login ID and password will be provided from the Promotion Target record. If you want to override the default, click **Override User/Password** and type in the new information.

Step 4 Select **Follow References** if you want to dynamically include items that are referenced by these records.

Step 5 Click **Submit**. Opwise logs in to the target machine specified in the URI field of the Promotion Target Record and copies the selected records to the target server.

This process creates audit records on the source and target machines. On the target machine, Opwise also creates a [Promotion History](#) record.

Promotion Error Messages

If your setup is incorrect, you may see the following error messages.

If you tried to promote a bundle or record using a non-existent user or invalid password on the Promotion Target:

Error Message	Location
<pre>Command Promote Bundle failed to execute: POST http://NN.NNN.NN.N:8080/opwise/resources/bundle/promote returned a response status of 401 Unauthorized.</pre>	User interface and Opwise log on source machine
<pre>2012-03-29-16:41:36:185 ERROR [http-8080-4] *** ERROR *** Login using Basic Authentication failed for: [userID]</pre>	Opwise log on target machine

If you tried to promote a Bundle or record using a valid user/password on the Promotion Record that does not have the ops_promotion_admin role:

Error Message	Location
<pre>Command Promote Bundle failed to execute: [Command Accept Bundle prohibited due to security constraints.]</pre>	User interface on source machine

Generating a Bundle Report

The Bundle Report feature allows you to display on a single page all the records included in the current Bundle, with a summary of records at the top.

To generate the report, display the Bundle and click the **Bundle Report** button. An example is shown below:

Bundle Report

Bundle Summary

Bundle Name:	SAP Bundle		
Bundle Description:			
Follow References:	Yes		
Promote Bundle Definition:	Yes		
Default Promotion Target:			
Bundled Items Total:	6		

Triggers:	0	Variables:	0	Virtual Resources:	0	SNMP Managers:	0
Tasks:	4	Business Services:	0	Scripts:	0	SAP Connections:	1
Calendars:	0	Credentials:	1	Email Templates:	0	Applications:	0
Custom Days:	0	Agent Clusters:	0	Email Connections:	0	Database Connections:	0

Tasks

Name	Type	Description	Added By
DISPLAY - InfoPackage - Status - SBT1	TASK: SAP		ops.admin
DISPLAY - InfoPackages - SBT*	TASK: SAP		ops.admin
DISPLAY Process Chain (SBT1)	TASK: SAP		ops.admin
DISPLAY Process Chain Start Condition (SBT1)	TASK: SAP		ops.admin

Credentials

Name	Type	Description	Added By
SAP - CB7 - STONEBRANCH1	CREDENTIAL	QA User for SAP CB7	*

SAP Connections

Name	Type	Description	Added By
CB7 - XBP 3.0 with Business Warehouse	SAP CONNECTION		*



Note

Items included in the promotion that were added directly to the Bundle have a user id in the **Added By** column. Items included in the promotion dynamically (that is, Tasks within a Workflow, Custom Days within a Calendar, or, when the Follow References option is enabled, the number of items referenced by the bundled items) have an asterisk (*) in the **Added By** column .

Audit Records

Whenever a Bundle or an individual record is promoted to a target server, Opwise creates audit records on both the source and target servers.

On the source system side, each time you promote a record or a Bundle, Opwise creates a single [audit record](#) for that event. If you promoted a Bundle, the audit message is PROMOTE_Bundle; if you promoted a single record or multiple records, the audit message is PROMOTE. An example audit record is shown below for a Bundle called Adjustments Workflow:

	Command	2011-10-05 14:40:41 -0700	ops.admin Executing Command: PROMOTE BUNDLE on Adjustments Workflow
--	---------	------------------------------	--

On the target server side, Opwise creates an ACCEPT_BUNDLE audit record, along with "child" audit records associated with that promotion (either record[s] or a Bundle). These may include UPDATE commands for records that existed on the target already and CREATE commands for records that did not previously exist.

For example, if you promote an updated Calendar record, Opwise creates an ACCEPT_BUNDLE audit for the promotion, and Update audit records for the calendar and each of the custom days used in the calendar, as shown in the following example:

Audit Records		Go to	Audit Date		1 to 50 of 920
Audit Type	Audit Date	Created by	Description		
Command	2011-10-05 15:18:34 -0700	ops.admin	Executing Command: ACCEPT_BUNDLE		
Update	2011-10-05 15:18:34 -0700	ops.admin	Update: CalendarBean Opwise - American Calendar2, sys_id: da24b953c0a8016501fbc18464d8328		
Update	2011-10-05 15:18:34 -0700	ops.admin	Update: CustomDayBean Ops - Christmas, sys_id: dadf6b84c0a8016500a01a10da5e2c15		
Update	2011-10-05 15:18:34 -0700	ops.admin	Update: CustomDayBean Ops - Labor Day, sys_id: dae33d0ec0a80165011afd42735c51c		
Update	2011-10-05 15:18:34 -0700	ops.admin	Update: CustomDayBean Ops - Columbus Day, sys_id: dae02128c0a8016501a418a3e7f2336f		
Update	2011-10-05 15:18:34 -0700	ops.admin	Update: CustomDayBean Ops - Flag Day, sys_id: dae1446ac0a80165018db581a2b4adff		
Update	2011-10-05 15:18:34 -0700	ops.admin	Update: CustomDayBean Ops - Veterans Day, sys_id: dae8ee83c0a801650185982cf458979b		
Update	2011-10-05 15:18:34 -0700	ops.admin	Update: CustomDayBean Ops - President's Day, sys_id: dae815e6c0a80165006044d56880230e		
Update	2011-10-05 15:18:34 -0700	ops.admin	Update: CustomDayBean Ops - New Year's Day, sys_id: dae77560c0a80165006d8051753b806c		
Update	2011-10-05 15:18:34 -0700	ops.admin	Update: CustomDayBean Ops - Memorial Day, sys_id: dae6360ac0a80165014bce4f528d0e4a		
Update	2011-10-05 15:18:34 -0700	ops.admin	Update: CustomDayBean Ops - Independence Day, sys_id: da252636c0a8016500c21e52ab3dacf9		

Promotion History and the Restore Option

Opwise creates a Promotion History record each time a record or a Bundle is promoted into the target system. You can access the Promotion History records by selecting **Automation Center Bundles & Promotion > Promotion History** from the navigation pane on the target server.

Each Promotion History record provides a complete list of all records promoted during this promotion event, along with their version numbers. This screen also provides the option of restoring records to the state they were in before the promotion. This applies only to records being updated by the promotion, not those being created by a promotion. See the field descriptions below for details.

The fields on the Promotion History screen are system-supplied and display-only.

Field Name	Description
Bundle Name	Name of this record.
Source Node	Machine name or URI of the machine where the source Opwise system is running.
Promotion User	UserID of the user who promoted the bundle or record(s).
Updated	Date and time this record was last updated.
Promotion History Items tab	This tab lists all the record (items) promoted as part of this bundle or set of records. Each item in the list provides the name and type of the record, the latest version number on the target, the previous version number on the target, and the source version number.

Run Report button	Generates a one page summary of the contents of the bundle. See Generating a Bundle Report .
Restore Unchanged button	For records that already existed on the target server, you can restore them to their state prior to the promotion. The Restore Unchanged button restores only those records that have not been changed since the promotion updated the record. For records that were created on the target machine by the promotion, no changes will occur since no previous version exists.
Restore All button	For records that already existed on the target server, you can restore them to their state prior to the promotion. The Restore All button restores all records that were updated by the promotion, including records that were modified since the promotion. For records that were created on the target machine by the promotion, no changes will occur since no previous version exists.
Delete button	Deletes the current record.

Backing Up and Purging Data

- Overview
- Creating a New Backup / Purge Instruction
 - Backup Definition Screen Field Descriptions
- Manually Running a Backup/Purge
- Importing Backed/Purged Data into Opwise
- Returning Virtual Resources for Purged Task Instances in Failure Status

Overview

Opwise maintains a record of all system activity, including:

- Audit records
- Activity
- History

The Backup screen allows you to configure automatic backups and/or purges of some or all of Opwise activity data. Depending on your organization's needs, you should schedule regular data backups. Depending on the volume of your installation, the amount of data in your Opwise database could become unwieldy if you do not schedule regular purges of old data.

The data is written to XML files in the directory you specify.



Note

For instructions on how to purge user-created Opwise records, see [Purging Old Versions of Records](#).

Creating a New Backup / Purge Instruction

Step 1 From the navigation pane, select **Automation Center Administration > Configuration > Data Backup / Purge**. The Backups List screen displays.

Name	Tables	Days Older Than	Schedule	Next Scheduled Time	Purge	Backup	Enabled
BP	Audit, Activity, History	31	true	2013-04-20 01:15:00 -0700	true	true	

Step 2 Click **New**. The Backup Definition screen displays.

Backup = Required field Submit

Name:

Table: Audit Activity History

Purge:

Backup:

Days Older Than:

Export Path:

Schedule:

Every # Days:

Time (hh:mm):

Next Scheduled Time:

Submit

Step 3	Using the field descriptions provided below as a guide, complete the fields as needed. If you want the backup/purge to run automatically, enable the Schedule field and specify how often and what time it should run. Otherwise, you can run it manually.
Step 4	Click the Submit button to save the record and return to the menu, or right-click the title bar and select Save to save the record and remain on the current display.
Step 5	To enable the instructions, right-click on the record name to display the menu and select Enable Backup . The list of Backup records identifies which records are enabled and disabled.
Step 6	If appropriate, repeat these steps for any additional Backup / Purge records you want to add.

Backup Definition Screen Field Descriptions

Field Name	Description
Name	Name of this backup specification.
Table	Specifies which records you want to back up and/or purge: <ul style="list-style-type: none"> • Audit • Activity • History
Purge	If enabled, the process will purge the selected data from your Opwise database.
Backup	If enabled, the process will write all the selected data to XML files.
Days Older Than	Allows you to specify the minimum number of days you wish to retain data. The process will run according to the schedule you specify, only processing data that is older than the number of days you specify in this field.
Export Path	Specifies the path to the directory to which you want the backed up data written. The data must be backed up to a location on the server's file system. It is written to a separate XML file for each record type, as shown in the following examples: <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <pre>Audit: ops_audit_Sat_Apr_30_08_30_00_PDT_2011.xml Activity: ops_exec_sleep_Sat_Apr_30_08_30_00_PDT_2011.xml ops_exec_unix_Sat_Apr_30_08_30_00_PDT_2011.xml ops_exec_workflow_Sat_Apr_30_08_30_00_PDT_2011.xml History: ops_history_Sat_Apr_30_08_30_00_PDT_2011.xml</pre> </div>
Schedule	If enabled, displays additional fields that allow you to specify an automated backup and/or purge schedule. If you do not select schedule, you must manually run the backup / purge process.
Every # Days	Specifies, in number of days, the frequency of the backup / purge process. Default is 1.
Time	Specifies the time of the backup / purge. Use 24:00 hour time.

Next Scheduled Time	Displays the next scheduled time the backup / purge process will run, based on the specifications in your schedule.
Submit button	Submits the new record to the database.
Update button	Saves updates to the record.
Enable Backup button	Enables these Backup / Purge instructions so that they will be processed by the Controller.
Disable Backup button	Disables these backup / purge instructions so they will not be processed by the Controller.
Delete button	Deletes the current record.
Run button	Manually runs the backup / purge instructions.

Manually Running a Backup/Purge

If you want to manually run a backup or purge, either:

- Right-click on the Backup/Purge record you want to run and select **Run**.
- Open the record and click the **Run** button.

Importing Backed/Purged Data into Opswise

If you need to import any of the XML files created by using the backup/purge function, you can copy the XML file(s) into the bulk export output path and run bulk import. See [Running Opswise Import](#).

Returning Virtual Resources for Purged Task Instances in Failure Status

Task instances that have their **Hold Resources on Failure** field enabled will hold their renewable [virtual resources](#) if the task instance is in [Failed](#) status.

However, since task instances with a Failed status can qualify for being purged, any renewable resources still held by a task instance when it is purged will be returned.