



stonebranch
you imagine IT. we automate IT.

Opswise Controller 6.1.x

Tasks and Workflows

© 2015 by Stonebranch, Inc. All Rights Reserved.

1. Tasks and Workflows	3
1.1 Creating Tasks	4
1.1.1 Linux Unix Task	20
1.1.2 Windows Task	38
1.1.3 zOS Task	58
1.1.3.1 Creating and Running a zOS Task	59
1.1.3.2 Special Processing on zOS Tasks	81
1.1.3.3 Creating Step Conditions	90
1.1.3.4 Creating Step Actions	98
1.1.3.5 Creating Restart Criteria	103
1.1.4 Indesca Task	108
1.1.5 SAP Task	127
1.1.6 File Transfer Task	159
1.1.7 Manual Task	197
1.1.8 Timer Task	208
1.1.9 SQL Task	219
1.1.10 Stored Procedure Task	234
1.1.11 Email Task	251
1.1.12 Task Monitor Task	264
1.1.13 File Monitor Task	279
1.1.14 FTP File Monitor Task	296
1.1.15 System Monitor Task	315
1.1.16 Creating Task Actions	329
1.1.16.1 Abort Actions	330
1.1.16.2 Email Notification Actions	333
1.1.16.3 Set Variable Actions	338
1.1.16.4 SNMP Notification Actions	339
1.1.16.5 System Operation Actions	342
1.1.17 Copying Tasks	347
1.1.18 Setting Mutually Exclusive Tasks	349
1.1.19 Creating Notes	351
1.2 Manually Running and Controlling Tasks	353
1.3 Creating and Maintaining Workflows	376
1.4 Monitoring Task Activity	420
1.4.1 Monitoring Activity from the Activity Monitor	421
1.4.2 Monitoring Activity from the Task Instances List	429
1.4.3 Monitoring Activity History	432
1.4.4 Monitoring Workflows	435
1.4.5 Viewing Task Instances for a Specific Task	438
1.4.6 Displaying Task Instance Status	441
1.4.7 Retrieving Output	444

Tasks and Workflows



Tasks

Creating Tasks

Task Types

Manually Running and Controlling Tasks

Retrieving Output from a Completed Task

Copying Tasks

Setting Mutually Exclusive Tasks

Creating Step Conditions

Creating Step Actions

Creating Notes



Workflows

Creating and Maintaining Workflows

Monitoring Workflows in Workflow Monitor



Monitoring Tasks

Monitoring Activity from

Monitoring Activity from

Monitoring Activity Hist

Viewing Task Instances

Displaying Task Instanc



Task Actions

Creating Task Actions

Abort Actions

Email Notification Actio

Set Variable Actions

SNMP Notification Actio

System Operation Actio



The information on these pages also is located in the Opswise Controller 6.1.x Tasks and Workflows.pdf.

Creating Tasks

- Tasks
 - Task Types
 - Built-In Variables
- Creating a Task
- All Tasks List
 - All Tasks List Information
- <Task Type> Tasks List
 - <Task Type> Tasks List Information
- Task Details
- Task Instance Details
- Task and Task Instance Details Fields
- Updating Tasks
- Deleting Tasks
- Viewing Task Parents
- Additional Information

Tasks

An Opwise Controller 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 INFINTRAN protocols.
Manual	Create a pause in the workflow during which the user must take some action.
Timer	Execute a timer 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 Controller 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.

Creating a Task

There are multiple ways to create a task:

1. On the [All Tasks list](#), click the **New** drop-down list and select a task type. A Task Details pop-up displays that lets you enter / select information for a new task.
2. On the Tasks list for a specific [task type](#), click the **New** icon. A Task Details pop-up displays that lets you enter / select information for a new task.
3. In the empty (except for default values) [Task Details](#) that displays below the task list for a specific task type, enter / select information for a new task. If the Task Details displays information for an existing task, click the **New** button to clear the Task Details and enter / select information for a new task.
4. Create a [copy](#) of a task by clicking the **Copy** button in the Task Details and renaming the task.

For detailed information on creating a task for a specific task type, click that task type in the [Task Types](#) table, above.

All Tasks List

To display a list of all currently defined tasks for all task types, from the [Automation Center](#) navigation pane select **Tasks > All Tasks**. The All Tasks list displays.

Task Name	Type	Task Description	Updated By	Updated
stonebranch-applicationcontroltask-01	Application Control		stonebranch-user-01	2014-06-13 14:23:55 -0400
stonebranch-applicationcontroltask-02	Application Control		stonebranch-user-02	2014-06-13 14:24:02 -0400
stonebranch-applicationcontroltask-03	Application Control		stonebranch-user-03	2014-06-13 14:24:07 -0400
stonebranch-applicationcontroltask-04	Application Control		stonebranch-user-04	2014-06-13 14:24:11 -0400
stonebranch-applicationcontroltask-05	Application Control		stonebranch-user-05	2014-06-13 14:24:16 -0400
stonebranch-emailtask-01	Email		stonebranch-user-02	2014-06-13 14:17:06 -0400
stonebranch-emailtask-02	Email		stonebranch-user-03	2014-06-13 14:17:19 -0400
stonebranch-emailtask-03	Email		stonebranch-user-04	2014-06-13 14:17:24 -0400
stonebranch-emailtask-04	Email		stonebranch-user-05	2014-06-13 14:17:27 -0400
stonebranch-emailtask-05	Email		stonebranch-user-02	2014-06-13 14:17:32 -0400
stonebranch-filemonitor-01	File Monitor		stonebranch-user-03	2014-06-30 11:36:05 -0400
stonebranch-filemonitor-02	File Monitor		stonebranch-user-04	2014-06-13 14:20:04 -0400
stonebranch-filemonitor-03	File Monitor		stonebranch-user-05	2014-06-13 14:20:09 -0400
stonebranch-filemonitor-04	File Monitor		stonebranch-user-02	2014-06-13 14:20:13 -0400
stonebranch-filemonitor-05	File Monitor		stonebranch-user-03	2014-06-13 14:20:18 -0400

All Tasks List Information

The following table provides a description of the default columns that display on the All Tasks list.

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

Column	Description
Task Name	User-defined. Name assigned to this task.

Type	Type of task. Options: <ul style="list-style-type: none"> • Workflow • Linux/Unix • Windows • z/OS • Indesca • SAP • File Transfer • Manual • Timer • SQL • Stored Procedure • Email • Task Monitor • File Monitor • FTP File Monitor • System Monitor • Application Control
Task Description	User-defined. Copied from the Task Description field in the task.
Updated By	System-supplied; User that last updated this record.
Updated	System-supplied; Date and time this record was last updated.

<Task Type> Tasks List

If you select **Tasks > <Task Type>** from the [Automation Center](#) navigation pane, a tasks list for that selected task type displays.

For example:

The screenshot displays the 'Linux/Unix Tasks' section of the Opwise Controller. At the top, there is a list of 5 tasks with columns for Task Name, Task Description, Command or Script, Updated By, and Updated. Below this is a 'Linux/Unix Task Details' panel with tabs for General, Variables, Actions, Virtual Resources, Mutually Exclusive, Instances, Triggers, Notes, and Versions. The 'General' tab is active, showing fields for Task Name, Version (set to 1), Task Description, Member of Business Services, Hold on Start, Virtual Resource Priority (set to 10), and Hold Resources on Failure. At the bottom, there are fields for Agent and Agent Cluster.

<Task Type> Tasks List Information

The default columns that display on a <Task Type> Tasks list are specific to that type of task.

Task Details

When you click a task in a Tasks List, Task Details below the list displays all currently defined information for that task.

The screenshot displays the 'Linux/Unix Tasks' interface. At the top, there is a table listing five tasks. Below the table, the 'Linux/Unix Task Details' section is visible, showing configuration options for the selected task 'stonebranch-linuxunixtask-01'.

Task Name	Task Description	Command or Script	Updated By	Updated
stonebranch-linuxunixtask-01		Command	stonebranch-user-01	2014-07-07 15:18:10 -0400
stonebranch-linuxunixtask-02		Command	stonebranch-user-02	2014-07-02 13:22:43 -0400
stonebranch-linuxunixtask-03		Command	stonebranch-user-03	2014-06-13 13:56:11 -0400
stonebranch-linuxunixtask-04		Command	stonebranch-user-04	2014-06-13 13:56:21 -0400
stonebranch-linuxunixtask-05		Command	stonebranch-user-05	2014-06-13 13:56:36 -0400

The 'Linux/Unix Task Details' section includes the following fields:

- General:**
 - Task Name: stonebranch-linuxunixtask-01
 - Version: 7
 - Task Description: (empty)
 - Member of Business Services: stonebranchbusinessservice 01
 - Hold on Start:
 - Virtual Resource Priority: 10
 - Hold Resources on Failure:
- Linux/Unix Details:**
 - Agent: stonebranch-linuxunixagent-01
 - Agent Cluster: (empty)
 - Agent Variable:
 - Credentials: (empty)
 - Cluster Broadcast: (empty)
 - Run as sudo:

For more information on viewing Details of any record type, including tasks, see [Records](#).

For information on Details for a specific task type, click the appropriate link in [Task Types](#), above.

Task Instance Details

When you run a task, the Controller create a task instance of that task.

You can view task instance details either by:

- From the Task Details, click the Instances tab and select a task instance of that task.
- Select a task instance from the [Activity Monitor](#).
- Select a task instance from the [Task Instances list](#).

Linux/Unix Task Instance Details: stonebranch-linuxunixtask-01
Update Re-run View Parent Retrieve Output... Delete Refresh Close

Linux/Unix Task Instance
Virtual Resources
Exclusive Requests
Output
Notes

General

Instance Name: Reference Id:

Task: Invoked By:

Task Description:

Member of Business Services: Execution User:

Virtual Resource Priority: Hold Resources on Failure:

Status

Status: Exit Code:

Status Description:

Queued Time:

Start Time: End Time:

Duration: CPU Time:

Process ID:

Linux/Unix Details

Agent: Agent Cluster:

Agent Variable: Agent Cluster Variable:

Credentials: Credentials Variable:

Command or Script: Run as sudo:

Command:

Parameters:

Runtime Directory:

Environment Variables:

Name	Value
No items to show.	

Exit Code Processing:

Exit Codes:

Automatic Output Retrieval:

Start Line: Number of Lines:

Scan Text:

Retry Options

Retry Exit Codes:

Maximum Retries: Retry Indefinitely:

Retry Interval (Seconds): Current Retry Count:

Statistics

User Estimated End Time: Average Estimated End Time:

Shortest Estimated End Time: Longest Estimated End Time:

Update Re-run Retrieve Output... Delete Refresh Close

Task and Task Instance Details Fields

The following table identifies all fields that appear in task and/or task instance Details of one or more task type(s).

Unless otherwise indicated, each field appears in the Details for both tasks and task instances.

Each field name is a link to a description of that field.



Note

Links within these field descriptions are active only when viewing the field descriptions table for a specific task type.

Field Name	Task only	Task Instance only	Email	File Monitor	File Transfer (FTP/SFTP)	File Transfer (INFITRAN)	FTP File Monitor	Indesca	Linux Unix	Manual	SAP	SQL	Storage Process
Agent				✓	✓		✓		✓				
Agent Cluster				✓	✓		✓		✓				
Agent Cluster Variable				✓	✓		✓		✓				
Agent Variable				✓	✓		✓		✓				
Append Destination Open Option					✓								
Append Infitran Options							✓						
Append Source Open Options						✓							
Auto Cleanup												✓	✓
Auto Layout													
Automatic Output Retrieval								✓	✓				
Auto-Restart Option													
Average Estimated End Time	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Average Instance Time			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
BCC			✓										
Body			✓										
By Percentage (+/-)				✓									
By Scale (File Monitor)				✓									
By Scale (System Monitor)													
By Size (+/-)				✓									
CC			✓										
Cluster Broadcast				✓	✓		✓		✓				

Codepage						✓							
Column Name												✓	✓
Command								✓	✓				
Command (FTP)				✓									
Command Group											✓		
Command or Script								✓	✓				
Compress						✓							
Condition													
CPU Time		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Credentials				✓	✓		✓		✓				✓
Credentials Variable				✓	✓		✓		✓				✓
Current Retry Count		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Database Connection												✓	✓
Database Connection Variable												✓	✓
Default Calendar													
Delete Override JCL													
Destination Credentials						✓							
Destination Credentials Variable						✓							
Destination Filename(s)						✓							
Destination File System						✓							
Destination Infitran Agent						✓							
Destination Infitran Agent Option						✓							
Duration		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Duration													
Early Finish			✓	✓	✓		✓	✓	✓	✓	✓	✓	✓
Early Finish Duration			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Early Finish Time			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Early Finish Type			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Email Connection			✓										
Email Template			✓										
Encrypt						✓							
End Time		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Environment Variables									✓	✓			
Execution User		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Exit Code		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Exit Code Processing						✓		✓	✓		✓		
Exit Codes						✓		✓	✓		✓		
Exit Codes (SQL)												✓	
Exit Codes (Stored Procedure)													✓
File Creation Option						✓							
File Owner				✓									
Finished Early		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Finished Late		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
First Time Ran			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
From [+/-]hh:mm:													
FTP Credentials					✓		✓						
FTP Credentials Variable					✓		✓						
Highest Instance Time			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Hold on Start	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Hold Reason	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Hold Resources on Failure			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Hold Resources on Failure (Workflow)													

Indesca Agent								✓					
Indesca Agent Option								✓					
Indesca Credentials								✓					
Indesca Credentials Variable								✓					
Indesca Options								✓					
Instance Name		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Interact with Desktop													
Invoked by		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
JCL Location													
JCL Override Location													
Job Card (z/OS only)					✓		✓						
Last Instance Duration			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Last Override Deletion													
Last Time Ran			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Late Finish			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Late Finish Duration			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Late Finish Time			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Late Finish Type			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Late Start			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Late Start Duration			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Late Start Time			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Late Start Type			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
List Format Options							✓						
Local Filename					✓								
Longest Estimated End Time		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Lowest Instance Time			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Maximum Files				✓									
Maximum Polls							✓						
Maximum Retries				✓	✓		✓	✓		✓	✓	✓	
Maximum Rows											✓	✓	
Member of Business Services			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Monitor File(s)				✓									
Monitoring Type													
Monitor Type (File Monitor)				✓									
Monitor Type (FTP File Monitor)							✓						
Monitor Type (System Monitor)													
Mount Points or Drives													
Network Fault Tolerant						✓							
New Jobclass													
New Jobname													
New Msgclass													
Number of Instances			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Number of Lines							✓	✓		✓			
Number of Lines (z/OS)													
Number of Override Instances													
Operator												✓	✓
Output File (for Exit Code Processing)						✓		✓	✓		✓		
Output File (for Automatic Output Retrieval)								✓	✓		✓		
Output Type						✓		✓	✓		✓		
Override Instance Count for Deletion													

Parameter Position													✓
Parameters									✓				
Poll Interval (Seconds)							✓						
Procedure Library													
Process ID		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Queued Time		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Recursive				✓									
Reference Id		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Remote Filename (File Transfer)					✓								
Remote Filename (FTP File Monitor)							✓						
Remote Server (File Transfer)					✓								
Remote Server (FTP File Monitor)							✓						
Reply-To			✓										
Resource Available													
Result Processing (SQL)												✓	
Result Processing (Stored Procedure)													✓
Retry Exit Codes					✓	✓		✓	✓				
Retry Indefinitely					✓	✓		✓	✓		✓	✓	✓
Retry Interval (Seconds)					✓	✓		✓	✓		✓	✓	✓
Run as sudo									✓				
Runtime Directory								✓	✓		✓		
Run with Highest Privileges													
SAP Command Options											✓		
SAP Connection											✓		

SAP Credentials													✓	
SAP Credentials Variable										✓				
SAP Language													✓	
Scan Forward				✓										
Scan Output For						✓		✓	✓			✓		
Scan Text (File Monitor)					✓									
Scan Text (for Automatic Output Retrieval)								✓	✓			✓		
Scan Text (for Automatic Output Retrieval - z/OS)														
Schedule ID														
Script													✓	
Script File									✓					
Script Options									✓					
Server Type								✓						
Shortest Estimated End Time		✓	✓	✓	✓		✓	✓	✓	✓	✓	✓	✓	✓
Show/Hide Skipped Tasks														
Source Credentials							✓							
Source Credentials Variable							✓							
Source Filename(s)							✓							
Source File System							✓							
Source Infitran Agent							✓							
Source Infitran Agent Option							✓							
SQL Command													✓	
Stable (seconds)				✓										
Stable (Seconds) - FTP								✓						

Started Late		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Start Line								✓	✓		✓		
Start Line (z/OS)													
Start Time		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Status		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Status Description		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Status To Monitor													
Stored Procedure Name													✓
Subcommands (z/OS only)					✓								
Subject			✓										
Task		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Task Description			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Task Name	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Task Name Condition													
Task Name Contains													
Task Name Ends With													
Task Name Starts With													
Task to Monitor													
Task Type to Monitor													
Time in Seconds													
Timer Type													
Time Scope													
To			✓										
To Scale				✓									
To Size				✓									
To [+/-]hh:mm:													
Transfer Method					✓								

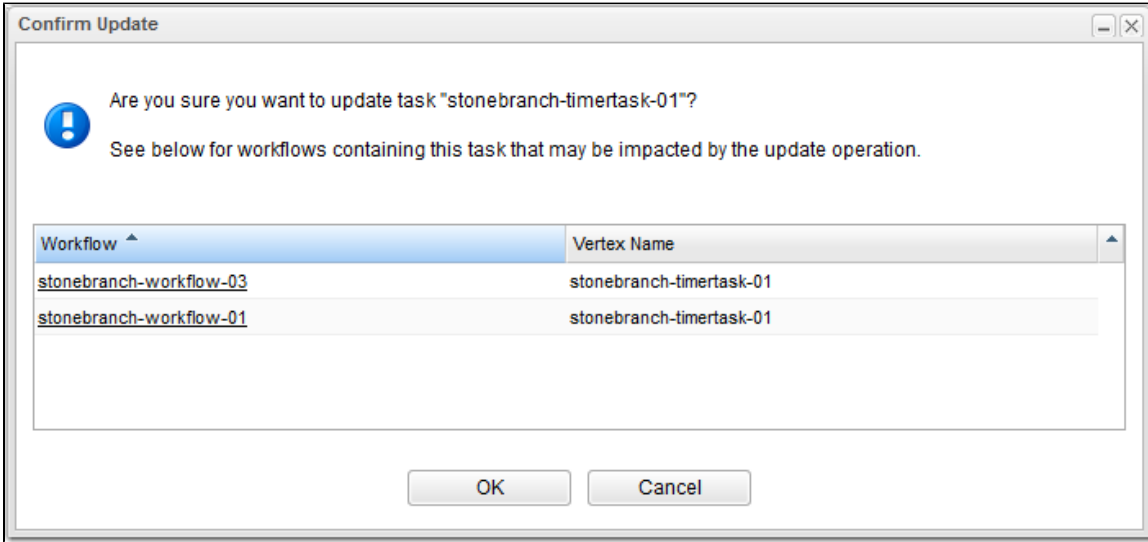
Transfer Mode (FTP/SFTP)					✓								
Transfer Mode (INFITRAN)						✓							
Transfer Mode (FTP File Monitor)							✓						
Transfer Type					✓	✓							
Trigger on Existence				✓									
Trim Trailing Spaces						✓							
Use JCL Override Library													
User Estimated Duration	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
User Estimated End Time		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Utility Agent						✓		✓			✓		
Utility Agent Cluster						✓		✓			✓		
Utility Agent Cluster Variable						✓		✓			✓		
Utility Agent Variable						✓		✓			✓		
Utility Credentials						✓		✓			✓		
Utility Credentials Variable						✓		✓			✓		
Value												✓	✓
Version	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Virtual Resource Priority			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Wait Until Satisfied							✓						
Wait Until Time													
Workflow Name Condition													
Workflow Name Contains													
Workflow Name Ends With													

Workflow Name Equals													
Workflow Name Starts With													

Updating Tasks

To save information that you have changed in a task, you must click the **Update** button that displays above and below the [Task Details](#).

If the task is contained in one or more Workflows, and the [Confirm Update For Tasks In Workflows](#) Opswise Controller system property value is set to **true**, the Confirm Update dialog displays when you click **Update**.



The Confirm Update dialog allows you to see which Workflows could be impacted by the update.

It displays, by default, two columns of information:

Workflow	Name of a Workflow that contain the task.
Vertex Name	Name of the task (or task alias) within the Workflow.

You also can display the following additional column by right-clicking either column header and selecting it from **Columns** on the [Action menu](#):

Vertex Id	ID of the task vertex within the Workflow.
------------------	--

Click the **OK** button to update the task, or click the **Cancel** button.



Note

You also can see this parent Workflows information for a task without updating a task (see [Viewing Task Parents](#)).

For information on updating multiple tasks, see [Updating Multiple Records](#).

Deleting Tasks

To delete a task, either:

- Right-click the task in a tasks list and, on the displayed [Action menu](#), click **Delete**.

- Open the task and click the **Delete** button.

**Note**

You cannot delete a task if it is either:

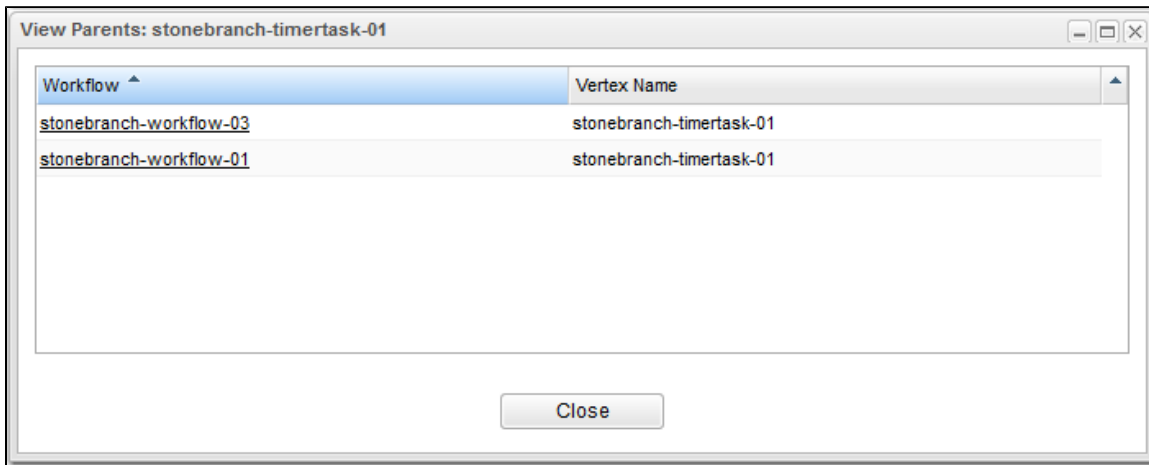
- Specified in an enabled Trigger.
- The only task specified in a disabled Trigger.

Viewing Task Parents

You can view any parent Workflow information for a task either by:

- Clicking the **View Parents** button in the [Task Details](#)
- Displaying the [Action menu](#) from the Task Details or Tasks list and then selecting [View Parents](#).

In either case, a View Parents dialog displays:



The View Parents dialog displays, by default, two columns of information:

Workflow	Name of a Workflow that contain the task.
Vertex Name	Name of the task (or task alias) within the Workflow.

You also can display the following additional column by right-clicking either column header and selecting it from **Columns** on the [Action menu](#):

Vertex Id	ID of the task vertex within the Workflow.
------------------	--

**Note**

It also is possible to see this parent Workflows information for a task when you are [updating a task](#).

Additional Information

The following pages provide additional information related to the creation of tasks:

- [Creating Task Actions](#)
- [Copying Tasks](#)
- [Setting Mutually Exclusive Tasks](#)
- [Creating Notes](#)

Linux Unix Task

- [Before You Begin](#)
- [Built-In Variables](#)
- [Creating a Linux/Unix Task](#)
 - [Linux/Unix Task Details](#)
 - [Linux/Unix Task Details Field Descriptions](#)
- [Viewing a Linux/Unix Task Instance](#)
 - [Linux/Unix Task Instance Details](#)
 - [Linux/Unix Task Instance Details Field Descriptions](#)
- [Running a Linux/Unix Task](#)
- [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](#) Opswise Universal Agent for Linux/Unix 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 Details Field Descriptions](#).

Built-In Variables

The following [built-in variables](#) can be used in a Linux/Unix task to pass data where appropriate:

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

Creating a Linux/Unix Task

Step 1 From the [Automation Center](#) navigation pane, select **Tasks > Linux/Unix Tasks**. The Linux/Unix Tasks list displays a list of all currently defined Linux/Unix tasks.

Below the list, Linux/Unix Task Details for a new Linux/Unix task displays. (You also can click the **New** button to display Linux/Unix Task Details for a new Linux/Unix task.)

The screenshot shows the 'Linux/Unix Tasks' interface. At the top, there is a list of 5 tasks with columns for Task Name, Task Description, Command or Script, Updated By, and Updated. Below the list is the 'Linux/Unix Task Details' form, which is currently in a collapsed state. The form has tabs for 'Linux/Unix Task', 'Variables', 'Actions', 'Virtual Resources', 'Mutually Exclusive', 'Instances', 'Triggers', 'Notes', and 'Versions'. The 'Linux/Unix Task' tab is active, showing fields for Task Name, Task Description, Member of Business Services, Hold on Start, Virtual Resource Priority, Hold Resources on Failure, and Agent Cluster.

Task Name	Task Description	Command or Script	Updated By	Updated
stonebranch-linuxunixtask-01		Command	ops.admin	2014-06-13 13:55:34 -0400
stonebranch-linuxunixtask-02		Command	ops.admin	2014-06-13 13:55:52 -0400
stonebranch-linuxunixtask-03		Command	ops.admin	2014-06-13 13:56:11 -0400
stonebranch-linuxunixtask-04		Command	ops.admin	2014-06-13 13:56:21 -0400
stonebranch-linuxunixtask-05		Command	ops.admin	2014-06-13 13:56:36 -0400

Step 2 Enter / select Details for a new Linux/Unix task, using the [field descriptions](#) below as a guide.

- Required fields display in **boldface**.
- Default values for fields, if available, display automatically.

To display more of the Details fields on the screen, you can temporarily [hide the list](#).



Note

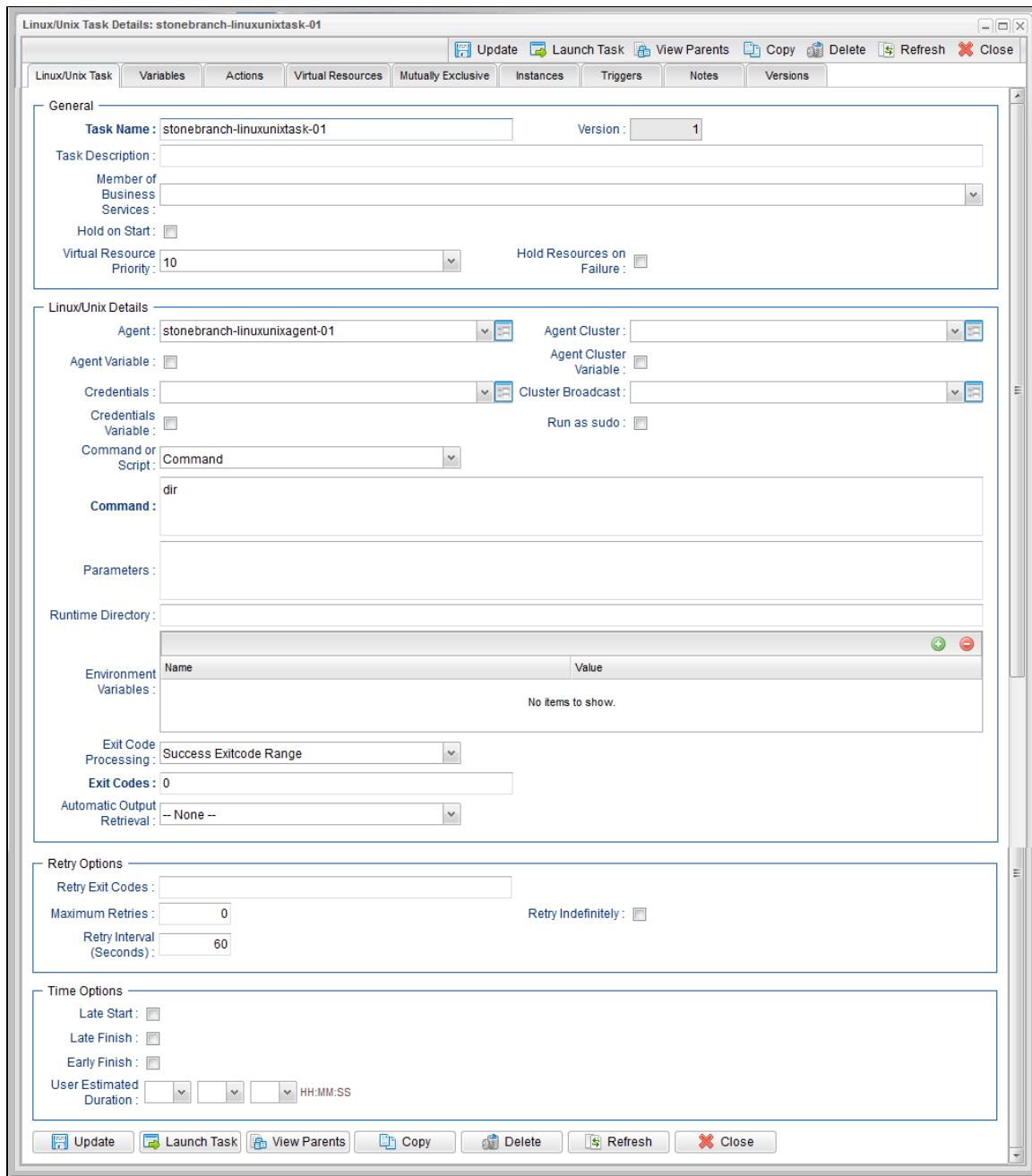
If you view [Linux/Unix Task Details](#) for an existing Linux/Unix task by clicking a task in the list, and then want to create a new Linux/Unix task, you must click the **New** button that displays above and below the Details.

Step 3 Click the **Save** button. The task is added to the database, and all buttons and tabs in the Task Details are enabled.

Linux/Unix Task Details

The following Linux/Unix Task Details is for an existing Linux/Unix task.



Depending on the values that you enter / select for these fields, and whether or not the Linux/Unix task has ever been launched, more (or less) fields may display. See the [field descriptions](#), below, for a description of all fields that may display in the Linux/Unix Task Details.




Linux/Unix Task Details Field Descriptions

The following table describes the fields, buttons, and tabs that display in the Linux/Unix Task Details.

Field Name	Description
General	This section contains general information about the task.
Task Name	User-defined name of this task, containing a maximum 255 alphanumeric characters; variables supported. It is the responsibility of the user to develop a workable naming scheme for tasks.
Version	System-supplied; version number of the current record, which is incremented by the Controller every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .


Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
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.
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.
Linux/Unix Details	This section contains assorted detailed information about the task.
Agent	Name of the Agent resource that identifies the machine where the operation will run. If you do not specify an Agent, you must specify an Agent Cluster .
Agent Cluster	Group of Agents, one of which the Controller will choose to run this task. You can specify an agent cluster in addition to or in place of a specific Agent. If you specify an Agent and an agent cluster, the Controller first tries to run the task on the specific agent. If the Agent is not available, the Controller reverts to the agent cluster. See Agent Clusters for more information.
Agent Variable	If enabled, 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. Use the format: <code>\${variable name}</code> . The variable must be a supported type as described in Variables and Functions . <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Agent Variable when updating multiple tasks, the Agent field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Agent Unresolved - into which you enter a variable, variable and text, or Agent name. The Agent field is not used.</p> </div>
Agent Cluster Variable	If enabled, 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. Use the format: <code>\${variable name}</code> . The variable must be a supported type as described in Variables and Functions . <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Agent Cluster Variable when updating multiple tasks, the Agent Cluster field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Agent Cluster Unresolved - into which you enter a variable, variable and text, or Agent Cluster name. The Agent Cluster field is not used.</p> </div>

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	<p>If enabled, the Credentials field 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> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Credentials Variable when updating multiple tasks, the Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Credentials Unresolved - into which you enter a variable, variable and text, or Credentials name. The Credentials field is not used.</p> </div>
Cluster Broadcast	Group of Agents, all of which will run this task. You can specify a Cluster Broadcast in place of a specific Agent and/or agent cluster. Each instance of the task running on its own Agent becomes a separate task instance record in the database and displays separately on the Activity screen. See Agent Clusters for more information about defining agent clusters.
Run as sudo	Run the command as Sudo (superuser do).
Command or Script	<p>Specifies whether a single command or a script is being executed.</p> <p>Options:</p> <ul style="list-style-type: none"> • Command (default) • Script
Script	If Command or Script = Script; Required. Name of the script in the Controller database that will be executed by this task.
Command	If Command or Script = Command; Required. Command being executed on the remote machine. Variables supported.
Parameters	Any arguments needed by the program to execute properly. Variables supported.
Runtime Directory	Directory from which the application should be executed. Variables supported.
Environment Variables	<p>Allows you to enter environment variables needed by the program to run.</p> <p>To add a variable, click the + icon and enter a Name and Value. To delete a variable, select in the list of variables and click the - icon.</p> <p>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.</p>

Exit Code Processing	<p>Specifies how the Controller should determine whether the executed command failed or completed successfully.</p> <p>Options:</p> <ul style="list-style-type: none"> • Success Exitcode Range - Command is considered completed successfully if its exit code falls within the range specified in the Exit Codes field. • Failure Exitcode Range - Command is considered failed if its exit code falls within the range specified in the Exit Codes field. • Success Output Contains - Command is considered completed successfully if its output contains the text specified in the Scan Output For field. • Failure Output Contains - Command is considered failed if its output contains the text specified in the Scan Output For field. • Step Conditions (z/OS only) - 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	<p>Required if Exit Code Processing = Success Output Contains or Failure Output Contains; type of output.</p> <p>Options:</p> <ul style="list-style-type: none"> • Standard Output (STDOUT) • Standard Error (STDERR) • File
Scan Output For	<p>Required if Exit Code Processing = Success Output Contains or Failure Output Contains; text for which the Controller should scan the output file. The Controller will process this field as a regular expression.</p>
Output File (for Exit Code Processing)	<p>Required if Output Type = File; path and file name of the output file that should be scanned for the text in the Scan Output For field.</p>
Exit Codes	<p>Required if Exit Code Processing = Success Exitcode Range or Failure Exitcode Range; range of exit codes. Format: Numeric. Use commas to list a series of exit codes; use hyphens to specify a range. Example: 1,5, 22-30.</p>
Automatic Output Retrieval	<p>Specifies whether you want the Controller to automatically retrieve any output from the job and attach it to the task instance record.</p> <p>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. • Standard Output/Error - Attach all standard output and standard error output.
Start Line	<p>If Automatic Output Retrieval = Standard Output, Standard Error, File, or Standard Output/Error; Instructs the Controller to retrieve data beginning at the line indicated.</p> <ul style="list-style-type: none"> • If a Start Line value is not specified, the default is 1. • If the Start Line value is -1, data will be retrieved starting at the end of the file.
Number of Lines	<p>If Automatic Output Retrieval = Standard Output, Standard Error, File, or Standard Output/Error; 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 Controller system property.</p>

Scan Text	<p>If Automatic Output Retrieval = Standard Output, Standard Error, File, or Standard Output/Error; Regex pattern that the Controller will search for a match for in STDOUT/STDERR or a specified file. The Controller will include the Number of Lines above and below the first line matched.</p> <p>if the Regex pattern is not found, the following message is returned: OPSWISE WARNING - Scan text string not found.</p>
Output File (for Automatic Output Retrieval)	Required if Automatic Output Retrieval = File; path and file name containing the output that you want automatically retrieved and attached to the task instance.
Retry Options	This section contains specifications for retrying the task.
Retry Exit Codes	<p>Exit code range for which an auto-retry of tasks in FAILED status will occur. Exit code ranges must be in the same format as ranges specified in the Exit Codes field. Maximum Retries must be greater than 0.</p> <p>If this field is empty, any exit code potentially will cause a retry.</p>
Maximum Retries	User-defined; maximum number of times that the Controller should retry this task after it has started and gone to a failed state.
Retry Indefinitely	User-defined; indicates whether the Controller should continue trying indefinitely to run this task. If you enable this field, it overrides any value placed in the Maximum Retries field.
Retry Interval (Seconds)	User-defined; number of seconds between each retry.
Time Options	This section contains time-related specifications for the task.
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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.
Late Start Type	<p>Required if Late Start is enabled.</p> <p>Options:</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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.
Late Start Duration	<p>If Late Start Type = 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.</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>

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: <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 = Time; Time after which the task finish time is considered late. Use HH:MM, 24-hour time.
Late Finish Duration	If Late Finish Type = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
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: <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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
User Estimated Duration	Estimated amount of time it should normally take to run this task. The Controller uses this information to calculate the User Estimated End Time on a task instance record.
Statistics	This section contains time-related statistics for task instances of the task.
First Time Ran	System-supplied; date and time this task first ran.
Last Time Ran	System-supplied; date and time the task last ran.
Last Instance Duration	System-supplied; amount of time the task took to run the last time it ran.

Lowest Instance Time	System-supplied; shortest amount of time this task has taken to run.
Average Instance Time	System-supplied; average amount of time this task takes to run.
Highest Instance Time	System-supplied; longest amount of time this task has taken to run.
Number of Instances	System-supplied; number of instances in the database for this task.
Buttons	This section identifies the buttons displayed above and below the Task Details that let you perform various actions.
Save	Saves a new task record in the Controller database.
Update	Saves updates to the record.
New	Displays empty (except for default values) Details for creating a new task.
Launch Task	Manually launches the task.
View Parents	Displays a list of any parent Workflow tasks for this task.
Copy	Creates a copy of this task, which you are prompted to rename.
Delete	<p>Deletes the current record.</p> <div style="background-color: #ffffcc; padding: 10px; margin: 10px 0;"> <p> Note You cannot delete a task if it is either:</p> <ul style="list-style-type: none"> Specified in an enabled Trigger. The only task specified in a disabled Trigger. </div>
Refresh	Refreshes any dynamic data displayed in the Details.
Close	For pop-up view only; closes the pop-up view of this task.
Tabs	This section identifies the tabs across the top of the Task Details that provide access to additional information about the task.
Variables	Lists all variables associated with this record.

Actions	<p>Allows you to specify actions that the Controller will take automatically based on events that occur during the execution of this task.</p> <p>Events are:</p> <ul style="list-style-type: none"> • Task instance status • Exit codes • Late start • Late finish • Early finish <p>Actions are:</p> <table border="1" data-bbox="371 478 1484 840"> <tr> <td data-bbox="371 478 511 548">Abort Action</td> <td data-bbox="511 478 1484 548">Abort the task if certain events occur. For details, see Abort Actions.</td> </tr> <tr> <td data-bbox="371 548 511 617">Email Notification</td> <td data-bbox="511 548 1484 617">Send an email if certain events occur. For details, see Email Notification Actions.</td> </tr> <tr> <td data-bbox="371 617 511 686">Set Variable</td> <td data-bbox="511 617 1484 686">Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow.</td> </tr> <tr> <td data-bbox="371 686 511 756">SNMP Notification</td> <td data-bbox="511 686 1484 756">Send an email if certain events occur. For details, see SNMP Notification Actions.</td> </tr> <tr> <td data-bbox="371 756 511 825">System Operation</td> <td data-bbox="511 756 1484 825">Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions.</td> </tr> </table>	Abort Action	Abort the task if certain events occur. For details, see Abort Actions .	Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .	Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .	SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .	System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .
Abort Action	Abort the task if certain events occur. For details, see Abort Actions .										
Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .										
Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .										
SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .										
System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .										
Virtual Resources	Lists all Virtual Resources to which this task is assigned.										
Mutually Exclusive	Lists all tasks that have been set to be mutually exclusive of this task.										
Instances	Lists all instances of the task.										
Triggers	Lists 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, the Controller 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 Triggers Overview .										
Notes	Lists all notes associated with this record.										
Versions	Stores copies of all previous versions of the current record. See Record Versioning .										

Viewing a Linux/Unix Task Instance

When a Linux/Unix task is launched, the Controller creates a task instance record of that task.

A task instance contains detailed information about a single execution of that task.

You can access a task instance from:

- **Instances tab** on the [Linux/Unix Task Details](#) for that task
- [Activity Monitor](#)
- [Task Instances list](#)

Linux/Unix Task Instance Details

The following Linux/Unix Task Instance Details contains information on the execution of the task shown in the [Linux/Unix Task Details](#).

Linux/Unix Task Instance Details: stonebranch-linuxunixtask-01

Linux/Unix Task Instance
Virtual Resources
Exclusive Requests
Output
Notes

General

Instance Name: Reference Id:

Task: Invoked By:

Task Description:

Member of Business Services: Execution User:

Virtual Resource Priority: Hold Resources on Failure:

Status

Status: Exit Code:

Status Description:

Queued Time:

Trigger Time:

Start Time:

Launch Time: End Time:

Duration: CPU Time:

Process ID:

Linux/Unix Details

Agent: Agent Cluster:

Agent Variable: Agent Cluster Variable:

Credentials: Credentials Variable:

Run as sudo:

Command or Script:

Command:

Parameters:

Runtime Directory:

Environment Variables:

Name	Value
No items to show.	

Exit Code Processing:

Exit Codes:

Automatic Output Retrieval:

Start Line: Number of Lines:

Scan Text:

Retry Options

Retry Exit Codes:

Maximum Retries: Retry Indefinitely:

Retry Interval (Seconds): Current Retry Count:

Statistics

User Estimated End Time: Average Estimated End Time:



Shortest Estimated End Time: Longest Estimated End Time:


Linux/Unix Task Instance Details Field Descriptions

The following table describes the fields, buttons, and tabs that display in Linux/Unix Task Instance Details.

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

General	This section contains general information about the task instance.
Instance Name	Name of this task instance.
Reference Id	System-supplied; the Controller increments this number each time the task is run.
Task	Name of the task that was run to create this task instance. Click the icon to display Task Details for the task.
Invoked by	System-supplied; how the task instance was launched. Options: <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 identify the user, check the Execution User column for that task instance on the Task Instances screen or, on most task instance screens, the Execution User field.
Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
Execution User	System-supplied; if the task was launched manually, the ID of the user who launched it.
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.
Status	This section contains information about the current status of the task instance.
Status	System-supplied; see Task Instance Statuses .
Exit Code	System-supplied; the exit code captured by the Agent when executing the task (for example, a command or script).
Status Description	System-supplied; additional information, if any, about the status of the task instance.
Queued Time	System-supplied; Date and time the task was queued for processing.
Trigger Time	System-supplied; Date and time the task instance was triggered.
Launch Time	System-supplied; Date and time the task instance was launched.

Start Time	System-supplied; Date and time the task instance started.
End Time	System-supplied; Date and time the task instance completed.
Duration	System-supplied; amount of time the task instance took to run.
CPU Time	System-supplied; amount of CPU time the task took to run.
Process ID	system-supplied; the ID of the process that was launched.
Linux/Unix Details	This section contains assorted detailed information about the task instance.
Agent	Name of the Agent resource that identifies the machine where the operation will run. If you do not specify an Agent, you must specify an Agent Cluster .
Agent Variable	<p>If enabled, 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. Use the format: <code>\${variable name}</code>. The variable must be a supported type as described in Variables and Functions.</p> <div style="background-color: #ffffcc; padding: 10px; margin: 10px 0;"> <p> Note If you select Agent Variable when updating multiple tasks, the Agent field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Agent Unresolved - into which you enter a variable, variable and text, or Agent name. The Agent field is not used.</p> </div>
Agent Cluster	Group of Agents, one of which the Controller will choose to run this task. You can specify an agent cluster in addition to or in place of a specific Agent. If you specify an Agent and an agent cluster, the Controller first tries to run the task on the specific agent. If the Agent is not available, the Controller reverts to the agent cluster. See Agent Clusters for more information.
Agent Cluster Variable	<p>If enabled, 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. Use the format: <code>\${variable name}</code>. The variable must be a supported type as described in Variables and Functions.</p> <div style="background-color: #ffffcc; padding: 10px; margin: 10px 0;"> <p> Note If you select Agent Cluster Variable when updating multiple tasks, the Agent Cluster field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Agent Cluster Unresolved - into which you enter a variable, variable and text, or Agent Cluster name. The Agent Cluster field is not used.</p> </div>
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	<p>If enabled, the Credentials field 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> <div style="background-color: #ffffcc; padding: 10px; margin: 10px 0;"> <p> Note If you select Credentials Variable when updating multiple tasks, the Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Credentials Unresolved - into which you enter a variable, variable and text, or Credentials name. The Credentials field is not used.</p> </div>
Run as sudo	Run the command as Sudo (superuser do).
Command or Script	<p>Specifies whether a single command or a script is being executed.</p> <p>Options:</p> <ul style="list-style-type: none"> • Command (default) • Script
Script	If Command or Script = Script; Required. Name of the script in the Controller database that will be executed by this task.
Command	If Command or Script = Command; Required. Command being executed on the remote machine. Variables supported.
Parameters	Any arguments needed by the program to execute properly. Variables supported.
Runtime Directory	Directory from which the application should be executed. Variables supported.
Environment Variables	<p>Allows you to enter environment variables needed by the program to run.</p> <p>To add a variable, click the + icon and enter a Name and Value. To delete a variable, select in the list of variables and click the - icon.</p> <p>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.</p>
Exit Code Processing	<p>Specifies how the Controller should determine whether the executed command failed or completed successfully.</p> <p>Options:</p> <ul style="list-style-type: none"> • Success Exitcode Range - Command is considered completed successfully if its exit code falls within the range specified in the Exit Codes field. • Failure Exitcode Range - Command is considered failed if its exit code falls within the range specified in the Exit Codes field. • Success Output Contains - Command is considered completed successfully if its output contains the text specified in the Scan Output For field. • Failure Output Contains - Command is considered failed if its output contains the text specified in the Scan Output For field. • Step Conditions (z/OS only) - 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	<p>Required if Exit Code Processing = Success Output Contains or Failure Output Contains; type of output.</p> <p>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 Exitcode Range; range of exit codes. Format: Numeric. Use commas to list a series of exit codes; 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; text for which the Controller should scan the output file. The Controller will process this field as a regular expression.</p>
Output File (for Exit Code Processing)	<p>Required if Output Type = File; path and file name of the output file that should be scanned for the text in the Scan Output For field.</p>
Automatic Output Retrieval	<p>Specifies whether you want the Controller to automatically retrieve any output from the job and attach it to the task instance record.</p> <p>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. • Standard Output/Error - Attach all standard output and standard error output.
Start Line	<p>If Automatic Output Retrieval = Standard Output, Standard Error, File, or Standard Output/Error; Instructs the Controller to retrieve data beginning at the line indicated.</p> <ul style="list-style-type: none"> • If a Start Line value is not specified, the default is 1. • If the Start Line value is -1, data will be retrieved starting at the end of the file.
Number of Lines	<p>If Automatic Output Retrieval = Standard Output, Standard Error, File, or Standard Output/Error; 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 Controller system property.</p>
Scan Text	<p>If Automatic Output Retrieval = Standard Output, Standard Error, File, or Standard Output/Error; Regex pattern that the Controller will search for a match for in STDOUT/STDERR or a specified file. The Controller will include the Number of Lines above and below the first line matched.</p> <p>if the Regex pattern is not found, the following message is returned: OPSWISE WARNING - Scan text string not found.</p>
Output File (for Automatic Output Retrieval)	<p>Required if Automatic Output Retrieval = File; path and file name containing the output that you want automatically retrieved and attached to the task instance.</p>
Retry Options	<p>This section contains specifications for retrying the task.</p>
Retry Exit Codes	<p>Exit code range for which an auto-retry of tasks in FAILED status will occur. Exit code ranges must be in the same format as ranges specified in the Exit Codes field. Maximum Retries must be greater than 0.</p> <p>If this field is empty, any exit code potentially will cause a retry.</p>

Maximum Retries	User-defined; maximum number of times that the Controller should retry this task after it has started and gone to a failed state.
Retry Indefinitely	User-defined; indicates whether the Controller should continue trying indefinitely to run this task. If you enable this field, it overrides any value placed in the Maximum Retries field.
Retry Interval (Seconds)	User-defined; number of seconds between each retry.
Current Retry Count	System-supplied; current number of times that the Controller has retried the task after it first went to failure status.
Time Options	This section contains time-related specifications for the task instance.
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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.
Started Late	System-supplied; this field is flagged if the task started later than the time specified in the Late Start fields.
Late Start Type	Required if Late Start is enabled. Options: <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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.
Late Start Duration	If Late Start Type = 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.
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.
Finished Late	System-supplied; this field is flagged if the task finished later than the time or duration specified in the Late Finish fields.

Late Finish Type	<p>Required if Late Finish is enabled.</p> <p>Options:</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 = Time; Time after which the task finish time is considered late. Use HH:MM, 24-hour time.
Late Finish Duration	If Late Finish Type = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
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.
Finished Early	System-supplied; this field is flagged if the task finished earlier than the time specified in the Early Finish fields.
Early Finish Type	<p>Required if Early Finish is enabled.</p> <p>Options:</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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Statistics	This section contains time-related statistics for the the task instance.
User Estimated End Time	System-supplied; if the user entered information into the User Estimated Duration field in the task Details, the Controller 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	System-supplied; shortest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Average Estimated End Time	System-supplied; average estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Longest Estimated End Time	System-supplied; longest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Buttons	This section identifies the buttons displayed above and below the Task Instance Details that let you perform various actions.

Update	Saves updates to the record.
Force Finish	See Force Finishing a Task .
Hold	Places the task instance on Hold (see Putting a Task on Hold).
Skip	For tasks loaded into the schedule that have not yet run; allows you to tell the Controller to skip this task. See Skipping a Task .
Re-run	See Re-running a Task .
View Parent	Displays the task instance Details for the parent Workflow of this task instance.
Retrieve Output	See Retrieving Output .
Delete	Deletes the current record.
Refresh	Refreshes any dynamic data displayed in the Details.
Close	For pop-up view only; closes the pop-up view of this task instance.
Tabs	This section identifies the tabs across the top of the Task Instance Details that provide access to additional information about the task instance.
Virtual Resources	Lists all Virtual Resources to which this task is assigned.
Exclusive Requests	Lists all records in the Exclusive Requests table (<code>ops_exclusive_order</code>) for this task instance.
Output	Displays output generated from the process, if any, based on specifications provided by the user in the Automatic Output Retrieval fields in the task Details. If automatic output retrieval was not available or was not selected, output can be obtained by clicking the Retrieve Output button.
Notes	Lists all notes associated with this record.

Running a Linux/Unix Task

You can run a Linux/Unix task:

- Manually, by clicking the [Launch Task](#) or [Launch Task with Variables](#) button in the Linux/Unix tasks list or Linux/Unix Task Details [Action menu](#).
- As part of a [Workflow](#).
- [Specify triggers](#) that run the task automatically based on times or events.

Monitoring Task Execution

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

Windows Task

- [Before You Begin](#)
- [Built-In Variables](#)
- [Creating a Windows Task](#)
 - [Windows Task Details](#)
 - [Windows Task Details Field Descriptions](#)
- [Viewing a Windows Task Instance](#)
 - [Windows Task Instance Details](#)
 - [Windows Task Instance Details Field Descriptions](#)
- [Running a Windows Task](#)
- [Issuing Commands via cscript.exe](#)
- [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](#) Opswise Universal Agent for Windows 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 Details Field Descriptions](#).

Built-In Variables

The following [built-in variables](#) can be used in a Windows task to pass data where appropriate:

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

Creating a Windows Task

Step 1 From the [Automation Center](#) navigation pane, select **Tasks > Windows Tasks**. The Windows Tasks list displays a list of all currently defined Windows tasks.

Below the list, Windows Task Details for a new Windows task displays. (You also can click the **New** button to display Windows Task Details for a new Windows task.)

The screenshot shows the 'Windows Tasks' interface. At the top, there is a 'Windows Tasks' tab with a dropdown menu showing '5 Windows Tasks'. Below this is a table with columns: Task Name, Task Description, Command or Script, Updated By, and Updated. The table contains five rows of tasks, all named 'stonebranch-windowstask-01' through '05', with 'Command' as the script type and 'ops.admin' as the updated by user. Below the table is the 'Windows Task Details' form, which has tabs for 'Windows Task', 'Variables', 'Actions', 'Virtual Resources', 'Mutually Exclusive', 'Instances', 'Triggers', 'Notes', and 'Versions'. The 'Windows Task' tab is active, showing fields for 'Task Name', 'Version' (set to 1), 'Task Description', 'Member of Business Services', 'Hold on Start', 'Virtual Resource Priority' (set to 10), 'Hold Resources on Failure', 'Agent', 'Agent Cluster', 'Agent Variable', and 'Agent Cluster Variable'.

Step 2 Enter/select Details for a new Windows task, using the field descriptions below as a guide.

- Required fields display in **boldface**.
- Default values for fields, if available, display automatically.

To display more of the Details fields on the screen, you can temporarily [hide the list](#).



Note

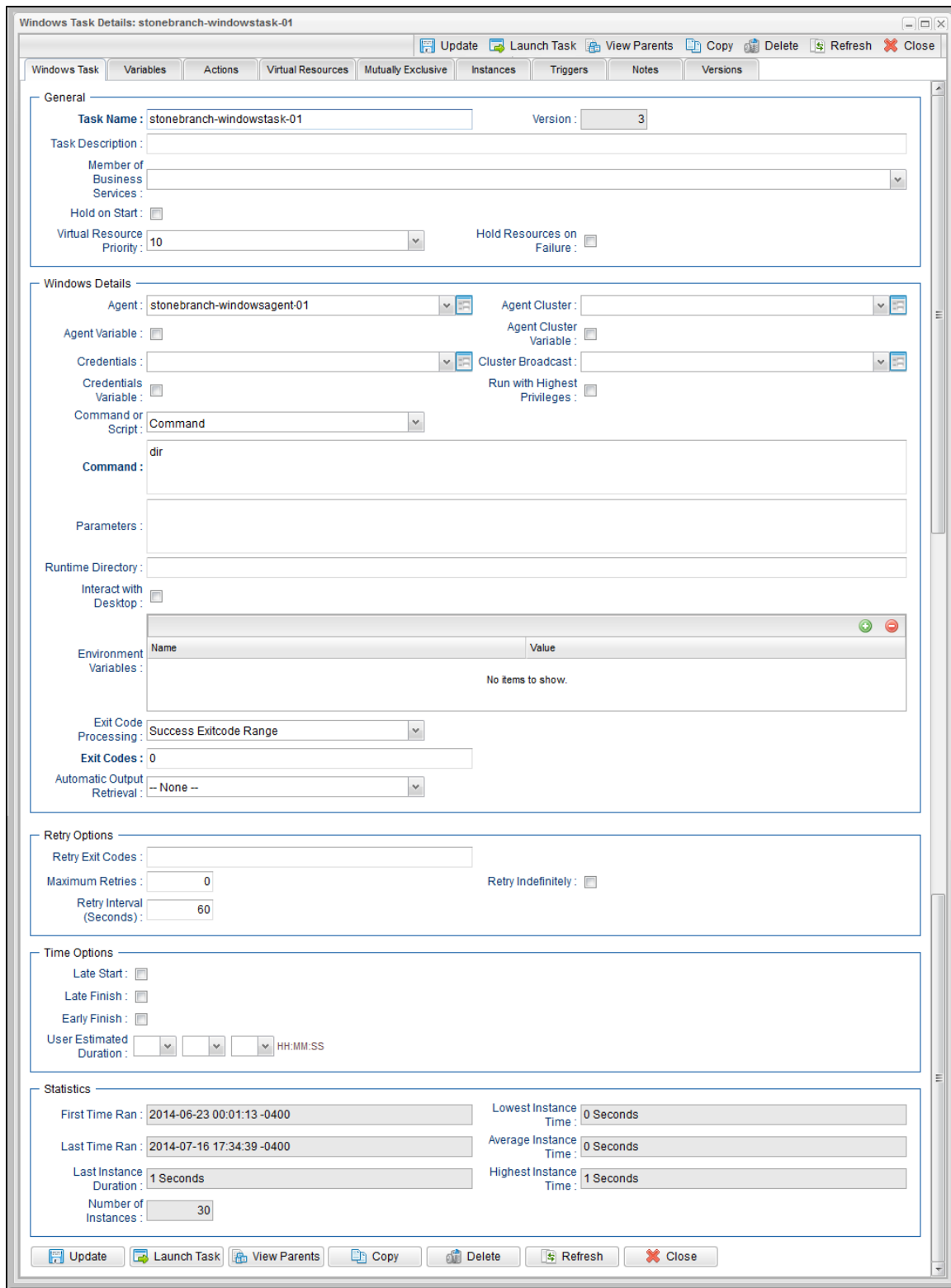
If you view [Windows Task Details](#) for an existing Windows task by clicking a task in the list, and then want to create a new Windows task, you must click the **New** button that displays above and below the Details.

Step 3 Click the **Save** button. The task is added to the database, and all buttons and tabs in the Task Details are enabled.

Windows Task Details

The following Windows Task Details is for an existing Windows task.


Depending on the values that you enter / select for these fields, and whether or not the Windows task has ever been launched, more (or less) fields may display. See the [field descriptions](#), below, for a description of all fields that may display in the Windows Task Details.








Windows Task Details Field Descriptions

The following table describes the fields, buttons, and tabs that display in the Windows Task Details.

Field Name	Description
General	This section contains general information about the task.

Task Name	User-defined name of this task, containing a maximum 255 alphanumeric characters; variables supported. It is the responsibility of the user to develop a workable naming scheme for tasks.
Version	System-supplied; version number of the current record, which is incremented by the Controller every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
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.
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.
Windows Details	This section contains assorted detailed information about the task.
Agent	Name of the Agent resource that identifies the machine where the operation will run. If you do not specify an Agent, you must specify an Agent Cluster .
Agent Cluster	Group of Agents, one of which the Controller will choose to run this task. You can specify an agent cluster in addition to or in place of a specific Agent. If you specify an Agent and an agent cluster, the Controller first tries to run the task on the specific agent. If the Agent is not available, the Controller reverts to the agent cluster. See Agent Clusters for more information.
Agent Variable	If enabled, 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. Use the format: \${variable name}. The variable must be a supported type as described in Variables and Functions . <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;">  Note If you select Agent Variable when updating multiple tasks, the Agent field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Agent Unresolved - into which you enter a variable, variable and text, or Agent name. The Agent field is not used. </div>


Agent Cluster Variable	<p>If enabled, 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. Use the format: <code>\${variable name}</code>. The variable must be a supported type as described in Variables and Functions.</p> <div data-bbox="412 283 1445 464" style="background-color: #ffffcc; padding: 10px;"> <p> Note If you select Agent Cluster Variable when updating multiple tasks, the Agent Cluster field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Agent Cluster Unresolved - into which you enter a variable, variable and text, or Agent Cluster name. The Agent Cluster field is not used.</p> </div>
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>
Credentials Variable	<p>If enabled, the Credentials field 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> <div data-bbox="412 768 1445 949" style="background-color: #ffffcc; padding: 10px;"> <p> Note If you select Credentials Variable when updating multiple tasks, the Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Credentials Unresolved - into which you enter a variable, variable and text, or Credentials name. The Credentials field is not used.</p> </div>
Cluster Broadcast	<p>Group of Agents, all of which will run this task. You can specify a Cluster Broadcast in place of a specific Agent and/or agent cluster. Each instance of the task running on its own Agent becomes a separate task instance record in the database and displays separately on the Activity screen. See Agent Clusters for more information about defining agent clusters.</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 data-bbox="412 1400 1445 1556" style="background-color: #ffffcc; padding: 10px;"> <p> Note This option only will 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> </div>

Command or Script	<p>Specifies whether a single command or a script is being executed.</p> <p>Options:</p> <ul style="list-style-type: none"> • Command (default) • Script <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note For both command-based tasks that call a .vbs/.js file directly, and script-based tasks that also rely on the systems association with file extension, GUI-based wscript.exe is associated with the vbs and js file extensions. Without explicitly calling one or the other, the Controller would use wscript.exe.</p> <p>The Agent system may need to be adjusted to properly use the Windows Scripting Host from the scheduler/agent environment.</p> <p>The following command can be used to set the default script host to cscript.exe: <code>C:\tmp>cscript //h:cscript //s</code></p> </div>
Command	If Command or Script = Command; Required. Command being executed on the remote machine. Variables supported.
Script	If Command or Script = Script; Required. Name of the script in the Controller database that will be executed by this task.
Parameters	Any arguments needed by the program to execute properly. Variables supported.
Runtime Directory	Directory from which the application should be executed. Variables supported.
Interact with Desktop	<p>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).</p> <div 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, Server 2008, Windows 8, and Server 2012. 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>
Environment Variables	<p>Allows you to enter environment variables needed by the program to run.</p> <p>To add a variable, click the + icon and enter a Name and Value. To delete a variable, select in the list of variables and click the - icon.</p> <p>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.</p>

Exit Code Processing	<p>Specifies how the Controller should determine whether the executed command failed or completed successfully.</p> <p>Options:</p> <ul style="list-style-type: none"> • Success Exitcode Range - Command is considered completed successfully if its exit code falls within the range specified in the Exit Codes field. • Failure Exitcode Range - Command is considered failed if its exit code falls within the range specified in the Exit Codes field. • Success Output Contains - Command is considered completed successfully if its output contains the text specified in the Scan Output For field. • Failure Output Contains - Command is considered failed if its output contains the text specified in the Scan Output For field. • Step Conditions (z/OS only) - 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	<p>Required if Exit Code Processing = Success Output Contains or Failure Output Contains; type of output.</p> <p>Options:</p> <ul style="list-style-type: none"> • Standard Output (STDOUT) • Standard Error (STDERR) • File
Scan Output For	<p>Required if Exit Code Processing = Success Output Contains or Failure Output Contains; text for which the Controller should scan the output file. The Controller will process this field as a regular expression.</p>
Output File (for Exit Code Processing)	<p>Required if Output Type = File; path and file name of the output file that should be scanned for the text in the Scan Output For field.</p>
Exit Codes	<p>Required if Exit Code Processing = Success Exitcode Range or Failure Exitcode Range; range of exit codes. Format: Numeric. Use commas to list a series of exit codes; use hyphens to specify a range. Example: 1,5, 22-30.</p>
Automatic Output Retrieval	<p>Specifies whether you want the Controller to automatically retrieve any output from the job and attach it to the task instance record.</p> <p>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. • Standard Output/Error - Attach all standard output and standard error output.
Start Line	<p>If Automatic Output Retrieval = Standard Output, Standard Error, File, or Standard Output/Error; Instructs the Controller to retrieve data beginning at the line indicated.</p> <ul style="list-style-type: none"> • If a Start Line value is not specified, the default is 1. • If the Start Line value is -1, data will be retrieved starting at the end of the file.
Number of Lines	<p>If Automatic Output Retrieval = Standard Output, Standard Error, File, or Standard Output/Error; 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 Controller system property.</p>

Scan Text	<p>If Automatic Output Retrieval = Standard Output, Standard Error, File, or Standard Output/Error; Regex pattern that the Controller will search for a match for in STDOUT/STDERR or a specified file. The Controller will include the Number of Lines above and below the first line matched.</p> <p>if the Regex pattern is not found, the following message is returned: OPSWISE WARNING - Scan text string not found.</p>
Output File (for Automatic Output Retrieval)	Required if Automatic Output Retrieval = File; path and file name containing the output that you want automatically retrieved and attached to the task instance.
Retry Options	This section contains specifications for retrying the task.
Retry Exit Codes	<p>Exit code range for which an auto-retry of tasks in FAILED status will occur. Exit code ranges must be in the same format as ranges specified in the Exit Codes field. Maximum Retries must be greater than 0.</p> <p>If this field is empty, any exit code potentially will cause a retry.</p>
Maximum Retries	User-defined; maximum number of times that the Controller should retry this task after it has started and gone to a failed state.
Retry Indefinitely	User-defined; indicates whether the Controller should continue trying indefinitely to run this task. If you enable this field, it overrides any value placed in the Maximum Retries field.
Retry Interval (Seconds)	User-defined; number of seconds between each retry.
Time Options	This section contains time-related specifications for the task.
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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.
Late Start Type	<p>Required if Late Start is enabled.</p> <p>Options:</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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.
Late Start Duration	<p>If Late Start Type = 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.</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>

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: <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 = Time; Time after which the task finish time is considered late. Use HH:MM, 24-hour time.
Late Finish Duration	If Late Finish Type = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
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: <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 = Time; 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.
User Estimated Duration	Estimated amount of time it should normally take to run this task. The Controller uses this information to calculate the User Estimated End Time on a task instance record.
Statistics	This section contains time-related statistics for task instances of the task.
First Time Ran	System-supplied; date and time this task first ran.
Last Time Ran	System-supplied; date and time the task last ran.
Last Instance Duration	System-supplied; amount of time the task took to run the last time it ran.
Lowest Instance Time	System-supplied; shortest amount of time this task has taken to run.
Average Instance Time	System-supplied; average amount of time this task takes to run.

Highest Instance Time	System-supplied; longest amount of time this task has taken to run.
Number of Instances	System-supplied; number of instances in the database for this task.
Buttons	This section identifies the buttons displayed above and below the Task Details that let you perform various actions.
Save	Saves a new task record in the Controller database.
Update	Saves updates to the record.
New	Displays empty (except for default values) Details for creating a new task.
Launch Task	Manually launches the task.
View Parents	Displays a list of any parent Workflow tasks for this task.
Copy	Creates a copy of this task, which you are prompted to rename.
Delete	<p>Deletes the current record.</p> <div style="background-color: #ffffcc; padding: 10px; margin: 10px 0;"> <p> Note You cannot delete a task if it is either:</p> <ul style="list-style-type: none"> Specified in an enabled Trigger. The only task specified in a disabled Trigger. </div>
Refresh	Refreshes any dynamic data displayed in the Details.
Close	For pop-up view only; closes the pop-up view of this task.
Tabs	This section identifies the tabs across the top of the Task Details that provide access to additional information about the task.
Variables	Lists all variables associated with this record.

Actions	<p>Allows you to specify actions that the Controller will take automatically based on events that occur during the execution of this task.</p> <p>Events are:</p> <ul style="list-style-type: none"> • Task instance status • Exit codes • Late start • Late finish • Early finish <p>Actions are:</p> <table border="1" data-bbox="370 478 1481 842"> <tr> <td data-bbox="370 478 509 548">Abort Action</td> <td data-bbox="509 478 1481 548">Abort the task if certain events occur. For details, see Abort Actions.</td> </tr> <tr> <td data-bbox="370 548 509 617">Email Notification</td> <td data-bbox="509 548 1481 617">Send an email if certain events occur. For details, see Email Notification Actions.</td> </tr> <tr> <td data-bbox="370 617 509 686">Set Variable</td> <td data-bbox="509 617 1481 686">Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow.</td> </tr> <tr> <td data-bbox="370 686 509 756">SNMP Notification</td> <td data-bbox="509 686 1481 756">Send an email if certain events occur. For details, see SNMP Notification Actions.</td> </tr> <tr> <td data-bbox="370 756 509 825">System Operation</td> <td data-bbox="509 756 1481 825">Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions.</td> </tr> </table>	Abort Action	Abort the task if certain events occur. For details, see Abort Actions .	Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .	Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .	SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .	System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .
Abort Action	Abort the task if certain events occur. For details, see Abort Actions .										
Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .										
Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .										
SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .										
System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .										
Virtual Resources	Lists all Virtual Resources to which this task is assigned.										
Mutually Exclusive	Lists all tasks that have been set to be mutually exclusive of this task.										
Instances	Lists of all instances of the task.										
Triggers	Lists 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, the Controller 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 Triggers Overview .										
Notes	Lists all notes associated with this record.										
Versions	Stores copies of all previous versions of the current record. See Record Versioning .										

Viewing a Windows Task Instance

When a Windows task is launched, the Controller creates a task instance record of that task.

A task instance contains detailed information about a single execution of that task.

You can access a task instance from:

- **Instances tab** on the [Windows Task Details](#) for that task
- [Activity Monitor](#)
- [Task Instances list](#)

Windows Task Instance Details

The following Windows Task Instance Details contains information on the execution of the task shown in the [Windows Task Details](#).

Windows Task Instance Details: stonebranch-windowstask-01

Update Re-run View Parent Retrieve Output... Delete Refresh Close

Windows Task Instance Virtual Resources Exclusive Requests Output Notes

General

Instance Name: stonebranch-windowstask-01 Reference Id: 14

Task: stonebranch-windowstask-01 Invoked By: Trigger: stonebranch-timetrigger-01

Task Description:

Member of Business Services: Execution User: stonebranch-user-01

Virtual Resource Priority: 10 Hold Resources on Failure:

Status

Status: Success Exit Code: 0

Status Description:

Queued Time: 2015-02-06 14:45:37 -0500

Trigger Time: Start Time: 2015-02-06 14:45:41 -0500

Launch Time: 2015-02-06 14:45:37 -0500 End Time: 2014-06-20 13:09:49 -0400

Duration: 0 Seconds CPU Time: 0

Process ID: 22872

Windows Details

Agent: stonebranch-windowsagent-01 Agent Cluster:

Agent Variable: Agent Cluster Variable:

Credentials: Credentials Variable: Run with Highest Privileges:

Command or Script: Command

Command: dir

Parameters:

Runtime Directory:

Interact with Desktop:

Environment Variables:

Name	Value
No items to show.	

Exit Code Processing: Success Exitcode Range

Exit Codes: 0

Automatic Output Retrieval: -- None --

Retry Options

Retry Exit Codes:

Maximum Retries: 0 Retry Indefinitely:

Retry Interval (Seconds): 60 Current Retry Count: 0

Statistics

User Estimated End Time: Average Estimated End Time: 2014-07-22 11:09:33 -0400

Shortest Estimated End Time: 2014-07-22 11:09:33 -0400 Longest Estimated End Time: 2014-07-22 11:09:34 -0400



Update Re-run View Parent Retrieve Output... Delete Refresh Close




Windows Task Instance Details Field Descriptions

The following table describes the fields, buttons, and tabs that display in Windows Task Instance Details.

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

General	This section contains general information about the task instance.
Instance Name	Name of this task instance.
Reference Id	System-supplied; the Controller increments this number each time the task is run.
Task	Name of the task that was run to create this task instance. Click the icon to display Task Details for the task.
Invoked by	System-supplied; how the task instance was launched. Options: <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 identify the user, check the Execution User column for that task instance on the Task Instances screen or, on most task instance screens, the Execution User field.
Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
Execution User	System-supplied; if the task was launched manually, the ID of the user who launched it.
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.
Status	This section contains information about the current status of the task instance.
Status	System-supplied; see Task Instance Statuses .
Exit Code	System-supplied; the exit code captured by the Agent when executing the task (for example, a command or script).
Status Description	System-supplied; additional information, if any, about the status of the task instance.
Queued Time	System-supplied; Date and time the task was queued for processing.
Trigger Time	System-supplied; Date and time the task instance was triggered.
Launch Time	System-supplied; Date and time the task instance was launched.

Start Time	System-supplied; Date and time the task instance started.
End Time	System-supplied; Date and time the task instance completed.
Duration	System-supplied; amount of time the task instance took to run.
CPU Time	System-supplied; amount of CPU time the task took to run.
Process ID	system-supplied; the ID of the process that was launched.
Windows Details	This section contains assorted detailed information about the task instance.
Agent	Name of the Agent resource that identifies the machine where the operation will run. If you do not specify an Agent, you must specify an Agent Cluster .
Agent Variable	<p>If enabled, 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. Use the format: <code>\${variable name}</code>. The variable must be a supported type as described in Variables and Functions.</p> <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note If you select Agent Variable when updating multiple tasks, the Agent field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Agent Unresolved - into which you enter a variable, variable and text, or Agent name. The Agent field is not used.</p> </div>
Agent Cluster	Group of Agents, one of which the Controller will choose to run this task. You can specify an agent cluster in addition to or in place of a specific Agent. If you specify an Agent and an agent cluster, the Controller first tries to run the task on the specific agent. If the Agent is not available, the Controller reverts to the agent cluster. See Agent Clusters for more information.
Agent Cluster Variable	<p>If enabled, 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. Use the format: <code>\${variable name}</code>. The variable must be a supported type as described in Variables and Functions.</p> <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note If you select Agent Cluster Variable when updating multiple tasks, the Agent Cluster field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Agent Cluster Unresolved - into which you enter a variable, variable and text, or Agent Cluster name. The Agent Cluster field is not used.</p> </div>
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.

<p>Credentials Variable</p>	<p>If enabled, the Credentials field 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> <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note If you select Credentials Variable when updating multiple tasks, the Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Credentials Unresolved - into which you enter a variable, variable and text, or Credentials name. The Credentials field is not used.</p> </div>
<p>Command or Script</p>	<p>Specifies whether a single command or a script is being executed.</p> <p>Options:</p> <ul style="list-style-type: none"> • Command (default) • Script <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note For both command-based tasks that call a .vbs/.js file directly, and script-based tasks that also rely on the systems association with file extension, GUI-based wscript.exe is associated with the vbs and js file extensions. Without explicitly calling one or the other, the Controller would use wscript.exe.</p> <p>The Agent system may need to be adjusted to properly use the Windows Scripting Host from the scheduler/agent environment.</p> <p>The following command can be used to set the default script host to cscrip.exe: <code>C:\tmp>cscript //h:cscript //s</code></p> </div>
<p>Command</p>	<p>If Command or Script = Command; Required. Command being executed on the remote machine. Variables supported.</p>
<p>Script</p>	<p>If Command or Script = Script; Required. Name of the script in the Controller database that will be executed by this task.</p>
<p>Parameters</p>	<p>Any arguments needed by the program to execute properly. Variables supported.</p>
<p>Runtime Directory</p>	<p>Directory from which the application should be executed. Variables supported.</p>
<p>Interact with Desktop</p>	<p>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).</p> <div 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, Server 2008, Windows 8, and Server 2012. 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>

Environment Variables	<p>Allows you to enter environment variables needed by the program to run.</p> <p>To add a variable, click the + icon and enter a Name and Value. To delete a variable, select in the list of variables and click the - icon.</p> <p>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.</p>
Exit Code Processing	<p>Specifies how the Controller should determine whether the executed command failed or completed successfully.</p> <p>Options:</p> <ul style="list-style-type: none"> • Success Exitcode Range - Command is considered completed successfully if its exit code falls within the range specified in the Exit Codes field. • Failure Exitcode Range - Command is considered failed if its exit code falls within the range specified in the Exit Codes field. • Success Output Contains - Command is considered completed successfully if its output contains the text specified in the Scan Output For field. • Failure Output Contains - Command is considered failed if its output contains the text specified in the Scan Output For field. • Step Conditions (z/OS only) - 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	<p>Required if Exit Code Processing = Success Output Contains or Failure Output Contains; type of output.</p> <p>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 Exitcode Range; range of exit codes. Format: Numeric. Use commas to list a series of exit codes; 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; text for which the Controller should scan the output file. The Controller will process this field as a regular expression.</p>
Output File (for Exit Code Processing)	<p>Required if Output Type = File; path and file name of the output file that should be scanned for the text in the Scan Output For field.</p>
Automatic Output Retrieval	<p>Specifies whether you want the Controller to automatically retrieve any output from the job and attach it to the task instance record.</p> <p>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. • Standard Output/Error - Attach all standard output and standard error output.
Start Line	<p>If Automatic Output Retrieval = Standard Output, Standard Error, File, or Standard Output/Error; Instructs the Controller to retrieve data beginning at the line indicated.</p> <ul style="list-style-type: none"> • If a Start Line value is not specified, the default is 1. • If the Start Line value is -1, data will be retrieved starting at the end of the file.

Number of Lines	If Automatic Output Retrieval = Standard Output, Standard Error, File, or Standard Output/Error; 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 Controller system property.
Scan Text	If Automatic Output Retrieval = Standard Output, Standard Error, File, or Standard Output/Error; Regex pattern that the Controller will search for a match for in STDOUT/STDERR or a specified file. The Controller will include the Number of Lines above and below the first line matched. if the Regex pattern is not found, the following message is returned: OPSWISE WARNING - Scan text string not found.
Output File (for Automatic Output Retrieval)	Required if Automatic Output Retrieval = File; path and file name containing the output that you want automatically retrieved and attached to the task instance.
Retry Options	This section contains specifications for retrying the task.
Retry Exit Codes	Exit code range for which an auto-retry of tasks in FAILED status will occur. Exit code ranges must be in the same format as ranges specified in the Exit Codes field. Maximum Retries must be greater than 0. If this field is empty, any exit code potentially will cause a retry.
Maximum Retries	User-defined; maximum number of times that the Controller should retry this task after it has started and gone to a failed state.
Retry Indefinitely	User-defined; indicates whether the Controller should continue trying indefinitely to run this task. If you enable this field, it overrides any value placed in the Maximum Retries field.
Retry Interval (Seconds)	User-defined; number of seconds between each retry.
Current Retry Count	System-supplied; current number of times that the Controller has retried the task after it first went to failure status.
Time Options	This section contains time-related specifications for the task instance.
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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.
Started Late	System-supplied; this field is flagged if the task started later than the time specified in the Late Start fields.
Late Start Type	Required if Late Start is enabled. Options: <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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.

Late Start Duration	<p>If Late Start Type = 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.</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>
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>
Finished Late	<p>System-supplied; this field is flagged if the task finished later than the time or duration specified in the Late Finish fields.</p>
Late Finish Type	<p>Required if Late Finish is enabled.</p> <p>Options:</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 = Time; 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 = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.</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>
Finished Early	<p>System-supplied; this field is flagged if the task finished earlier than the time specified in the Early Finish fields.</p>
Early Finish Type	<p>Required if Early Finish is enabled.</p> <p>Options:</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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.</p>
Statistics	<p>This section contains time-related statistics for the task instance.</p>

User Estimated End Time	System-supplied; if the user entered information into the User Estimated Duration field in the task Details, the Controller uses this information to calculate an end time for the task instance, based on the date/time the task instance started.
Average Estimated End Time	System-supplied; average estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Shortest Estimated End Time	System-supplied; shortest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Longest Estimated End Time	System-supplied; longest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Buttons	This section identifies the buttons displayed above and below the Task Instance Details that let you perform various actions.
Update	Saves updates to the record.
Force Finish	See Force Finishing a Task .
Hold	Places the task instance on Hold (see Putting a Task on Hold).
Skip	For tasks loaded into the schedule that have not yet run; allows you to tell the Controller to skip this task. See Skipping a Task .
Re-run	See Re-running a Task .
View Parent	Displays the task instance Details for the parent Workflow of this task instance.
Retrieve Output	See Retrieving Output .
Delete	Deletes the current record.
Refresh	Refreshes any dynamic data displayed in the Details.
Close	For pop-up view only; closes the pop-up view of this task instance.
Tabs	This section identifies the tabs across the top of the Task Instance Details that provide access to additional information about the task instance.
Virtual Resources	Lists all Virtual Resources to which this task is assigned.
Exclusive Requests	Lists all records in the Exclusive Requests table (<code>ops_exclusive_order</code>) for this task instance.

Output	<p>Displays output generated from the process, if any, based on specifications provided by the user in the Automatic Output Retrieval fields in the task Details.</p> <p>If automatic output retrieval was not available or was not selected, output can be obtained by clicking the Retrieve Output button.</p>
Notes	<p>Lists all notes associated with this record.</p>

Running a Windows Task

You can run a Windows task:

- Manually, by clicking the [Launch Task](#) or [Launch Task with Variables](#) button in the Windows tasks list or Windows Task Details [Action menu](#).
- As part of a [Workflow](#).
- [Specify triggers](#) that run the task automatically based on times or events.

Issuing Commands via cscript.exe

The following information applies to Windows tasks (and [Indesca tasks](#)) that are issuing commands rather than using [scripts](#).

By default, Windows uses a GUI-based VBScript interpreter (`wscript.exe`) to issue commands on an Agent system.

However, we recommend you use the console version of the VBScript interpreter (`cscript.exe`).

(When using scripts, the task does not explicitly call `wscript.exe` or `cscript.exe`.)

To do so, specify `cscript.exe` before the server operation name in the **Command** field, as shown in the following example:

```
"cscript.exe C:\Work\script.vbs".
```

With `wscript.exe`, if your server operation tries to display an error message that requires a user-response (for example, **Click OK**), you will never see the dialog box. The operation therefore gets stuck in the [Running](#) state.

Monitoring Task Execution

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

zOS Task

Information on z/OS tasks is contained on the following pages:

- [Creating and Running a zOS Task](#)
- [Special Processing on zOS Tasks](#)
- [Creating Step Conditions](#)
- [Creating Step Actions](#)
- [Creating Restart Criteria](#)

Creating and Running a z/OS Task

- [Before You Begin](#)
- [Built-In Variables](#)
- [Creating a z/OS Task](#)
 - [z/OS Task Details](#)
 - [z/OS Task Details Field Descriptions](#)
- [Viewing a z/OS Task Instance](#)
 - [z/OS Task Instance Details](#)
 - [z/OS Task Instance Details Field Descriptions](#)
- [Running a z/OS Task](#)
- [Monitoring Task Execution](#)

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

Built-In Variables

The following [built-in variables](#) can be used in a z/OS task to pass data where appropriate:

- Task and Task Instance variables
- Agent variables

Creating a z/OS Task

Step 1 From the Automation Center navigation pane, select **Tasks > z/OS Tasks**. The z/OS Tasks list displays a list of all currently defined z/OS tasks.

Below the list, z/OS Task Details for a new z/OS task displays. (You also can click the **New** button to display z/OS Task Details for a new z/OS task.)

The screenshot shows the 'z/OS Tasks' interface. At the top, there is a list of tasks with columns for Task Name, Task Description, JCL Location, Updated By, and Updated. Below the list, the 'z/OS Task Details' form is displayed, showing fields for Task Name, Version, Task Description, Member of Business Services, Hold on Start, Virtual Resource Priority, and Hold Resources on Failure.

Task Name	Task Description	JCL Location	Updated By	Updated
stonebranch-zOSTask-01		C:	ops.admin	2014-06-13 13:59:55 -0400
stonebranch-zOSTask-02		C:	ops.admin	2014-06-13 14:00:08 -0400
stonebranch-zOSTask-03		C:	ops.admin	2014-06-13 14:00:15 -0400
stonebranch-zOSTask-04		C:	ops.admin	2014-06-13 14:00:21 -0400
stonebranch-zOSTask-05		C:	ops.admin	2014-06-13 14:00:29 -0400

The 'z/OS Task Details' form includes the following fields:

- Task Name**: [Text Input]
- Version**: [Text Input] (Default: 1)
- Task Description**: [Text Input]
- Member of Business Services**: [Dropdown Menu]
- Hold on Start**:
- Virtual Resource Priority**: [Dropdown Menu] (Default: 10)
- Hold Resources on Failure**:
- Agent**: [Dropdown Menu]
- Agent Variable**:

Step 2 Enter/select Details for a new z/OS task, using the [field descriptions](#) below as a guide.

- Required fields display in **boldface**.
- Default values for fields, if available, display automatically.

To display more of the Details fields on the screen, you can temporarily [hide the list](#).



Note

If you view [z/OS Task Details](#) for an existing z/OS task by clicking a task in the list, and then want to create a new z/OS task, you must click the **New** button that displays above and below the Details.

Step 3 Click the **Save** button. The task is added to the database, and all buttons and tabs in the Task Details are enabled.

z/OS Task Details

The following z/OS Task Details is for an existing Linux/Unix task.

Depending on the values that you enter / select for these fields, and whether or not the z/OS task has ever been launched, more (or less) fields may display. See the [field descriptions](#), below, for a description of all fields that may display in the z/OS Task Details.

The screenshot displays the 'z/OS Task Details' window for a task named 'stonebranch-zOStask-01'. The window has a title bar and a menu bar with options like Update, Launch Task, View Parents, Copy, Delete, Refresh, and Close. Below the menu bar are several tabs: z/OS Task, Step Conditions, Restart Criteria, Variables, Actions, Virtual Resources, Mutually Exclusive, Instances, Triggers, Notes, and Versions. The 'General' tab is selected and contains the following fields:

- Task Name:** stonebranch-zOStask-01
- Version:** 8
- Task Description:** IF/THEN/ELSE - \${city}
- Member of Business Services:** (dropdown menu)
- Hold on Start:**
- Virtual Resource Priority:** 10
- Hold Resources on Failure:**

The 'z/OS Details' tab is also visible and contains:

- Agent:** stonebranch-zOSagent-01
- Agent Variable:**
- Credentials:** stonebranch-credential-01
- Credentials Variable:**
- JCL Location:** TEST.CNTL(D237RC)
- Use JCL Override Library:**
- New Jobname:** (text field)
- Procedure Library:** (text field)
- New Jobclass:** (text field)
- Schedule Id:** (text field)
- New Msgclass:** (text field)
- Parameters:** A table with two columns: Name and Value.

Name	Value
*@CODE1	4
*@CODE2	8
- Exit Code Processing:** Success Exitcode Range
- Exit Codes:** 0-16
- Automatic Output Retrieval:** -- None --

The 'Retry Options' tab shows:

- Auto-Restart Option:** -- None --
- Maximum Retries:** 0
- Retry Interval (Seconds):** 60
- Retry Indefinitely:**

The 'Time Options' tab shows:

- Late Start:**
- Late Finish:**
- Early Finish:**
- User Estimated Duration:** (HH:MM:SS format)

The 'Statistics' tab shows:


- First Time Ran:** 2014-03-13 10:32:43 -0400
- Last Time Ran:** 2014-06-23 08:55:17 -0400
- Last Instance Duration:** 0 Seconds
- Number of Instances:** 23
- Lowest Instance Time:** 0 Seconds
- Average Instance Time:** 1 Seconds
- Highest Instance Time:** 3 Seconds



At the bottom of the window, there is a toolbar with buttons for Update, Launch Task, View Parents, Copy, Delete, Refresh, and Close.


z/OS Task Details Field Descriptions

The following table describes the fields, buttons, and tabs that display in the z/OS Task Details.


Field Name	Description
General	This section contains general information about the task.
Task Name	User-defined name of this task, containing a maximum 255 alphanumeric characters; variables supported. It is the responsibility of the user to develop a workable naming scheme for tasks.

Version	System-supplied; version number of the current record, which is incremented by the Controller every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
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.
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.
z/OS Details	This section contains assorted detailed information about the task.
Agent	Name of the Agent resource that identifies the machine where the operation will run.
Agent Variable	If enabled, 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. Use the format: \${variable name}. The variable must be a supported type as described in Variables and Functions . <div style="background-color: #ffffcc; padding: 10px; margin: 10px 0;"> <p> Note If you select Agent Variable when updating multiple tasks, the Agent field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Agent Unresolved - into which you enter a variable, variable and text, or Agent name. The Agent field is not used.</p> </div>
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.

<p>Credentials Variable</p>	<p>If enabled, the Credentials field 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> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Credentials Variable when updating multiple tasks, the Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Credentials Unresolved - into which you enter a variable, variable and text, or Credentials name. The Credentials field is not used.</p> </div>
<p>JCL Location</p>	<p>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; width: fit-content; margin: 10px auto;"> <pre>&PRODLIB (PAYJOB01)</pre> </div>
<p>Use JCL Override Library</p>	<p>Allows the task to work with a JCL override library on the target system. If this option is selected, the Agent will check the JCL override path (specified in JCL Override Location) before submitting the job from JCL Location.</p> <p>If a JCL member is found at the override path, the job is submitted from there. Otherwise, the Agent will look to the JCL Location path for submission.</p> <p>The task instance will display the actual path that was used for job submission in Submitted JCL Location.</p>
<p>JCL Override Location</p>	<p>If Use JCL Override Library is selected; Required. Specifies the file and member name potentially containing an override JCL script. The Agent will check this location for JCL before looking in the standard JCL Location. If JCL is found in this location, the job will be submitted from there. If JCL is not found in this location, the Agent will submit the job from the standard JCL Location.</p> <p>Missing override JCL is not considered an error condition.</p> <p>As with JCL Location, 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 <code>uags.conf</code> file with the JCL_library definitions.</p>
<p>Delete Override JCL</p>	<p>If Use JCL Override Library is selected; Optional. Allows the task to define criteria that will control the automated clean-up of the override JCL. If this option is not selected, the Controller and Agent will take no action to delete the override JCL from the target system.</p>
<p>Override Instance Count for Deletion</p>	<p>If Delete Override JCL is selected; Required. Specifies the number of successful override instances that must occur before the override JCL library member is deleted.</p> <p>When the deletion criteria has been satisfied, the Controller will instruct the Agent to delete the member specified in JCL Override Location.</p> <p>An override instance is considered successful only if the ending state in the controller is SUCCESS.</p> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note Manual resubmissions of a task instance do not increment the tracked number of successful override instances that go towards satisfying the deletion criteria.</p> <p>If a task is not submitted from the override location (that is, no override member was found), the task's "successful override instance count" is reset to 0.</p> </div>
<p>Last Override Deletion</p>	<p>If Delete Override JCL is selected; Read-only, system-supplied. Displays after the specified override JCL member is deleted. The date and time the last override JCL deletion occurred.</p>


Number of Override Instances	<p>If Delete Override JCL is selected; Read-only, system-supplied. Indicates the number of successful override instances that have occurred for this task. This number is checked against the deletion criteria to determine when the override JCL member should be deleted.</p> <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note This number is automatically reset to 0 by the system if a task instance does not submit from override JCL (that is, no override member was found).</p> </div>
New Jobname	<p>Job name 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> <p>This value should be validated before the job is launched to avoid JES start failures.</p> <p>The syntax of a job name is:</p> <ul style="list-style-type: none"> • 1-8 characters • Upper case • Name must start with an alphabetic or \$, #, @ character. • Remaining characters are alphanumeric or \$, #, @. • No spaces or tabs.
New Jobclass	<p>New Jobclass to replace the one in the JCL member. This allows you to override the value in your JCL from the Controller without having to modify the JCL.</p>
New Msgclass	<p>New MSGCLASS to replace the one in the JCL member. This allows you to override the value in your JCL from the Controller without having to modify the JCL.</p>
Procedure Library	<p>The PROCLIB field allows for defining a JES2 PROCLIB control statement in the job JCL. For example, a PROCLIB value of PROC01 will result in the following JES2 control statement generated in the job JCL:</p> <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <pre>/*JESPARM PROCLIB=PROC01</pre> </div> <p>The PROCLIB value must refer to a ddname defined in the JES2 procedure. Refer to IBM MVS JCL Reference for more information regarding the JES2 PROCLIB control statement.</p>
Schedule ID	<p>CA7 Schedule ID; for CA7 toleration only (see Special Processing on zOS Tasks#CA7/CA11 Toleration Click).</p>

Parameters	<p>Displays a list of parameters that will be inserted into the JCL. Each parameter consists of a Name and a Value. You can enter as many parameters as needed.</p> <p>To add a parameter, click the + icon; add a Name and Value, and click the Update button. To delete a parameter, click the parameter on the list, the - icon, and the Update button.</p> <p>Each parameter that you enter creates a separate JCL construct called the SET command. Each one appears as a new line inserted dynamically into the JCL submitted to the Controller for the current execution. The JCL is not permanently modified.</p> <p>For example, you might specify a parameter Name = RUNTYPE and Value = 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>The Parameter 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.
Exit Code Processing	<p>Specifies how the Controller should determine whether the executed command failed or completed successfully.</p> <p>Options:</p> <ul style="list-style-type: none"> • Success Exitcode Range - Command is considered completed successfully if its exit code falls within the range specified in the Exit Codes field. • Failure Exitcode Range - Command is considered failed if its exit code falls within the range specified in the Exit Codes field. • Success Output Contains - Command is considered completed successfully if its output contains the text specified in the Scan Output For field. • Failure Output Contains - Command is considered failed if its output contains the text specified in the Scan Output For field. • Step Conditions (z/OS only) - 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	<p>Required if Exit Code Processing = Success Output Contains or Failure Output Contains; type of output.</p> <p>Options:</p> <ul style="list-style-type: none"> • Standard Output (STDOUT) • Standard Error (STDERR) • File
Scan Output For	<p>Required if Exit Code Processing = Success Output Contains or Failure Output Contains; text for which the Controller should scan the output file. The Controller will process this field as a regular expression.</p>
Output File (for Exit Code Processing)	<p>Required if Output Type = File; path and file name of the output file that should be scanned for the text in the Scan Output For field.</p>
Exit Codes	<p>Required if Exit Code Processing = Success Exitcode Range or Failure Exitcode Range; range of exit codes. Format: Numeric. Use commas to list a series of exit codes; use hyphens to specify a range. Example: 1,5, 22-30.</p>

Automatic Output Retrieval	<p>Specifies whether you want the Controller to automatically retrieve output from the job and attach it to the task instance record.</p> <p>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.
Start Line	<p>If Automatic Output Retrieval = Joblog or File; Allows you to instruct the Controller 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>If Automatic Output Retrieval = Joblog or File; 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 Controller system property.</p>
Scan Text	<p>If Automatic Output Retrieval = Joblog or File; Instructs the Controller to scan the data for the text specified and retrieve only that. The Controller will process this field as a regular expression.</p>
Output File (for Automatic Output Retrieval)	<p>Required if Automatic Output Retrieval = File; path and file name containing the output that you want automatically retrieved and attached to the task instance.</p>
Retry Options	<p>This section contains specifications for retrying the task.</p>
Auto-Restart Option	<p>Allows the z/OS job to be resubmitted with controlled step selection. This option is processed when/if a task transitions to a failed state. It works in conjunction with the Maximum Retries, Retry Interval, and Retry Indefinitely options.</p> <div style="background-color: #ffffcc; padding: 10px; margin: 10px 0;"> <p> Note The Maximum Retries value must be greater than 0 for the Auto-Restart Option to be processed.</p> </div> <p>Options:</p> <ul style="list-style-type: none"> • None - No job steps will be automatically selected for restart. • Restart From First Job Step - All restartable job steps will be selected for restart. • Restart From Failed Job Step - All restartable job steps from the failed step to the last job step will be selected for restart. • Use Restart Criteria - The entries in the Restart Criteria tab will be evaluated. If a Restart Criteria entry matches the failure scenario, the step selection will be based on the option specified in the matching Restart Criteria entry. <p>If you select an option other than None for a task that ends in a failed state, audit records will be generated to record the step selection that took place for the restart. The audit records include all restart options, criteria matching, and directives that were used to select the set of job steps to be re-run. z/OS auto-restart audit records show up as audit type z/OS Auto-Restart from source Task Instance.</p>
Maximum Retries	<p>User-defined; maximum number of times that the Controller should retry this task after it has started and gone to a failed state.</p>
Retry Indefinitely	<p>User-defined; indicates whether the Controller should continue trying indefinitely to run this task. If you enable this field, it overrides any value placed in the Maximum Retries field.</p>
Retry Interval (Seconds)	<p>User-defined; number of seconds between each retry.</p>
Time Options	<p>This section contains time-related specifications for the task.</p>

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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.
Late Start Type	Required if Late Start is enabled. Options: <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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.
Late Start Duration	If Late Start Type = 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.
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: <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 = Time; Time after which the task finish time is considered late. Use HH:MM, 24-hour time.
Late Finish Duration	If Late Finish Type = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
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: <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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
User Estimated Duration	Estimated amount of time it should normally take to run this task. The Controller uses this information to calculate the User Estimated End Time on a task instance record.
Statistics	This section contains time-related statistics for task instances of the task.
First Time Ran	System-supplied; date and time this task first ran.
Last Time Ran	System-supplied; date and time the task last ran.
Last Instance Duration	System-supplied; amount of time the task took to run the last time it ran.
Lowest Instance Time	System-supplied; shortest amount of time this task has taken to run.
Average Instance Time	System-supplied; average amount of time this task takes to run.
Highest Instance Time	System-supplied; longest amount of time this task has taken to run.
Number of Instances	System-supplied; number of instances in the database for this task.
Buttons	This section identifies the buttons displayed above and below the Task Details that let you perform various actions.
Save	Saves a new task record in the Controller database.
Update	Saves updates to the record.
New	Displays empty (except for default values) Details for creating a new task.
Launch Task	Manually launches the task.
View Parents	Displays a list of any parent Workflow tasks for this task.
Copy	Creates a copy of this task, which you are prompted to rename.

<p>Delete</p>	<p>Deletes the current record.</p> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note You cannot delete a task if it is either:</p> <ul style="list-style-type: none"> Specified in an enabled Trigger. The only task specified in a disabled Trigger. </div>										
<p>Refresh</p>	<p>Refreshes any dynamic data displayed in the Details.</p>										
<p>Close</p>	<p>For pop-up view only; closes the pop-up view of this task.</p>										
<p>Tabs</p>	<p>This section identifies the tabs across the top of the Task Details that provide access to additional information about the task.</p>										
<p>Step Conditions</p>	<p>Lists all step conditions defined for this task.</p>										
<p>Restart Criteria</p>	<p>Displays a list of all restart criteria defined for this task.</p>										
<p>Variables</p>	<p>Lists all variables associated with this record.</p>										
<p>Actions</p>	<p>Allows you to specify actions that the Controller will take automatically based on events that occur during the execution of this task.</p> <p>Events are:</p> <ul style="list-style-type: none"> Task instance status Exit codes Late start Late finish Early finish <p>Actions are:</p> <table border="1" data-bbox="277 1192 1484 1556"> <tr> <td data-bbox="277 1192 418 1266">Abort Action</td> <td data-bbox="418 1192 1484 1266">Abort the task if certain events occur. For details, see Abort Actions.</td> </tr> <tr> <td data-bbox="277 1266 418 1339">Email Notification</td> <td data-bbox="418 1266 1484 1339">Send an email if certain events occur. For details, see Email Notification Actions.</td> </tr> <tr> <td data-bbox="277 1339 418 1413">Set Variable</td> <td data-bbox="418 1339 1484 1413">Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow.</td> </tr> <tr> <td data-bbox="277 1413 418 1486">SNMP Notification</td> <td data-bbox="418 1413 1484 1486">Send an email if certain events occur. For details, see SNMP Notification Actions.</td> </tr> <tr> <td data-bbox="277 1486 418 1556">System Operation</td> <td data-bbox="418 1486 1484 1556">Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions.</td> </tr> </table>	Abort Action	Abort the task if certain events occur. For details, see Abort Actions .	Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .	Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .	SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .	System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .
Abort Action	Abort the task if certain events occur. For details, see Abort Actions .										
Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .										
Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .										
SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .										
System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .										
<p>Virtual Resources</p>	<p>Lists all Virtual Resources to which this task is assigned.</p>										
<p>Mutually Exclusive</p>	<p>Lists all tasks that have been set to be mutually exclusive of this task.</p>										
<p>Instances</p>	<p>Lists all instances of the task.</p>										
<p>Triggers</p>	<p>Lists 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, the Controller 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 Triggers Overview.</p>										

Notes	Lists all notes associated with this record.
Versions	Stores copies of all previous versions of the current record. See Record Versioning .

Viewing a z/OS Task Instance

When a z/OS task is launched, the Controller creates a task instance record of that task.

A task instance contains detailed information about a single execution of that task.

You can access a task instance from:

- **Instances tab** on the [z/OS Task Details](#) for that task
- [Activity Monitor](#)
- [Task Instances list](#)

z/OS Task Instance Details

The following z/OS Task Instance Details contains information on the successful completion of a z/OS task.

z/OS Task Instance Details: stonebranch-zOSTask-01

Update Re-run View Parent Retrieve Output... Delete Refresh Close

z/OS Task Instance Step Conditions Restart Criteria Restartable Job Steps Confirm JCL Changes Virtual Resources Exclusive Requests Output Notes

General

Instance Name: stonebranch-zOSTask-01 Reference Id: 14
 Task: stonebranch-zOSTask-01 Invoked By: Manually Launched
 Task Description: IF/THENELSE - \${city}
 Member of Business Services: Execution User: stonebranch-user-01
 Virtual Resource Priority: 10 Hold Resources on Failure:

Status

Status: Success Exit Code: 16
 Status Description:
 Queued Time: 2015-02-06 14:45:37 -0500
 Trigger Time: Start Time: 2015-02-06 14:45:41 -0500
 Launch Time: 2015-02-06 14:45:37 -0500 End Time: 2014-06-20 13:09:49 -0400
 Duration: 1 Seconds CPU Time: 1
 Job ID: JOB01536 Job Name: D03237RC

z/OS Details

Agent: stonebranch-zOSagent-01
 Agent Variable:
 Credentials: stonebranch-credential-01
 Credentials Variable:
 JCL Location: TESTCNTL(D237RC)
 Use JCL Override Library:
 New Jobname: Procedure Library:
 New Jobclass: Schedule Id:
 New Msgclass: JCL Changes Confirmed:

Name	Value
*@CODE1	4
*@CODE2	8

Exit Code Processing: Success Exitcode Range
 Exit Codes: 0-16
 Automatic Output Retrieval: -- None --

Retry Options

Auto-Restart Option: -- None --
 Maximum Retries: 0
 Retry Interval (Seconds): 60
 Retry Indefinitely:
 Current Retry Count: 0

Statistics

User Estimated End Time: Average Estimated End Time: 2014-06-08 16:01:59 -0400
 Shortest Estimated End Time: 2014-06-08 16:01:59 -0400
 Longest Estimated End Time: 2014-06-08 16:02:00 -0400



Update Re-run View Parent Retrieve Output... Delete Refresh Close



z/OS Task Instance Details Field Descriptions

The following table describes the fields, buttons, and tabs that display in z/OS Task Instance Details.

Field Name	Description
General	This section contains general information about the task instance.
Instance Name	Name of this task instance.


Reference Id	System-supplied; the Controller increments this number each time the task is run.
Task	Name of the task that was run to create this task instance. Click the icon to display Task Details for the task.
Invoked by	System-supplied; how the task instance was launched. Options: <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 identify the user, check the Execution User column for that task instance on the Task Instances screen or, on most task instance screens, the Execution User field.
Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
Execution User	System-supplied; if the task was launched manually, the ID of the user who launched it.
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.
Status	This section contains information about the current status of the task instance.
Status	System-supplied; see Task Instance Statuses .
Exit Code	System-supplied; the exit code captured by the Agent when executing the task (for example, a command or script).
Status Description	System-supplied; additional information, if any, about the status of the task instance.
Queued Time	System-supplied; Date and time the task was queued for processing.
Trigger Time	System-supplied; Date and time the task instance was triggered.
Launch Time	System-supplied; Date and time the task instance was launched.
Start Time	System-supplied; Date and time the task instance started.

End Time	System-supplied; Date and time the task instance completed.
Duration	System-supplied; amount of time the task instance took to run.
CPU Time	System-supplied; amount of CPU time the task took to run.
Job ID	Job identifier of the job executed by the task instance.
Job Name	Name of the job executed by the task instance.
z/OS Details	This section contains assorted detailed information about the task instance.
Agent	Name of the Agent resource that identifies the machine where the operation will run.
Agent Variable	<p>If enabled, 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. Use the format: \${variable name}. The variable must be a supported type as described in Variables and Functions.</p> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Agent Variable when updating multiple tasks, the Agent field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Agent Unresolved - into which you enter a variable, variable and text, or Agent name. The Agent field is not used.</p> </div>
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	<p>If enabled, the Credentials field 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> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Credentials Variable when updating multiple tasks, the Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Credentials Unresolved - into which you enter a variable, variable and text, or Credentials name. The Credentials field is not used.</p> </div>
JCL Location	<p>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-top: 10px; width: fit-content;"> <pre>&PRODLIB (PAYJOB01)</pre> </div>

Use JCL Override Library	<p>Allows the task to work with a JCL override library on the target system. If this option is selected, the Agent will check the JCL override path (specified in JCL Override Location) before submitting the job from JCL Location.</p> <p>If a JCL member is found at the override path, the job is submitted from there. Otherwise, the Agent will look to the JCL Location path for submission.</p> <p>The task instance will display the actual path that was used for job submission in Submitted JCL Location.</p>
JCL Override Location	<p>If Use JCL Override Library is selected; Required. Specifies the file and member name potentially containing an override JCL script.</p> <p>The Agent will check this location for JCL before looking in the standard JCL Location. If JCL is found in this location, the job will be submitted from there. If JCL is not found in this location, the Agent will submit the job from the standard JCL Location.</p> <p>Missing override JCL is not considered an error condition.</p> <p>As with JCL Location, 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 <code>uags.conf</code> file with the JCL_library definitions.</p>
Delete Override JCL	<p>If Use JCL Override Library is selected; Optional. Allows the task to define criteria that will control the automated clean-up of the override JCL. If this option is not selected, the Controller and Agent will take no action to delete the override JCL from the target system.</p>
Submitted JCL Location	<p>System-supplied; actual path that was used for job submission.</p>
Override Instance Count for Deletion	<p>If Delete Override JCL is selected; Required. Specifies the number of successful override instances that must occur before the override JCL library member is deleted.</p> <p>When the deletion criteria has been satisfied, the Controller will instruct the Agent to delete the member specified in JCL Override Location.</p> <p>An override instance is considered successful only if the ending state in the controller is SUCCESS.</p> <div data-bbox="318 982 1446 1184" style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note Manual resubmissions of a task instance do not increment the tracked number of successful override instances that go towards satisfying the deletion criteria.</p> <p>If a task is not submitted from the override location (that is, no override member was found), the task's "successful override instance count" is reset to 0.</p> </div>
Last Override Deletion	<p>If Delete Override JCL is selected; Read-only, system-supplied. Displays after the specified override JCL member is deleted.</p> <p>The date and time the last override JCL deletion occurred.</p>
Number of Override Instances	<p>If Delete Override JCL is selected; Read-only; system-supplied. Indicates the number of successful override instances that have occurred for this task. This number is checked against the deletion criteria to determine when the override JCL member should be deleted.</p> <div data-bbox="318 1440 1446 1570" style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note This number is automatically reset to 0 by the system if a task instance does not submit from override JCL (that is, no override member was found).</p> </div>
New Jobname	<p>Job name 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> <p>This value should be validated before the job is launched to avoid JES start failures.</p> <p>The syntax of a job name is:</p> <ul style="list-style-type: none"> • 1-8 characters • Upper case • Name must start with an alphabetic or \$, #, @ character. • Remaining characters are alphanumeric or \$, #, @. • No spaces or tabs.

New Jobclass	New Jobclass to replace the one in the JCL member. This allows you to override the value in your JCL from the Controller without having to modify the JCL.
New Msgclass	New MSGCLASS to replace the one in the JCL member. This allows you to override the value in your JCL from the Controller without having to modify the JCL.
Procedure Library	<p>The PROCLIB field allows for defining a JES2 PROCLIB control statement in the job JCL. For example, a PROCLIB value of PROC01 will result in the following JES2 control statement generated in the job JCL:</p> <pre data-bbox="321 472 1443 558" style="border: 1px solid #ccc; padding: 5px;">/*JESPARM PROCLIB=PROC001</pre> <p>The PROCLIB value must refer to a ddname defined in the JES2 procedure. Refer to IBM MVS JCL Reference for more information regarding the JES2 PROCLIB control statement.</p>
Schedule ID	CA7 Schedule ID; for CA7 toleration only (see Special Processing on zOS Tasks#CA7/CA11 Toleration Click).
JCL Changes Confirmed	If Status = Confirmation Required ; indicates that JCL changes have been confirmed. You cannot rerun a job if this field is not selected.
Parameters	<p>Displays a list of parameters that will be inserted into the JCL. Each parameter consists of a Name and a Value. You can enter as many parameters as needed.</p> <p>To add a parameter, click the + icon; add a Name and Value, and click the Update button. To delete a parameter, click the parameter on the list, the - icon, and the Update button.</p> <p>Each parameter that you enter creates a separate JCL construct called the SET command. Each one appears as a new line inserted dynamically into the JCL submitted to the Controller for the current execution. The JCL is not permanently modified.</p> <p>For example, you might specify a parameter Name = RUNTYPE and Value = PROD. This results in the following JCL SET statement being inserted in the job after the job card:</p> <pre data-bbox="321 1182 1443 1268" style="border: 1px solid #ccc; padding: 5px;">// SET RUNTYPE=PROD</pre> <p>The Parameter fields also support two additional special functions:</p> <ul data-bbox="332 1339 1443 1440" 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.
Exit Code Processing	<p>Specifies how the Controller should determine whether the executed command failed or completed successfully.</p> <p>Options:</p> <ul data-bbox="332 1604 1492 1850" style="list-style-type: none"> • Success Exitcode Range - Command is considered completed successfully if its exit code falls within the range specified in the Exit Codes field. • Failure Exitcode Range - Command is considered failed if its exit code falls within the range specified in the Exit Codes field. • Success Output Contains - Command is considered completed successfully if its output contains the text specified in the Scan Output For field. • Failure Output Contains - Command is considered failed if its output contains the text specified in the Scan Output For field. • Step Conditions (z/OS only) - 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	<p>Required if Exit Code Processing = Success Output Contains or Failure Output Contains; type of output.</p> <p>Options:</p> <ul style="list-style-type: none"> • Standard Output (STDOUT) • Standard Error (STDERR) • File
Scan Output For	<p>Required if Exit Code Processing = Success Output Contains or Failure Output Contains; text for which the Controller should scan the output file. The Controller will process this field as a regular expression.</p>
Output File (for Exit Code Processing)	<p>Required if Output Type = File; path and file name of the output file that should be scanned for the text in the Scan Output For field.</p>
Exit Codes	<p>Required if Exit Code Processing = Success Exitcode Range or Failure Exitcode Range; range of exit codes. Format: Numeric. Use commas to list a series of exit codes; use hyphens to specify a range. Example: 1,5, 22-30.</p>
Automatic Output Retrieval	<p>Specifies whether you want the Controller to automatically retrieve any output from the job and attach it to the task instance record.</p> <p>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. • Standard Output/Error - Attach all standard output and standard error output.
Start Line	<p>If Automatic Output Retrieval = Joblog or File; Allows you to instruct the Controller 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>If Automatic Output Retrieval = Joblog or File; 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 Controller system property.</p>
Scan Text	<p>If Automatic Output Retrieval = Joblog or File; Instructs the Controller to scan the data for the text specified and retrieve only that. The Controller will process this field as a regular expression.</p>
Output File (for Automatic Output Retrieval)	<p>Required if Automatic Output Retrieval = File; path and file name containing the output that you want automatically retrieved and attached to the task instance.</p>
Retry Options	<p>This section contains specifications for retrying the task.</p>

Auto-Restart Option	<p>Allows the z/OS job to be resubmitted with controlled step selection. This option is processed when/if a task transitions to a failed state. It works in conjunction with the Maximum Retries, Retry Interval, and Retry Indefinitely options.</p> <div style="background-color: #ffffcc; padding: 10px; margin: 10px 0;"> <p> Note The Maximum Retries value must be greater than 0 for the Auto-Restart Option to be processed.</p> </div> <p>Options:</p> <ul style="list-style-type: none"> • None - No job steps will be automatically selected for restart. • Restart From First Job Step - All restartable job steps will be selected for restart. • Restart From Failed Job Step - All restartable job steps from the failed step to the last job step will be selected for restart. • Use Restart Criteria - The entries in the Restart Criteria tab will be evaluated. If a Restart Criteria entry matches the failure scenario, the step selection will be based on the option specified in the matching Restart Criteria entry. <p>If you select an option other than None for a task that ends in a failed state, audit records will be generated to record the step selection that took place for the restart. The audit records include all restart options, criteria matching, and directives that were used to select the set of job steps to be re-run. z/OS auto-restart audit records show up as audit type z/OS Auto-Restart from source Task Instance.</p>
Maximum Retries	User-defined; maximum number of times that the Controller should retry this task after it has started and gone to a failed state.
Retry Indefinitely	User-defined; indicates whether the Controller should continue trying indefinitely to run this task. If you enable this field, it overrides any value placed in the Maximum Retries field.
Retry Interval (Seconds)	User-defined; number of seconds between each retry.
Current Retry Count	System-supplied; current number of times that the Controller has retried the task after it first went to failure status.
Time Options	This section contains time-related specifications for the task instance.
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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.
Started Late	System-supplied; this field is flagged if the task started later than the time specified in the Late Start fields.
Late Start Type	<p>Required if Late Start is enabled.</p> <p>Options:</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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.

Late Start Duration	<p>If Late Start Type = 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.</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>
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>
Finished Late	<p>System-supplied; this field is flagged if the task finished later than the time or duration specified in the Late Finish fields.</p>
Late Finish Type	<p>Required if Late Finish is enabled.</p> <p>Options:</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 = Time; 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 = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.</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>
Finished Early	<p>System-supplied; this field is flagged if the task finished earlier than the time specified in the Early Finish fields.</p>
Early Finish Type	<p>Required if Early Finish is enabled.</p> <p>Options:</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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.</p>
Statistics	<p>This section contains time-related statistics for the task instance.</p>

User Estimated End Time	System-supplied; if the user entered information into the User Estimated Duration field in the task Details, the Controller 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	System-supplied; shortest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Average Estimated End Time	System-supplied; average estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Longest Estimated End Time	System-supplied; longest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Buttons	This section identifies the buttons displayed above and below the Task Instance Details that let you perform various actions.
Update	Saves updates to the record.
Force Finish	See Force Finishing a Task .
Hold	Places the task instance on Hold (see Putting a Task on Hold).
Skip	For tasks loaded into the schedule that have not yet run; allows you to tell the Controller to skip this task. See Skipping a Task .
Re-run	See Re-running a Task .
View Parent	Displays the task instance Details for the parent Workflow of this task instance.
Retrieve Output	See Retrieving Output .
Delete	Deletes the current record.
Refresh	Refreshes any dynamic data displayed in the Details.
Close	For pop-up view only; closes the pop-up view of this task instance.
Tabs	This section identifies the tabs across the top of the z/OS Task Instance Details that provide access to additional information about the task instance.
Step Conditions	Lists all step conditions defined for this task.
Restart Criteria	Displays a list of all restart criteria defined for this task.
Restartable Job Steps	(See Re-running a z/OS Task .)
Confirm JCL Changes	(See Re-running a z/OS Task .)

Virtual Resources	Lists all Virtual Resources to which this task is assigned.
Exclusive Requests	Lists all records in the Exclusive Requests table (<code>ops_exclusive_order</code>) for this task instance.
Output	<p>Displays output generated from the process, if any, based on specifications provided by the user in the Automatic Output Retrieval fields in the task Details.</p> <p>If automatic output retrieval was not available or was not selected, output can be obtained by clicking the Retrieve Output button.</p>
Notes	Lists all notes associated with this record.

Running a z/OS Task

You can run a z/OS task:

- Manually, by clicking the [Launch Task](#) or [Launch Task with Variables](#) button in the z/OS tasks list or z/OS Task Details [Action menu](#).
- As part of a [Workflow](#).
- [Specify triggers](#) that run the task automatically based on times or events.

Monitoring Task Execution

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

Special Processing on zOS Tasks

- Overview
- Using Variables in JCL and In-Stream Data Sets
 - JCL Symbolic Parameters
 - Opwise Controller Parameters
- Skipping Steps during Initial Run
- Overriding Key JCL Parameters from Opwise Controller
- Disabling Automatic Data Set Deletion
- Re-running a z/OS Task
 - Confirming JCL Changes
 - Re-run a z/OS Task from a Specific Step
 - Restartable Job Steps List Column Descriptions
 - Confirm JCL Changes Tab Column Descriptions
- Interactively Ignoring a Stepcode to Force a Task to Complete
- CA7/CA11 Toleration
 - Non-Restartable Steps
 - SCHID - Overriding the CA7 Schedule ID
- Viewing Re-run Reports
- Viewing Audit Trails on a Restart

Overview

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 Controller](#)

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](#)

The Controller supports compatibility with other schedulers:

- [CA7/CA11 Toleration](#)

The Controller provides the following reports that track error processing:

- [Viewing Re-run 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 in z/OS task Details:

- JCL Symbolic Parameters
- Opwise Controller Parameters

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

JCL Symbolic Parameters

Use the z/OS Task Details [Parameters](#) field 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 the Controller adding a JCL SET statement to the JCL before the first step EXEC statement.

As an example, a z/OS Task 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.

Opwise Controller Parameters

Use the z/OS Task Details [Parameters](#) field 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. the Controller will substitute the parameter values in the JCL statements and in the in-stream data before the JCL is submitted to JES.

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

The following steps add an Opwise Controller parameter with the name *@DATE1 and a value of 20110601:

Step 1	Open the z/OS task.
Step 2	In the Parameters field, click the + icon.
Step 3	Add a parameter in the following format: <ul style="list-style-type: none"> Name is the name of a variable preceded with @. For example: *@DATE1. Value is the value you want to set to the variable. For example: 20110601.
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: <pre>//INPUT DD * @DATE1 /*</pre> <pre>//AIF IF @DATE1 > 20110101 THEN</pre> <p>If an Opwise Controller 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:</p> <pre>//INPUT DD * DATE@DATE1.ACT9898 /*</pre>

Skipping Steps during Initial Run

In a z/OS task, you can specify that one or more steps from the JCL should be skipped when the Controller 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 z/OS task.
Step 2	In the Parameters field, click the + icon.

Step 3	<p>Add a parameter in the following format:</p> <ul style="list-style-type: none"> • Name = SKIPSTAA (SKIPST is a required string. AA is any combination of alphanumerics used to make this SKIPST command unique. (You can add as many SKIPST commands as needed.) • Value = STEPNAME (JCL step name)
Step 4	<p>Repeat Steps 2 and 3 for each step you want to skip. Change the AA portion of the SKIPST for each parameter you add. Each Name must be unique.</p> <p>For example, you could enter parameters:</p> <ul style="list-style-type: none"> • SKIPST01, STEP03 • SKIPST02, STEP05

Overriding Key JCL Parameters from Opwise Controller

When you launch a z/OS task from the Controller, you can specify a different Jobname, Jobclass, Msgclass, Schedule ID or add a JOBPARM card. This enables you to run your JCL jobs from the Controller without having to go in and modify your JCL. You can do so by entering new value into the appropriate field in the [z/OS Task Details](#).

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 z/OS task.
Step 2	In the Parameters field, click the + icon.
Step 3	<p>Add a parameter in the following format:</p> <ul style="list-style-type: none"> • Name = OPSDSEDEL • Value = NO

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 [Re-run Report](#) will indicate if the feature has been disabled.

Re-running a z/OS Task

When you re-run a z/OS task, you must select job steps in the task to include in the re-run; you cannot simply click a **Re-run** button or **Re-run** in an [Action menu](#), as with other tasks.

The **Restartable Job Steps** tab in the z/OS Task Instance Details provides a list of all job steps in the task and indicates from which steps you can start the re-run.

When you re-run a z/OS task, the Controller automatically performs the following:

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



Note

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

Confirming JCL Changes

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

Re-run a z/OS Task from a Specific Step

Step 1 On the [Activity Monitor](#), click the Details icon next to the **Instance Name** of the task you want to re-run. The z/OS Task Instance Details displays.

Step 2 Click the **Restartable Job Steps** tab to display a list of all job steps in the task.

Step Number	Restartable	Selected for Re-run	Depends On	Step	Procedure	Program	Step Code	Failed	Updated By	Updated
1	Yes	No	N/A	OPSSTP00		UAGRERUN	0000	No	ops.system	2014-06-08 16:01:59 -0400
2	Yes	No	N/A	J10		WMSSETRC	0008	No	ops.system	2014-06-08 16:01:59 -0400
3	Yes	No	N/A	J20		WMSSETRC	0016	No	ops.system	2014-06-08 16:02:00 -0400
4	Yes	No	N/A	EFBR		EFBR14	0000	No	ops.system	2014-06-08 16:02:00 -0400

If applicable, the **Failed** column indicates at which step(s) the task failed.

Step 3 Assuming any failures were caused by error(s) in the JCL, examine the JCL and make your corrections. The Controller uses a background process to determine whether changes have been made to the JCL. Any changes trigger a confirmation process.

Step 4 Check the **Restartable** column to see from which steps you can start the re-run (**Yes** indicates that the task can be re-run starting from that job step.)

Click the Details icon next to the **Step Number** of any step to see detail information about that step.

Restartable Job Step Details

Update Select for Re-run Refresh Close

Restartable Job Step

Details

Task Instance: stonebranch-zOstask-01 Attempt: 1

Step Number: 3

Restartable: Selected for Re-run:

Step: J20

Procedure:

Program: WMSSETRC

Step Code: 0016

Failed:

Depends On: N/A

IO Reads: 0 CPU Time: 0

IO Writes: 0 Memory Used: 0

IO Other: 13 Memory Peak: 252

IO Total: 13

Update Select for Re-run Refresh Close

Step 5 From the Restartable Job Steps list, select the steps that you want included in the re-run:

1. Click the first step that you want included in the re-run.
2. Press and hold the <Shift> key.
3. Click the last step that you want included in the re-run.
4. Right-click any step in the selected group to display an [Action menu](#).
5. Click **Select for Re-run**.

(You also can right-click a step and select **Select to End for Re-run** to include that selected step and all following steps in the re-run.)

If you want to de-select a job step in the group for inclusion in the re-run, right-click the step and, in the [Action menu](#), click **Deselect for Re-run**.

Step Number	Restartable	Selected for Re-run	Depends On	Step	Procedure	Program	Step Code	Failed	Updated By	Updated
1	Yes	No	N/A	OPSSTP00		UAGRERUN	0000	No	ops.admin	2014-06-25 17:20:45 -0400
2	Yes	No	N/A	J10		WMSSETRC	0008	No	ops.admin	2014-06-25 17:21:09 -0400
3	Yes	Yes	N/A	J20		WMSSETRC	0016	No	ops.admin	2014-06-25 17:22:20 -0400
4	Yes	Yes	N/A	IEFBR		IEFBR14	0000	No	ops.admin	2014-06-25 17:22:20 -0400

Step 6 Click the **z/OS Task Instance** tab to redisplay the z/OS Task Instance Details.

Step 7 Click the **Re-run** button to re-run the task for the selected job steps.

Step 8 If you have made changes to the JCL, the task instance goes into a status of **Confirmation Required** on the Activity Monitor.

To confirm the changes:

1. From the Activity Monitor, [open](#) the task instance.
2. Click the **Confirm JCL Changes** tab to display a list of all changes that were made to the JCL:

Job Id	Step Number	Error Message	Old Data	New Data	Confirmed	Updated By	Updated
JOB08060	2	DSN does not match DD=SYSUT1	DSN=QA.TEST.OPSWISE.CNTX DD=SYSUT1	DSN=QA.TEST.OPSWISE.CNTL DD=SYSUT1	No	ops.system	2014-08-08 12:28:36
JOB08060	3	DSN does not match DD=SYSUT1	DSN=QA.TEST.OPSWISE.JCLA DD=SYSUT1	DSN=QA.TEST.OPSWISE.CNTL DD=SYSUT1	No	ops.system	2014-08-08 12:28:36

3. Click the **Confirm** button at the top of the list.
4. Click the **z/OS Task Instance** tab. The **JCL Changes Confirmed** field and then click the **Re-run** button.

Step 9 After the re-run is complete, the **Failed** column in the **Restartable Job Steps** list should show **No** for each step.



Note

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

Restartable Job Steps List Column Descriptions

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

Column Name	Description
Step Number	Number assigned to this step by the Controller.
Restartable	If the JCL job fails, the Controller determines the latest step you can restart from. Yes indicates that you can restart from this step.
Selected for Re-run	Indicates (Yes or No) whether or not this step has been selected for Re-run in the Restartable Job Steps Details.
Depends On	Indicates which other step(s) 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 of the JCL step (extracted from the JCL).
Procedure	Procedure 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	Indicates (Yes or No) whether or not this step failed.
Updated	Name of the user that last updated this step.
Updated	Date and time that this step was last updated.

Confirm JCL Changes Tab Column Descriptions

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

Column Name	Description
Jobid	Number assigned to this step by the Controller.
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, but you still want to finish the job, you can change the exit code to force finish the task.

Step 1	From the Activity Monitor, open the task instance.
---------------	--

Step 2 From the **Restartable Steps** tab, click on the step whose exit code you want to change. The Restartable Job Step Details for this job step displays:

Step 3 Enter the new code in the **Step Code** field and click **Update**.

Step 4 Click the **z/OS Task Instance** tab and then click the **Re-run** button.

CA7/CA11 Toleration

Non-Restartable Steps

The Controller 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 the Controller without modifying the JCL. When the Controller encounters one of the step names listed below in your JCL, the Controller will skip the step during a restart:

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

In the following example, step 2 has one of the above DD Names and therefore is a non-restartable step.

z/OS Task Instance	Step Conditions	Restart Criteria	Restartable Job Steps	Confirm JCL Changes	Virtual Resources	Exclusive Requests	Output	Notes		
3 Restartable Job Steps										
Step Number	Restartable	Selected for Re-run	Depends On	Step	Procedure	Program	Step Code	Failed	Updated By	Updated
1	Yes	No	N/A	OPSSTP00	UAGRERUN	0000		No	ops.system	2015-06-11 18:59:57 -0400
2	No	No	Step# 2	STEP01	IEFBR14	0000		No	ops.system	2015-06-11 18:59:57 -0400
3	Yes	No	N/A	STEP10	IDCAMS	0000		No	ops.system	2015-06-11 18:59:57 -0400

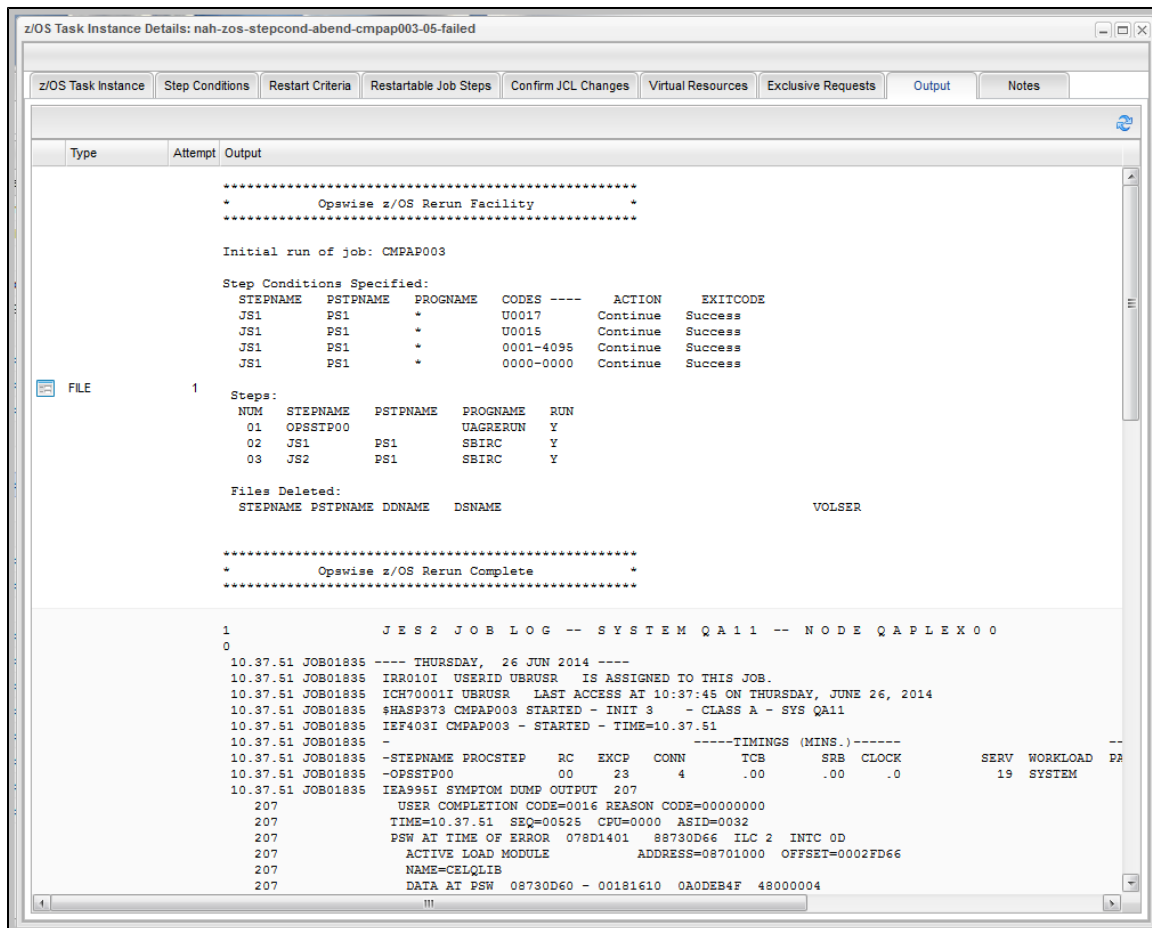
SCHID - Overriding the CA7 Schedule ID

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 in the z/OS Task Details. The 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 Re-run Reports

The Controller keeps a detailed record of task re-runs. 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

The Controller maintains [detailed audit records](#) on all system activity. The following audit record is for a re-run on a z/OS task.

Close

Audit
Child Audits

Details

Audit Type: <input type="text" value="Command"/>	Table Name: <input type="text" value="ops_exec_zos"/>
Audit Date: <input type="text" value="2014-06-10 21:33:14 -0400"/>	Table Key: <input type="text" value="bd4ffaff67764337af85c59efa0d5349"/>
Source: <input type="text" value="User Interface"/>	Parent Audit: <input type="text"/>
Created: <input type="text" value="2014-06-10 21:33:14 -0400"/>	Created By: <input type="text" value="ellen.ulrich"/>

Status: Command Success: Command Re-run executed successfully against task instance "zos-task-launch-rerun-with-gdg".

Description: Executing Command: RE-RUN on zos-task-launch-rerun-with-gdg (1)

Before:

```

ExecZosBean [ {abend_step=null} {acctinfo=null} {acquired_agent=} {acquired_agent_cluster=null}
{agent=1397156709115.192872b15b6d42c1a788cdd4c30d2c60} {agent_cluster=null} {agent_cluster_var_check=false}
{agent_id=1397156709115.192872b15b6d42c1a788cdd4c30d2c60} {agent_var=null} {agent_var_check=false} {all_dependencies_cleared=false}
{attempt_count=2} {auto_restart_option=- None ->} {avg_estimated_end=2014-06-10 21:31:46 -0400}
{calendar=77 171434c0a801c9015d5b2b5d17ddee} {cpu_time=0} {credentials=null} {credentials_var_check=false}
{duration=null} {duration_seconds=null} {early_finish=false} {ef_duration=null} {ef_enabled=false} {ef_time=00:00} {ef_type=TIME}
{end_time=2014-06-10 21:31:48 -0400} {error_desc=null} {exclusive_state=Initial} {execution_user=ellen.ulrich} {exit_code=JCLERR}
{exit_code_output=null} {exit_code_processing=Success Exitcode Range} {exit_code_text=null} {exit_codes=0} {forced=false}
{high_estimated_end=null} {hold_resources=false} {invoked_by=Manually Launched} {io_other=0} {io_reads=0} {io_writes=0} {jcd=null}
{jcd_location=QA.TEST.OPSWISE.CNTL(GDGTST4)} {job_id=JOB01642} {job_name=TESTGDG4} {jobclass_new=null} {jobname_new=null}
{late_finish=false} {late_start=false} {launch_time=2014-06-10 21:24:50 -0400} {if_duration=null} {if_enabled=false} {if_time=00:00} {if_type=TIME}
{low_estimated_end=null} {ls_duration=null} {ls_enabled=false} {ls_time=00:00} {ls_type=TIME} {memory_peak=0} {memory_used=0}
{msgclass_new=null} {name=zos-task-launch-rerun-with-gdg} {opswise_groups=null} {output_return_file=null} {output_return_nline=100}
{output_return_sline=1} {output_return_text=null} {output_return_type=JOBLOG} {output_type=STDOUT} {parameters=null} {parameters_list=null}
{priority=MEDIUM} {proclib_name=null} {queued_time=2014-06-10 21:31:46 -0400} {res_priority=10} {res_state=Initial} {resources_consumed=true}
{restart_available=false} {restart_command=false} {restart_files=null} {restart_force=false} {restart_jcd=null} {restart_step_name=null}
{restart_step_num=null} {restart_steps=null} {retry_counter=0} {retry_indefinitely=false} {retry_interval=60} {retry_maximum=0} {run_called=true}
{run_criteria_ft=false} {run_criteria_tt=false} {schedule_id=null} {security_name=zos-task-launch-rerun-with-gdg} {start_held=false}
{start_held_reason=null} {start_time=2014-06-10 21:31:46 -0400} {state_changed_time=2014-06-10 21:31:48 -0400} {status_code=FAILED}
{status_description=IEF452I TESTGDG4 - JOB NOT RUN - JCL ERROR } {summary=manual Launch task Re-run step02.SYSUT1 should
have the same gdg as the previous run.} {sys_class_name=ops_exec_zos} {sys_created_by=ellen.ulrich} {sys_created_on=2014-06-10 21:24:50
-0400} {sys_id=bd4ffaff67764337af85c59efa0d5349} {sys_updated_by=ops.system} {sys_updated_on=2014-06-10 21:31:48 -0400}
{task_id=3e91d3d91c9e4030bbeb49e764249ce5} {task_ref_count=1} {trigger_id=null} {type=z/OS} {user_estimated_end=null} {vertex_id=null}
{wait_for_exclusive=false} {wait_for_resources=false} {wf_progress=null} {workflow_definition_id=null} {workflow_id=null} {workflow_start_time=null}
{zosid=00000075} ]
                
```

After:

Difference:

Additional Information:

Close

Creating Step Conditions

- [Overview](#)
 - [Runtime Monitoring](#)
- [Creating a Step Condition](#)
- [Step Condition Details Field Descriptions](#)
- [Step Condition Logic](#)
- [Example Steps and Condition Codes](#)
 - [Example Job and Procedure](#)
 - [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 Details [Exit Code Processing](#) field specifies the method used to determine the task status for a z/OS batch job. When the **Step Conditions** 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 run time via the [Activity Monitor](#), 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 [Automation Center](#) navigation pane, select **Tasks > z/OS Tasks**. The z/OS Tasks list displays.

Step 2 Select the task for which you want to create one or more step conditions. The z/OS Task Details for that task displays.

z/OS Task Details: stonebranch-zOSTask-01

Update Launch Task Copy Delete Refresh Close

z/OS Task Step Conditions Restart Criteria Variables Actions Virtual Resources Mutually Exclusive Instances Triggers Notes Versions

General

Task Name: stonebranch-zOSTask-01 Version: 8

Task Description: IF/THEN/ELSE. - \${city}

Member of Business Services: [dropdown]

Hold on Start:

Virtual Resource Priority: 10 Hold Resources on Failure:

z/OS Details

Agent: stonebranch-zOSAgent-01

Agent Variable:

Credentials: stonebranch-credential-01 [dropdown]

Credentials Variable:

JCL Location: TESTCNTL(D237RC)

Use JCL Override Library:

New Jobname: [text] Procedure Library: [text]

New Jobclass: [text] Schedule Id: [text]

New Msgclass: [text]

Name	Value
*@CODE1	4
*@CODE2	8

Exit Code Processing: Success Exitcode Range [dropdown]

Exit Codes: 0-16 [text]

Automatic Output Retrieval: -- None -- [dropdown]

Retry Options

Auto-Restart Option: -- None -- [dropdown]

Maximum Retries: 0 Retry Indefinitely:

Retry Interval (Seconds): 60

Time Options

Late Start:

Late Finish:

Early Finish:

User Estimated Duration: [dropdown] [dropdown] [dropdown] HH:MM:SS

Statistics

First Time Ran: 2014-03-13 10:32:43 -0400 Lowest Instance Time: 0 Seconds

Last Time Ran: 2014-06-23 08:55:17 -0400 Average Instance Time: 1 Seconds

Last Instance Duration: 0 Seconds Highest Instance Time: 3 Seconds

Number of Instances: 23

Update Launch Task Copy Delete Refresh Close

Step 3 In the Exit Code Processing field, select **Step Conditions** from the drop-down list and then click the **Update** button.

Step 4 Click the **Step Conditions** tab. The Step Conditions list displays a list of any currently defined Step Conditions for this task.

Step	Procedure	Program	Condition Codes	Action	Updated By	Updated
S1	STEP1	IEBCGENER	0	Continue/Success	ops.admin	2014-07-01 14:27:04 -0400
S2	STEP2	IEBCGENER	8	Halt/Failed	ops.admin	2014-07-01 14:27:35 -0400

Step 5 Click **New**. The Step Condition Details pop-up dialog displays.

Step 6 Using the [field descriptions](#), below, as a guide, complete the fields as needed.

Step 7 Click the **Save** button, or right-click in the Details and then click **Save**, to save the record and return to the Step Conditions List.

Step 8 If appropriate, repeat these steps for any additional step conditions you want to add.

Step Condition Details Field Descriptions

The table below describes the fields and buttons in the Step Conditions Details pop-up dialog.

Field Name	Description
Step	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	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	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. For example: 1,6-4095,Sxxx,Unnnn,JCLERR
Action	Action to take and the task status to set if the step condition matches. See Step Condition Logic , below, for an explanation of the actions.

Buttons	This section identifies the buttons displayed above and below the Step Condition Details that let you perform various actions.
Save	Saves a new task record in the Controller database.
Update button	Saves updates to the record.
Refresh	Refreshes any dynamic data displayed in the Details.
Delete button	Deletes the current record.
Close	For pop-up view only; closes the pop-up view of this task.

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 (for example, 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 Controller 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, the Controller takes no action and normal JES processing of the job continues.



Note

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.

The Controller searches step condition definitions based on their order in the [Step Conditions list](#); the definition at the top of the list is searched first. To change the order of the definitions in the list, drag and drop them to any location.

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 the Controller 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, the Controller 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
```

User Interface Specifications and Actions

The following examples specify condition code checks for the example job above.

Example 1

The screenshot shows a dialog box titled "Step Condition Details". It has a title bar with standard window controls (minimize, maximize, close) and two buttons: "Save" and "Close". Below the title bar is a tab labeled "Step Condition". Inside the tab, there are several input fields: "Step:", "Procedure:", and "Program:". Below these is a "Condition Codes:" field containing the text "13-4095". At the bottom of the tab is an "Action:" dropdown menu currently set to "Halt/Failed". At the bottom of the dialog box are two buttons: "Save" and "Close".

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

The screenshot shows a dialog box titled "Step Condition Details". It has a "Step Condition" tab selected. The fields are:

- Step: S1
- Procedure: (empty text box)
- Program: (empty text box)
- Condition Codes: 8
- Action: Continue/Success (dropdown menu)

 There are "Save" and "Close" buttons at the top right and bottom left of the dialog.

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

The screenshot shows a dialog box titled "Step Condition Details". It has a "Step Condition" tab selected. The fields are:

- Step: (empty text box)
- Procedure: (empty text box)
- Program: IEBGENER
- Condition Codes: 0,12
- Action: Continue/Success (dropdown menu)

 There are "Save" and "Close" buttons at the bottom left of the dialog.

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 Details

Step Condition

Step : S1

Procedure : STEP2

Program :

Condition Codes : U0010

Action : Askoper

Save Close

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. The Controller 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 Details

Step Condition

Step : S1

Procedure : STEP3

Program :

Condition Codes : 0-7

Action : Continue/Success

Save Close

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

The screenshot shows a dialog box titled "Step Condition Details". It features a title bar with standard window controls (minimize, maximize, close) and two buttons: "Save" and "Close". Below the title bar is a tab labeled "Step Condition". The main content area contains the following fields:

- Step : S1
- Procedure : STEP1
- Program :
- Condition Codes : 1-4095
- Action : Continue/Failed (with a dropdown arrow)

At the bottom of the dialog, there are two larger buttons: "Save" and "Close".

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 Step Actions

- [Overview](#)
- [Creating a Step Action](#)
- [System Operation Step Action Field Descriptions](#)

Overview

You can specify actions to take on z/OS tasks in a Workflow based on [step condition codes](#) returned for any of the the steps in that z/OS task.

**Note**

Currently, the only step action that you can take on z/OS tasks in a workflow is a System Operation.

Step actions can be defined only at the [Workflow](#) level. They apply to one, more, or all z/OS tasks in that immediate Workflow; they do not apply to any z/OS tasks in sub-workflows. (Every task in a Workflow has a unique Vertex ID, which is how you can tell one task from another if the Workflow has more than one of the exact same task.)

When you create a step action for a task in the Workflow, you specify the name of the task in the **Task Id** field. If there is more than one of those tasks in the Workflow, the Vertex Id drop-down list shows the Id for all tasks. So you can apply the step action to all tasks of that name in the Workflow or just the task with that Vertex Id.

In the Workflow Editor, when you right-click a task, there's a **View/Edit Run Criteria** selection for every task. For z/OS tasks, there's also a **View/Edit z/OS Step Actions** selection.

Every z/OS task is comprised of one or more steps. Each step in a z/OS task ends in a step condition code. System Operation step actions optionally let you send system notifications based on the outcome of each step action with options for None, Operation Failure, Operation Success/Failure, and Operation Success.

Creating a Step Action

Step 1 From the [Automation Center](#) navigation pane, select **Tasks > Workflow Tasks**. The Workflow Tasks list displays.

Step 2 Select the Workflow for which you want to create one or more step actions. The Workflow Task Details for that Workflow displays.

The screenshot shows the 'Workflow Task Details' window for 'stonebranch-workflow-01'. The window has a title bar and a menu bar with options: Update, Launch Task, Edit Workflow, Copy, Delete, Refresh, and Close. Below the menu bar are several tabs: Workflow Task, Task Run Criteria, Variables, Actions, Virtual Resources, Mutually Exclusive, Step Conditions, Step Actions, Instances, and Triggers. The 'Workflow Task' tab is active, showing the following sections:

- General:** Task Name: stonebranch-workflow-01, Version: 7, Task Description: test, Member of Business Services: stonebranchbusinessservice 01, Hold on Start: , Virtual Resource Priority: 10, Hold Resources on Failure:
- Workflow Details:** Show/Hide Skipped Tasks: Show Skipped, Default Calendar: [dropdown]
- Time Options:** Late Start: , Late Finish: , Early Finish: , User Estimated Duration: [dropdown] [dropdown] [dropdown] HH:MM:SS
- Statistics:**
 - First Time Ran: 2014-04-17 15:20:10 -0400, Lowest Instance Time: 1 Minute 35 Seconds
 - Last Time Ran: 2014-06-18 15:15:41 -0400, Average Instance Time: 1 Minute 36 Seconds
 - Last Instance Duration: 1 Minute 35 Seconds, Highest Instance Time: 1 Minute 36 Seconds
 - Number of Instances: 2

At the bottom of the window, there is a row of buttons: Update, Launch Task, Edit Workflow, Copy, Delete, Refresh, and Close.

Step 3 Click the **Step Actions** tab. The System Operation step actions list displays.

The screenshot shows the 'Workflow Task Details' window for 'stonebranch-workflow-01' with the 'Step Actions' tab selected. The 'System Operation' section is expanded, showing a table with columns: Type Details, Task Id, Vertex Id, Step Criteria, Description, Updated By, and Updated. The table is currently empty, displaying 'No items to show.'

Step 4 Click **New**. The System Operation step actions pop-up dialog displays.

Step 5 Using the [field descriptions](#), below, as a guide, complete the fields as needed.


Step 6 Click the **Save** button, or right-click in the Details and then click **Save**, to save the record and return to the Step Actions list.

Step 7 If appropriate, repeat these steps for any additional step actions you want to add.

System Operation Step Action Field Descriptions

The following table describes the fields and buttons in the System Operation Step Action Details.

Field Name	Description
Step Action Criteria	This section contains criteria for performing the step action.
Task Id	Name of a task. In combination with the Vertex Id , it specifies a specific task within the Workflow to which the step action applies; if no task is specified, the step action applies to all z/OS tasks within the Workflow.
Vertex Id	Numerical ID of the task that identifies it uniquely from other tasks of the same type in the Workflow. In combination with the Task Id , it specifies a specific task within the Workflow to which the step action applies. Options: <ul style="list-style-type: none"> • Any - The action applies to any instance of the specified task in the Workflow. • <number> - The action applies only to this instance of the task in the Workflow.
Step	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	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	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.
Description	Description of this System Operation Step Action.

Step Action Details	This section contains assorted detailed information about the step action.
System Operation	<p>Specific system operation to perform.</p> <p>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 • Run Task Instance Command
System Notification	<p>Status of the specified system operation that will trigger a system notification.</p> <p>Options:</p> <ul style="list-style-type: none"> • None • Operation Failure (default) • Operation Success/Failure • Operation Success <div style="background-color: #ffffcc; padding: 5px; margin-top: 10px;"> <p> Note The 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 ; Indication of whether the Agent field is a reference field for selecting a specific Agent (unchecked) or is a text field for specifying the Agent as a variable (checked).
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 ; Indication of whether the Agent Cluster field is a reference field for selecting a specific Agent Cluster (unchecked) or is a text field for specifying the Agent Cluster as a variable (checked).
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 ; Indication of whether the Virtual Resource field is a reference field for selecting a specific Virtual Resource (unchecked) or is a text field for specifying the Virtual Resource as a variable (checked).
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>

Command	<p>If System Operation is Run Task Instance Command; Type of task instance command to run.</p> <p>Options:</p> <ul style="list-style-type: none"> • Cancel • Force Finish • Force Finish (Halt) • Force Finish/Cancel • Force Finish/Cancel (Halt) • Skip • Unskip • Hold • Release • Release Recursive • Clear All Dependencies • Clear Exclusive • Clear Predecessors • Clear Resources
Instance Lookup Option	<p>If System Operation is Run Task Instance Command; Specification for how to search for the task instance to run a command against.</p> <p>Options:</p> <ul style="list-style-type: none"> • Instance Name • Instance Name/Task • Instance Id • Task
Instance Name	<p>If Instance Lookup Option is Instance Name or Instance Name/Task; Name of the task instance to run the command against.</p>
Instance Criteria	<p>If Instance Lookup Option is Instance Name, Instance Name/Task, or Task; Additional criteria for selecting a specific task instance if multiple task instances have matching names.</p> <ul style="list-style-type: none"> • Newest Active Instance • Oldest Active Instance • Newest Instance (Unskip command only) • Oldest Instance (Unskip command only) <p>(An Active task instance is an instance that is not in any of these statuses: Skipped, Finished, Success.)</p>
Workflow Instance Name	<p>If Instance Lookup Option is Instance Name, Instance Name/Task, or Task; Name of the Workflow in which the specified task instance is contained.</p>
Task	<p>If Instance Lookup Option is Instance Name/Task or Task; Name of the task for which the task instance was run.</p>
Task Variable	<p>If Instance Lookup Option is Instance Name/Task or Task; Indication of whether the Task field is a reference field for selecting a specific Task (unchecked) or is a text field for specifying the Task as a variable (checked). For a variable, use the format: <code>\${variable name}</code>. The variable must be a supported type as described in Variables and Functions.</p>
Instance ID	<p>If Instance Lookup Option is Instance ID; ID of task instance to run the command against. The instance ID (sysid) is a 32-digit hexadecimal number. You can use the <code>\${ops_task_id}</code> variable or <code>\${_siblingid('mytask')}</code> function to get the instance id.</p>
Buttons	<p>This section identifies the buttons displayed above and below the Step Action Details that let you perform various actions.</p>
Save	<p>Saves a new task record in the Controller database.</p>
Update	<p>Saves updates to the record.</p>
Refresh	<p>Refreshes any dynamic data displayed in the Details.</p>
Delete	<p>Deletes the current record.</p>
Close	<p>For pop-up view only; closes the pop-up view of this task.</p>

Creating Restart Criteria

- [Overview](#)
- [Creating a Restart Criterion](#)
- [Restart Criteria Field Descriptions](#)
- [Restart Criteria Logic](#)
- [Restart Criteria Details - Directives](#)

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 task ends in a failed state, some or all of the job steps may need to be [re-run](#). The set of steps that should be re-run (if any) may vary depending on how and where the task failed. If the task is defined with [Auto-Restart Option = Use Restart Criteria](#), the Restart Criteria is used to select the appropriate steps to restart based on a particular failure scenario.

Creating a Restart Criterion

Step 1 From the [Automation Center](#) navigation pane, select **Tasks > z/OS Tasks**. The z/OS Tasks list displays.

Step 2 Select the task for which you want to create one or more restart criteria. The z/OS Task Details for that task displays.

z/OS Task Details: stonebranch-zOSTask-01

Update Launch Task Copy Delete Refresh Close

z/OS Task Step Conditions Restart Criteria Variables Actions Virtual Resources Mutually Exclusive Instances Triggers Notes Versions

General

Task Name: stonebranch-zOSTask-01 Version: 8

Task Description: IF/THEN/ELSE. - \${city}

Member of Business Services: [dropdown]

Hold on Start:

Virtual Resource Priority: 10 Hold Resources on Failure:

z/OS Details

Agent: stonebranch-zOSagent-01

Agent Variable:

Credentials: stonebranch-credential-01 [dropdown]

Credentials Variable:

JCL Location: TESTCNTL(D237RC)

Use JCL Override Library:

New Jobname: [text] Procedure Library: [text]

New Jobclass: [text] Schedule Id: [text]

New Msgclass: [text]

Name	Value
*@CODE1	4
*@CODE2	8

Exit Code Processing: Success Exitcode Range [dropdown]

Exit Codes: 0-16 [text]

Automatic Output Retrieval: -- None -- [dropdown]

Retry Options

Auto-Restart Option: -- None -- [dropdown]

Maximum Retries: 0 Retry Indefinitely:

Retry Interval (Seconds): 60

Time Options

Late Start:

Late Finish:

Early Finish:

User Estimated Duration: [dropdown] [dropdown] [dropdown] HH:MM:SS

Statistics

First Time Ran: 2014-03-13 10:32:43 -0400 Lowest Instance Time: 0 Seconds

Last Time Ran: 2014-06-23 08:55:17 -0400 Average Instance Time: 1 Seconds

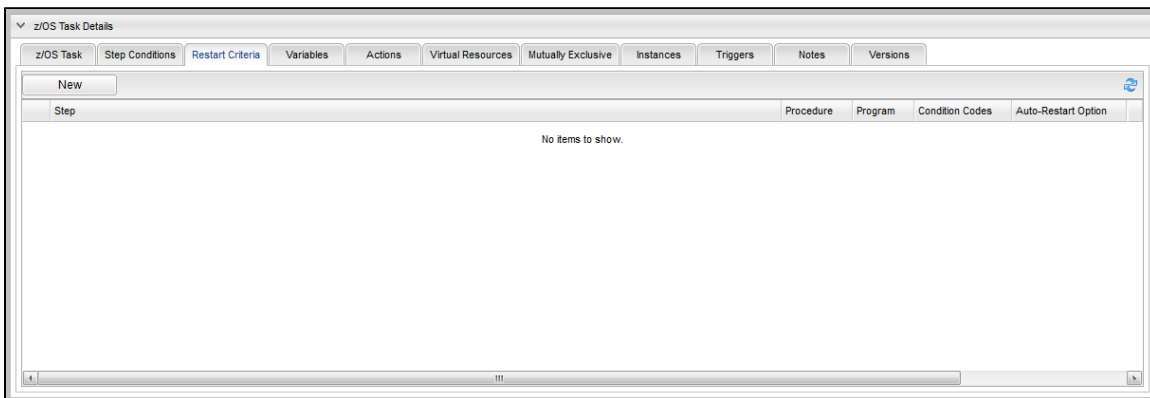
Last Instance Duration: 0 Seconds Highest Instance Time: 3 Seconds

Number of Instances: 23

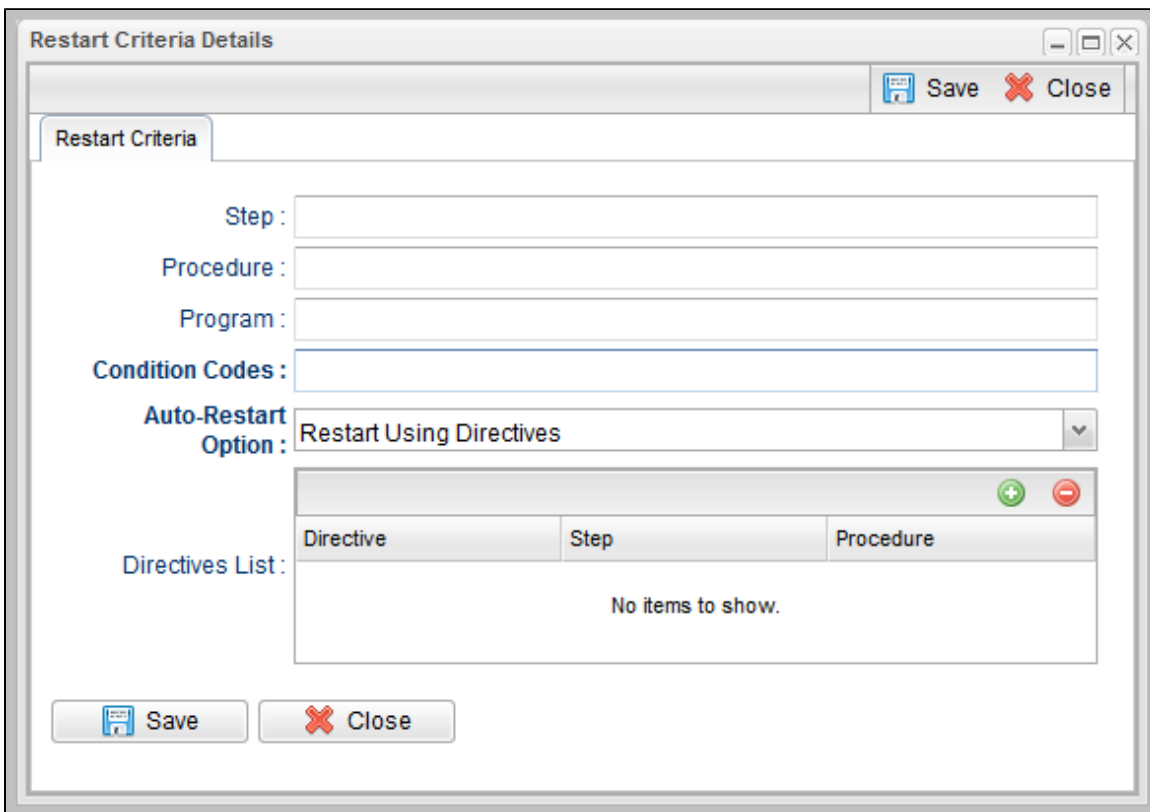
Update Launch Task Copy Delete Refresh Close

Step 3 In the Auto-Restart Option field, select **Use Restart Criteria** from the drop-down list and then click the **Update** button.

Step 4 Click the **Restart Criteria** tab. The Restart Criteria list displays.



Step 5 Click **New**. Restart Criteria Details displays.



Step 6 Using the field descriptions provided below as a guide, complete the fields as needed.

Step 7 Click the **Save** button to save the record and return to the **Restart Criteria** list, or access the **Action** menu and select **Save** to save the record.

Step 8 If appropriate, repeat these steps for any additional restart criteria that you want to add.

Restart Criteria Field Descriptions

The following table describes the fields and buttons in the [Restart Criteria Details](#).

Field Name	Description
Step	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	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	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. For example: 1,6-4095,Sxxx,Unnnn,JCLERR
Auto-Restart Option	Method of step selection to perform for the restart. See Restart Criteria Logic , below, for an explanation of the options.
Directives List	(See Restart Criteria Details - Directives , below.)

Restart Criteria Logic

Restart Criteria are processed when a task transitions to a Failed status, provided that:

- [Auto-Restart Option](#) = **Use Restart Criteria**.
- [Maximum Retries](#) = greater than zero.

When restart criteria processing begins, the Controller will search the list of restart criteria definitions to find a match for the current failure scenario. Matching is based on 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 restart criteria is found, the Auto-Restart Option for that restart criteria is performed. If no matching restart criteria is found, the Controller takes no action and no job steps will be selected for restart.

The Controller searches restart criteria definitions based on their order in the [Restart Criteria list](#); the definition at the top of the list is searched first. To change the order of the definitions in the list, drag and drop them to any location.

The Restart Criteria Details **Auto-Restart Option** specifies how steps will be selected for restart.

The following Auto-Restart Options are supported:

Restart From First Job Step	All restartable job steps, from first to last, will be selected for restart.
Restart From Matching Job Step	All restartable job steps, from the step that matched the restart criteria to the last job step, will be selected for restart.
Restart Using Directives	Job steps will be selected for restart based on the directives specified in the restart criteria directive table.

Restart Criteria Details - Directives

The following table describes the fields and button in the Directives section of the [Restart Criteria Details](#).

Field Name	Description
Directive	Specifies a directive for step selection. Options: <ul style="list-style-type: none"> • Start - Directive used to specify the starting step in a range of steps selected for restart. The directive table can contain zero or one Start directives. If a start directive is not specified, the first job step will be used as the starting step. • Skip - Directive used to specify a step that should be skipped. Skipped steps will not be restarted. The directive table can contain zero or more Skip directives. • End - Directive used to specify the ending step in a range of steps that will be selected for restart. The directive table can contain zero or one End directives. If an end directive is not specified, the last job step will be used as the ending job step.

Step	<p>The directive step optionally specifies:</p> <ol style="list-style-type: none"> 1. A relative step number. Relative step numbers begin with + or -, followed by some number of steps to offset from the step matched by the restart criteria. A value of -0 or +0 indicated that the directive is referring to the job step that matched the restart criteria. <p>Relative step numbers are mutually exclusive with the Procedure field.</p> <p>If a relative step number is specified, no matching is performed. The directive will apply to an explicit offset from the job step that matched the restart criteria definition.</p> <ol style="list-style-type: none"> 2. 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	<p>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.</p>
Add button	<p>Adds the directive defined by the fields above to the directive table.</p>

Indesca Task

- [Before You Begin](#)
- [Creating an Indesca Task](#)
 - [Indesca Task Details](#)
 - [Indesca Task Details Field Descriptions](#)
- [Viewing an Indesca Task Instance](#)
 - [Indesca Task Instance Details](#)
 - [Indesca Task Instance Details Field Descriptions](#)
- [Output Redirection](#)
- [Running an Indesca Task](#)
- [Monitoring Task Execution](#)

Before You Begin

The Indesca task allows you to run a platform-specific application on a machine where Opwise Universal Agent is running. Indesca is functionality provided by the Agent that 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:

- [Install](#) an Agent on the target machine.
- Launch the Agent. When the Agent connects with the Controller, it automatically creates an [Agent resource definition](#) in the Controller database.



Note

If you are running multiple **ubroker** started tasks and want to run an Indesca task on a specific **ubroker**, you must specify the service port number associated with that **ubroker** in the [Indesca Options](#) field on the Indesca Task Details. The syntax is `-port nnnn`.

Creating an Indesca Task

Step 1 From the [Automation Center](#) navigation pane, select **Tasks > Indesca Tasks**. The Indesca Tasks list displays a list of all currently defined Indesca tasks.

Below the list, Indesca Task Details for a new Indesca task displays. (You also can click the **New** button to display Indesca Task Details for a new Indesca task.)

The screenshot shows the 'Indesca Tasks' interface. At the top, there is a list of 5 tasks:

Task Name	Task Description	Command or Script	Updated By	Updated
stonebranch-indescatask-01		Command	ops.admin	2014-06-13 14:02:24 -0400
stonebranch-indescatask-02		Command	ops.admin	2014-06-13 14:02:35 -0400
stonebranch-indescatask-03		Command	ops.admin	2014-06-13 14:02:46 -0400
stonebranch-indescatask-04		Command	ops.admin	2014-06-13 14:02:54 -0400
stonebranch-indescatask-05		Command	ops.admin	2014-06-13 14:03:00 -0400

Below the list is the 'Indesca Task Details' form. The 'General' section contains the following fields:

- Task Name**: [Text Field]
- Version**: [Text Field] (value: 1)
- Task Description**: [Text Field]
- Member of Business Services**: [Dropdown Menu]
- Hold on Start**:
- Virtual Resource Priority**: [Text Field] (value: 10)
- Hold Resources on Failure**:

The 'Indesca Details' section contains the following fields:

- Utility Agent**: [Dropdown Menu]
- Utility Agent Variable**:
- Utility Credentials**: [Dropdown Menu]
- Utility Credentials Variable**:
- Utility Agent Cluster**: [Dropdown Menu]
- Cluster Variable**:

Step 2 Enter/select Details for a new Indesca task, using the field descriptions below as a guide.

- Required fields display in **boldface**.
- Default values for fields, if available, display automatically.

To display more of the Details fields on the screen, you can temporarily [hide the list](#).



Note

If you view [Indesca Task Details](#) for an existing Indesca task by clicking a task in the list, and then want to create a new Indesca task, you must click the **New** button that displays above and below the Details.

Step 3 Click the **Save** button. The task is added to the database, and all buttons and tabs in the Task Details are enabled.

Indesca Task Details



The following Indesca Task Details is for an existing Indesca task.



Depending on the values that you enter / select for these fields, and whether or not the Indesca task has ever been launched, more (or less) fields may display. See the [field descriptions](#), below, for a description of all fields that may display in the Indesca Task Details.

Indesca Task Details Field Descriptions

The following table describes the fields, buttons, and tabs that display in the Indesca Task Details.

Field Name	Description
General	This section contains general information about the task.
Task Name	User-defined name of this task, containing a maximum 255 alphanumeric characters; variables supported. It is the responsibility of the user to develop a workable naming scheme for tasks.
Version	System-supplied; version number of the current record, which is incremented by the Controller every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .


Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
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.
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.
Indesca Details	This section contains assorted detailed information about the task.
Utility Agent	Name of the Windows or Linux/Unix Agent resource that will communicate with the Universal Command Agent. If you do not specify an Agent, you must specify an agent cluster .
Utility Agent Variable	If enabled, the Utility 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. Use the format: <code>\$(variable name)</code> . The variable must be a supported type as described in Variables and Functions . <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Utility Agent Variable when updating multiple tasks, the Utility Agent field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Utility Agent Unresolved - into which you enter a variable, variable and text, or Utility Agent name. The Utility Agent field is not used.</p> </div>
Utility Agent Cluster	Group of Agents, one of which the Controller will choose to run this task. You can specify an agent cluster in addition to or in place of a specific Agent. If you specify an Agent and an agent cluster, the Controller first tries to run the task on the specific Agent. If the Agent is not available, the Controller reverts to the agent cluster. See Agent Clusters for more information.
Cluster Variable	If enabled, the Utility 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. Use the format: <code>\$(variable name)</code> . The variable must be a supported type as described in Variables and Functions . <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Utility Agent Cluster Variable when updating multiple tasks, the Utility Agent Cluster field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Utility Agent Cluster Unresolved - into which you enter a variable, variable and text, or Utility Agent Cluster name. The Utility Agent Cluster field is not used.</p> </div>

Utility Credentials	Login credentials that the Agent will use to access the Universal Command server machine.
Utility Credentials Variable	<p>If enabled, the Utility Credentials field 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> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Utility Credentials Variable when updating multiple tasks, the Utility Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Utility Credentials Unresolved - into which you enter a variable, variable and text, or Utility Credentials name. The Utility Credentials field is not used.</p> </div>
Indesca Agent	<p>Depending on the value in the Indesca Agent Option field, this field contains either:</p> <ul style="list-style-type: none"> Record name from the Indesca Agent table. Variable that will be resolved when the task is launched. Host name of a machine where the Indesca Agent is running.
Indesca Agent Option	<p>Specifies how the name of the Indesca Agent is being supplied in the Indesca Agent field.</p> <p>Options:</p> <ul style="list-style-type: none"> Indesca Agent - Indesca Agent record is selected from the Indesca Agent table. Indesca Agent Variable - Indesca Agent field contains a variable that will be resolved when the task is launched. Indesca Agent Hostname - Indesca Agent field contains the host name where the Indesca Agent is running. The host name must be accessible by the Controller.
Indesca Credentials	Login credentials that Controller will use to access the remote machine where the Indesca Agent is running. See Credentials .
Indesca Credentials Variable	<p>If enabled, the Indesca 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.</p> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Indesca Credentials Variable when updating multiple tasks, the Indesca Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Indesca Credentials Unresolved - into which you enter a variable, variable and text, or Indesca Credentials name. The Indesca Credentials field is not used.</p> </div>
Command or Script	<p>Specifies whether a single command or a script is being executed.</p> <p>Options:</p> <ul style="list-style-type: none"> Command (default) Script
Command	If Command or Script = Command; Required. Command being executed on the remote machine. Variables supported.
Script File	If Command or Script = Script; Required. Path and filename of the script file that will be executed on the remote machine.

Script Options	If Command or Script = Script; Optional. One or more command line options to pass to the script file.
Indesca Options	Any Indesca options needed by the program to execute properly. Variables supported.
Runtime Directory	Directory from which the application should be executed. Variables supported.
Exit Code Processing	Specifies how the Controller should determine whether the executed command failed or completed successfully. Options: <ul style="list-style-type: none"> • Success Exitcode Range - Command is considered completed successfully if its exit code falls within the range specified in the Exit Codes field. • Failure Exitcode Range - Command is considered failed if its exit code falls within the range specified in the Exit Codes field. • Success Output Contains - Command is considered completed successfully if its output contains the text specified in the Scan Output For field. • Failure Output Contains - Command is considered failed if its output contains the text specified in the Scan Output For field. • Step Conditions (z/OS only) - 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; type of output. Options: <ul style="list-style-type: none"> • Standard Output (STDOUT) • Standard Error (STDERR) • File
Scan Output For	Required if Exit Code Processing = Success Output Contains or Failure Output Contains; text for which the Controller should scan the output file. The Controller will process this field as a regular expression.
Output File (for Exit Code Processing)	Required if Output Type = File; path and file name of the output file that should be scanned for the text in the Scan Output For field.
Exit Codes	Required if Exit Code Processing = Success Exitcode Range or Failure Exitcode Range; range of exit codes. Format: Numeric. Use commas to list a series of exit codes; use hyphens to specify a range. Example: 1,5, 22-30.
Automatic Output Retrieval	Specifies whether you want the Controller 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. • Standard Output/Error - Attach all standard output and standard error output.
Start Line	If Automatic Output Retrieval = Standard Output, Standard Error, File, or Standard Output/Error; Instructs the Controller to retrieve data beginning at the line indicated. <ul style="list-style-type: none"> • If a Start Line value is not specified, the default is 1. • If the Start Line value is -1, data will be retrieved starting at the end of the file.

Number of Lines	If Automatic Output Retrieval = Standard Output, Standard Error, File, or Standard Output/Error; 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 Controller system property.
Scan Text	If Automatic Output Retrieval = Standard Output, Standard Error, File, or Standard Output/Error; Regex pattern that the Controller will search for a match for in STDOUT/STDERR or a specified file. The Controller will include the Number of Lines above and below the first line matched. if the Regex pattern is not found, the following message is returned: OPSWISE WARNING - Scan text string not found.
Output File (for Automatic Output Retrieval)	Required if Automatic Output Retrieval = File; path and file name containing the output that you want automatically retrieved and attached to the task instance.
Retry Options	This section contains specifications for retrying the task.
Retry Exit Codes	Exit code range for which an auto-retry of tasks in FAILED status will occur. Exit code ranges must be in the same format as ranges specified in the Exit Codes field. Maximum Retries must be greater than 0. If this field is empty, any exit code potentially will cause a retry.
Maximum Retries	User-defined; maximum number of times that the Controller should retry this task after it has started and gone to a failed state.
Retry Indefinitely	User-defined; indicates whether the Controller should continue trying indefinitely to run this task. If you enable this field, it overrides any value placed in the Maximum Retries field.
Retry Interval (Seconds)	User-defined; number of seconds between each retry.
Time Options	This section contains time-related specifications for the task.
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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.
Late Start Type	Required if Late Start is enabled. Options: <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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.
Late Start Duration	If Late Start Type = 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.

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: <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 = Time; Time after which the task finish time is considered late. Use HH:MM, 24-hour time.
Late Finish Duration	If Late Finish Type = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
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: <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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
User Estimated Duration	Estimated amount of time it should normally take to run this task. The Controller uses this information to calculate the User Estimated End Time on a task instance record.
Statistics	This section contains time-related statistics for task instances of the task.
First Time Ran	System-supplied; date and time this task first ran.
Last Time Ran	System-supplied; date and time the task last ran.
Last Instance Duration	System-supplied; amount of time the task took to run the last time it ran.
Lowest Instance Time	System-supplied; shortest amount of time this task has taken to run.

Average Instance Time	System-supplied; average amount of time this task takes to run.
Highest Instance Time	System-supplied; longest amount of time this task has taken to run.
Number of Instances	System-supplied; number of instances in the database for this task.
Buttons	This section identifies the buttons displayed above and below the Task Details that let you perform various actions.
Save	Saves a new task record in the Controller database.
Update	Saves updates to the record.
New	Displays empty (except for default values) Details for creating a new task.
Launch Task	Manually launches the task.
View Parents	Displays a list of any parent Workflow tasks for this task.
Copy	Creates a copy of this task, which you are prompted to rename.
Delete	<p>Deletes the current record.</p> <div style="background-color: #ffffcc; padding: 10px; margin: 10px 0;"> <p> Note You cannot delete a task if it is either:</p> <ul style="list-style-type: none"> Specified in an enabled Trigger. The only task specified in a disabled Trigger. </div>
Refresh	Refreshes any dynamic data displayed in the Details.
Close	For pop-up view only; closes the pop-up view of this task.
Tabs	This section identifies the tabs across the top of the Task Details that provide access to additional information about the task.
Variables	Lists all variables associated with this record.

Actions	<p>Allows you to specify actions that the Controller will take automatically based on events that occur during the execution of this task.</p> <p>Events are:</p> <ul style="list-style-type: none"> • Task instance status • Exit codes • Late start • Late finish • Early finish <p>Actions are:</p> <table border="1" data-bbox="266 478 1484 840"> <tr> <td data-bbox="266 478 412 548">Abort Action</td> <td data-bbox="412 478 1484 548">Abort the task if certain events occur. For details, see Abort Actions.</td> </tr> <tr> <td data-bbox="266 548 412 617">Email Notification</td> <td data-bbox="412 548 1484 617">Send an email if certain events occur. For details, see Email Notification Actions.</td> </tr> <tr> <td data-bbox="266 617 412 686">Set Variable</td> <td data-bbox="412 617 1484 686">Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow.</td> </tr> <tr> <td data-bbox="266 686 412 756">SNMP Notification</td> <td data-bbox="412 686 1484 756">Send an email if certain events occur. For details, see SNMP Notification Actions.</td> </tr> <tr> <td data-bbox="266 756 412 825">System Operation</td> <td data-bbox="412 756 1484 825">Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions.</td> </tr> </table>	Abort Action	Abort the task if certain events occur. For details, see Abort Actions .	Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .	Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .	SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .	System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .
Abort Action	Abort the task if certain events occur. For details, see Abort Actions .										
Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .										
Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .										
SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .										
System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .										
Virtual Resources	Lists all Virtual Resources to which this task is assigned.										
Instances	Lists all instances of the task.										
Mutually Exclusive	Lists all tasks that have been set to be mutually exclusive of this task.										
Triggers	Lists 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, the Controller 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 Triggers Overview .										
Notes	Lists all notes associated with this record.										
Versions	Stores copies of all previous versions of the current record. See Record Versioning .										

Viewing an Indesca Task Instance

When an Indesca task is launched, the Controller creates a task instance record of that task.

A task instance contains detailed information about a single execution of that task.

You can access a task instance from:

- **Instances tab** on the [Indesca Task Details](#) for that task
- [Activity Monitor](#)
- [Task Instances list](#)

Indesca Task Instance Details

The following Indesca Task Instance Details contains information on the execution of the task shown in the [Indesca Task Details](#).

The screenshot displays the 'Indesca Task Instance Details' window for 'stonebranch-indescatask-01'. The interface includes a title bar with standard window controls and a menu bar with options like 'Update', 'Re-run', 'View Parent', 'Retrieve Output...', 'Delete', 'Refresh', and 'Close'. Below the menu bar are tabs for 'Indesca Task Instance', 'Virtual Resources', 'Exclusive Requests', 'Output', and 'Notes'. The main content is organized into several sections:

- General:** Fields for Instance Name (stonebranch-indescatask-01), Reference Id (28), Task (stonebranch-indescatask-01), Invoked By (Trigger: stonebranch-limtrigger-01), Task Description, Member of Business Services, Execution User (stonebranch-user-01), Virtual Resource Priority (10), and Hold Resources on Failure (checkbox).
- Status:** Status (Success), Exit Code (0), Status Description, Queued Time (2015-02-06 14:45:37 -0500), Trigger Time, Launch Time (2015-02-06 14:45:37 -0500), Start Time (2015-02-06 14:45:41 -0500), End Time (2014-06-20 13:09:49 -0400), and Duration (0 Seconds).
- Indesca Details:** Utility Agent (stonebranch-agent-01), Utility Agent Cluster, Utility Agent Variable, Utility Credentials (stonebranch-credential-01), Utility Credentials Variable, Indesca Agent (stonebranch-agent-02), Indesca Agent Option (Indesca Agent), Indesca Credentials (stonebranch-credential-02), Indesca Credentials Variable, Command or Script (Command), Command (ls), Indesca Options, Runtime Directory, Exit Code Processing (Success Exitcode Range), Exit Codes (0), Automatic Output Retrieval (Standard Output/Error), Start Line (1), Number of Lines (100), and Scan Text.
- Retry Options:** Maximum Retries (0), Retry Interval (Seconds) (60), Retry Indefinitely (checkbox), and Current Retry Count (0).
- Statistics:** User Estimated End Time, Average Estimated End Time (2014-06-26 14:32:01 -0400), Shortest Estimated End Time (2014-06-26 14:32:01 -0400), and Longest Estimated End Time (2014-06-26 14:32:02 -0400).




At the bottom of the window, there is another set of buttons: 'Update', 'Re-run', 'View Parent', 'Retrieve Output...', 'Delete', 'Refresh', and 'Close'.


Indesca Task Instance Details Field Descriptions

The following table describes the fields, buttons, and tabs that display in Indesca Task Instance Details.

Field Name	Description
General	This section contains general information about the task instance.

Instance Name	Name of this task instance.
Reference Id	System-supplied; the Controller increments this number each time the task is run.
Task	Name of the task that was run to create this task instance. Click the icon to display Task Details for the task.
Invoked by	System-supplied; how the task instance was launched. Options: <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 identify the user, check the Execution User column for that task instance on the Task Instances screen or, on most task instance screens, the Execution User field.
Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
Execution User	System-supplied; if the task was launched manually, the ID of the user who launched it.
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.
Status	This section contains information about the current status of the task instance.
Status	System-supplied; see Task Instance Statuses .
Exit Code	System-supplied; the exit code captured by the Agent when executing the task (for example, a command or script).
Status Description	System-supplied; additional information, if any, about the status of the task instance.
Queued Time	System-supplied; Date and time the task was queued for processing.
Trigger Time	System-supplied; Date and time the task instance was triggered.
Launch Time	System-supplied; Date and time the task instance was launched.
Start Time	System-supplied; Date and time the task instance started.

End Time	System-supplied; Date and time the task instance completed.
Duration	System-supplied; amount of time the task instance took to run.
Indesca Details	This section contains assorted detailed information about the task instance.
Utility Agent	Name of the Windows or Linux/Unix Agent resource that will communicate with the Universal Command Agent. If you do not specify an Agent, you must specify an agent cluster .
Utility Agent Variable	<p>If enabled, the Utility 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. Use the format: <code>\${variable name}</code>. The variable must be a supported type as described in Variables and Functions.</p> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Utility Agent Variable when updating multiple tasks, the Utility Agent field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Utility Agent Unresolved - into which you enter a variable, variable and text, or Utility Agent name. The Utility Agent field is not used.</p> </div>
Utility Agent Cluster	Group of Agents, one of which the Controller will choose to run this task. You can specify an agent cluster in addition to or in place of a specific Agent. If you specify an Agent and an agent cluster, the Controller first tries to run the task on the specific Agent. If the Agent is not available, the Controller reverts to the agent cluster. See Agent Clusters for more information.
Cluster Variable	<p>If enabled, the Utility 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. Use the format: <code>\${variable name}</code>. The variable must be a supported type as described in Variables and Functions.</p> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Utility Agent Cluster Variable when updating multiple tasks, the Utility Agent Cluster field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Utility Agent Cluster Unresolved - into which you enter a variable, variable and text, or Utility Agent Cluster name. The Utility Agent Cluster field is not used.</p> </div>
Utility Credentials	Login credentials that the Agent will use to access the Universal Command server machine.
Utility Credentials Variable	<p>If enabled, the Utility Credentials field 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> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Utility Credentials Variable when updating multiple tasks, the Utility Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Utility Credentials Unresolved - into which you enter a variable, variable and text, or Utility Credentials name. The Utility Credentials field is not used.</p> </div>

Indesca Agent	<p>Depending on the value in the Indesca Agent Option field, this field contains either:</p> <ul style="list-style-type: none"> Record name from the Indesca Agent table. Variable that will be resolved when the task is launched. Host name of a machine where the Indesca Agent is running.
Indesca Agent Option	<p>Specifies how the name of the Indesca Agent is being supplied in the Indesca Agent field.</p> <p>Options:</p> <ul style="list-style-type: none"> Indesca Agent - Indesca Agent record is selected from the Indesca Agent table. Indesca Agent Variable - Indesca Agent field contains a variable that will be resolved when the task is launched. Indesca Agent Hostname - Indesca Agent field contains the host name where the Indesca Agent is running. The host name must be accessible by the Controller.
Indesca Credentials	<p>Login credentials that Controller will use to access the remote machine where the Indesca Agent is running. See Credentials.</p>
Indesca Credentials Variable	<p>If enabled, the Indesca 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.</p> <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note If you select Indesca Credentials Variable when updating multiple tasks, the Indesca Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Indesca Credentials Unresolved - into which you enter a variable, variable and text, or Indesca Credentials name. The Indesca Credentials field is not used.</p> </div>
Command or Script	<p>Specifies whether a single command or a script is being executed.</p> <p>Options:</p> <ul style="list-style-type: none"> Command (default) Script
Command	<p>If Command or Script = Command; Required. Command being executed on the remote machine. Variables supported.</p>
Script File	<p>If Command or Script = Script; Required. Path and filename of the script file that will be executed on the remote machine.</p>
Script Options	<p>If Command or Script = Script; Optional. One or more command line options to pass to the script file.</p>
Indesca Options	<p>Any Indesca options needed by the program to execute properly. Variables supported.</p>
Runtime Directory	<p>Directory from which the application should be executed. Variables supported.</p>

Exit Code Processing	<p>Specifies how the Controller should determine whether the executed command failed or completed successfully.</p> <p>Options:</p> <ul style="list-style-type: none"> • Success Exitcode Range - Command is considered completed successfully if its exit code falls within the range specified in the Exit Codes field. • Failure Exitcode Range - Command is considered failed if its exit code falls within the range specified in the Exit Codes field. • Success Output Contains - Command is considered completed successfully if its output contains the text specified in the Scan Output For field. • Failure Output Contains - Command is considered failed if its output contains the text specified in the Scan Output For field. • Step Conditions (z/OS only) - 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	<p>Required if Exit Code Processing = Success Output Contains or Failure Output Contains; type of output.</p> <p>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 Exitcode Range; range of exit codes. Format: Numeric. Use commas to list a series of exit codes; 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; text for which the Controller should scan the output file. The Controller will process this field as a regular expression.</p>
Output File (for Exit Code Processing)	<p>Required if Output Type = File; path and file name of the output file that should be scanned for the text in the Scan Output For field.</p>
Automatic Output Retrieval	<p>Specifies whether you want the Controller to automatically retrieve any output from the job and attach it to the task instance record.</p> <p>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. • Standard Output/Error - Attach all standard output and standard error output.
Start Line	<p>If Automatic Output Retrieval = Standard Output, Standard Error, File, or Standard Output/Error; Instructs the Controller to retrieve data beginning at the line indicated.</p> <ul style="list-style-type: none"> • If a Start Line value is not specified, the default is 1. • If the Start Line value is -1, data will be retrieved starting at the end of the file.
Number of Lines	<p>If Automatic Output Retrieval = Standard Output, Standard Error, File, or Standard Output/Error; 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 Controller system property.</p>

Scan Text	<p>If Automatic Output Retrieval = Standard Output, Standard Error, File, or Standard Output/Error; Regex pattern that the Controller will search for a match for in STDOUT/STDERR or a specified file. The Controller will include the Number of Lines above and below the first line matched.</p> <p>if the Regex pattern is not found, the following message is returned: OPSWISE WARNING - Scan text string not found.</p>
Output File (for Automatic Output Retrieval)	Required if Automatic Output Retrieval = File; path and file name containing the output that you want automatically retrieved and attached to the task instance.
Retry Options	This section contains specifications for retrying the task.
Retry Exit Codes	<p>Exit code range for which an auto-retry of tasks in FAILED status will occur. Exit code ranges must be in the same format as ranges specified in the Exit Codes field. Maximum Retries must be greater than 0.</p> <p>If this field is empty, any exit code potentially will cause a retry.</p>
Maximum Retries	User-defined; maximum number of times that the Controller should retry this task after it has started and gone to a failed state.
Retry Indefinitely	User-defined; indicates whether the Controller should continue trying indefinitely to run this task. If you enable this field, it overrides any value placed in the Maximum Retries field.
Retry Interval (Seconds)	User-defined; number of seconds between each retry.
Current Retry Count	System-supplied; current number of times that the Controller has retried the task after it first went to failure status.
Time Options	This section contains time-related specifications for the task instance.
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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.
Started Late	System-supplied; this field is flagged if the task started later than the time specified in the Late Start fields.
Late Start Type	<p>Required if Late Start is enabled.</p> <p>Options:</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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.

Late Start Duration	<p>If Late Start Type = 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.</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>
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>
Finished Late	<p>System-supplied; this field is flagged if the task finished later than the time or duration specified in the Late Finish fields.</p>
Late Finish Type	<p>Required if Late Finish is enabled.</p> <p>Options:</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 = Time; 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 = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.</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>
Finished Early	<p>System-supplied; this field is flagged if the task finished earlier than the time specified in the Early Finish fields.</p>
Early Finish Type	<p>Required if Early Finish is enabled.</p> <p>Options:</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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.</p>
Statistics	<p>This section contains time-related statistics for the task instance.</p>

User Estimated End Time	System-supplied; if the user entered information into the User Estimated Duration field in the task Details, the Controller 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	System-supplied; shortest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Average Estimated End Time	System-supplied; average estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Longest Estimated End Time	System-supplied; longest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Buttons	This section identifies the buttons displayed above and below the Task Instance Details that let you perform various actions.
Update	Saves updates to the record.
Force Finish	See Force Finishing a Task .
Hold	Places the task instance on Hold (see Putting a Task on Hold).
Skip	For tasks loaded into the schedule that have not yet run; allows you to tell the Controller to skip this task. See Skipping a Task .
Re-run	See Re-running a Task .
View Parent	Displays the task instance Details for the parent Workflow of this task instance.
Retrieve Output	See Retrieving Output .
Delete	Deletes the current record.
Refresh	Refreshes any dynamic data displayed in the Details.
Close	For pop-up view only; closes the pop-up view of this task instance.
Tabs	This section identifies the tabs across the top of the Task Instance Details that provide access to additional information about the task instance.
Virtual Resources	Lists all Virtual Resources to which this task is assigned.
Exclusive Requests	Lists all records in the Exclusive Requests table (<code>ops_exclusive_order</code>) for this task instance.
Output	<p>Displays output generated from the process, if any, based on specifications provided by the user in the Automatic Output Retrieval fields in the task Details.</p> <p>If automatic output retrieval was not available or was not selected, output can be obtained by clicking the Retrieve Output button.</p>

Notes	Lists all notes associated with this record.
--------------	--

Output Redirection

An Agent 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"
```

Running an Indesca Task

You can run an Indesca task:

- Manually, by clicking the [Launch Task](#) or [Launch Task with Variables](#) button in the Indesca tasks list or Indesca Task Details [Action menu](#).
- As part of a [workflow](#).
- [Specify triggers](#) that run the task automatically based on times or events.

Monitoring Task Execution

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

SAP Task

- Overview
- Before You Begin
- Creating an SAP Task
 - SAP Task Details
 - SAP Task Details Field Descriptions
- Viewing an SAP Task Instance
 - SAP Task Instance Details
 - SAP Task Instance Details Field Descriptions
- Output Redirection
- Universal Connector Commands
- Built-In Variables
- Running an SAP Task
- 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 Opwise Controller 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 [Opwise Universal Agent for Linux/Unix](#) that will interface with the SAP system.
- Define an [SAP connection](#) in the Controller database.

Creating an SAP Task

Step 1 From the Automation Center navigation pane, select **Tasks > SAP Tasks**. The SAP Tasks list displays a list of all currently defined SAP tasks.

Below the list, SAP Task Details for a new SAP task displays. (You also can click the **New** button to display SAP Task Details for a new SAP task.)

The screenshot shows the SAP Tasks interface. At the top, there is a header for 'SAP Tasks' with a dropdown menu showing '5 SAP Tasks' and a 'Custom Filter -- None --' option. Below this is a table listing tasks:

Task Name	Task Description	Updated By	Updated
stonebranch-saptask-01		ops.admin	2014-06-13 14:04:18 -0400
stonebranch-saptask-02		ops.admin	2014-06-13 14:04:32 -0400
stonebranch-saptask-03		ops.admin	2014-06-13 14:04:42 -0400
stonebranch-saptask-04		ops.admin	2014-06-13 14:04:50 -0400
stonebranch-saptask-05		ops.admin	2014-06-13 14:04:59 -0400

Below the table is the 'SAP Task Details' form. It has tabs for 'SAP Task', 'Variables', 'Actions', 'Virtual Resources', 'Mutually Exclusive', 'Instances', 'Triggers', 'Notes', and 'Versions'. The 'SAP Task' tab is active, showing the following fields:

General

Task Name: Version:

Task Description:

Member of Business Services:

Hold on Start:

Virtual Resource Priority: Hold Resources on Failure:

SAP Details

Utility Agent: Utility Agent Cluster:

Agent Variable: Agent Cluster Variable:

Credentials: Credentials Variable:

Step 2 Enter/select Details for a new SAP task, using the field descriptions below as a guide.

- Required fields display in **boldface**.
- Default values for fields, if available, display automatically.

To display more of the Details fields on the screen, you can temporarily [hide the list](#).



Note

If you view [SAP Task Details](#) for an existing SAP task by clicking a task in the list, and then want to create a new SAP task, you must click the **New** button that displays above and below the Details.

Step 3 Click the **Save** button. The task is added to the database, and all buttons and tabs in the Task Details are enabled.

SAP Task Details

The following SAP Task Details is for an existing SAP task.

Depending on the values that you enter / select for these fields, and whether or not the SAP task has ever been launched, more (or less) fields may display. See the [field descriptions](#), below, for a description of all fields that may display in the SAP Task Details.

SAP Task Details: stonebranch-saptask-01

Update Launch Task View Parents Copy Delete Refresh Close

SAP Task Variables Actions Virtual Resources Mutually Exclusive Instances Triggers Notes Versions

General

Task Name: stonebranch-saptask-01 Version: 1

Task Description:

Member of Business Services:

Hold on Start:

Virtual Resource Priority: 10 Hold Resources on Failure:

SAP Details

Utility Agent: qa-agentsim1 - simagent1 Utility Agent Cluster:

Utility Agent Variable: Cluster Variable:

Utility Credentials: Utility Credentials Variable:

SAP Connection: SAP Credentials: QATEST SAP Credentials Variable:

SAP Language:

Command Group: Run

Definition of Model: USAP Definition File

Script or File System: Script

Script: SAP DEF BTCLOOP (QAOPS-BTCLOOP)

Start Immediately:

SAP Target Server:

Print Application Log:

Print Application RC:

Use Application RC:

SAP Command Options:

Runtime Directory:

Environment Variables:

Name	Value
No items to show.	

Exit Code Processing: Success Exitcode Range

Exit Codes: 0

Automatic Output Retrieval: Standard Output/Error

Start Line: 1 Number of Lines: 100

Scan Text:

Retry Options

Maximum Retries: 0 Retry Indefinitely:

Retry Interval (Seconds): 60

Time Options

Late Start:

Late Finish:

Early Finish:


User Estimated Duration: HH:MM:SS




Update Launch Task View Parents Copy Delete Refresh Close

SAP Task Details Field Descriptions

The following table describes the fields, buttons, and tabs that display in the SAP Task Details.

Field Name	Description
General	This section contains general information about the task.


Task Name	User-defined name of this task, containing a maximum 255 alphanumeric characters; variables supported. It is the responsibility of the user to develop a workable naming scheme for tasks.
Version	System-supplied; version number of the current record, which is incremented by the Controller every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
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.
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.
SAP Details	This section contains assorted detailed information about the task.
Utility Agent	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 .
Utility Agent Variable	If enabled, the Utility 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. Use the format: <code>\$(variable name)</code> . The variable must be a supported type as described in Variables and Functions . <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Utility Agent Variable when updating multiple tasks, the Utility Agent field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Utility Agent Unresolved - into which you enter a variable, variable and text, or Utility Agent name. The Utility Agent field is not used.</p> </div>
Utility Agent Cluster	Group of Agents, one of which the Controller will choose to run this task. You can specify an agent cluster in addition to or in place of a specific Agent. If you specify an Agent and an agent cluster, the Controller first tries to run the task on the specific Agent. If the Agent is not available, the Controller reverts to the agent cluster. See Agent Clusters for more information.

Cluster Variable	<p>If enabled, the Utility 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. Use the format: <code>\${variable name}</code>. The variable must be a supported type as described in Variables and Functions.</p> <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note If you select Utility Agent Cluster Variable when updating multiple tasks, the Utility Agent Cluster field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Utility Agent Cluster Unresolved - into which you enter a variable, variable and text, or Utility Agent Cluster name. The Utility Agent Cluster field is not used.</p> </div>
Utility 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>
Utility Credentials Variable	<p>If enabled, the Credentials field 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> <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note If you select Credentials Variable when updating multiple tasks, the Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Credentials Unresolved - into which you enter a variable, variable and text, or Credentials name. The Credentials field is not used.</p> </div>
SAP Connection	<p>Name of the SAP connection. The SAP connection specifies information about the SAP server. Select an existing SAP Connection from the drop-down list or click the icon to create a new SAP Connection.</p>
SAP Language	<p>SAP logon language used when executing the SAP task. Valid values are:</p> <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	<p>Login credentials that the Controller will use to access the SAP system. The credentials are stored in the Opwise Controller credentials table; see Credentials.</p>
SAP Credentials Variable	<p>If enabled, the Credentials field 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> <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note If you select Credentials Variable when updating multiple tasks, the Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Credentials Unresolved - into which you enter a variable, variable and text, or Credentials name. The Credentials field is not used.</p> </div>
Command Group	<p>See Universal Connector Commands, below, for a description of all supported commands and their contingent fields (options).</p>
SAP Command Options	<p>Use this field to specify any additional command options supported by Universal Connector (USAP) .</p>

Runtime Directory	Directory from which the application should be executed. Variables supported.
Environment Variables	<p>Allows you to enter environment variables needed by the program to run.</p> <p>To add a variable, click the + icon and enter a Name and Value. To delete a variable, select in the list of variables and click the - icon.</p> <p>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.</p>
Exit Code Processing	<p>Specifies how the Controller should determine whether the executed command failed or completed successfully.</p> <p>Options:</p> <ul style="list-style-type: none"> • Success Exitcode Range - Command is considered completed successfully if its exit code falls within the range specified in the Exit Codes field. • Failure Exitcode Range - Command is considered failed if its exit code falls within the range specified in the Exit Codes field. • Success Output Contains - Command is considered completed successfully if its output contains the text specified in the Scan Output For field. • Failure Output Contains - Command is considered failed if its output contains the text specified in the Scan Output For field. • Step Conditions (z/OS only) - 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	<p>Required if Exit Code Processing = Success Output Contains or Failure Output Contains; type of output.</p> <p>Options:</p> <ul style="list-style-type: none"> • Standard Output (STDOUT) • Standard Error (STDERR) • File
Scan Output For	<p>Required if Exit Code Processing = Success Output Contains or Failure Output Contains; text for which the Controller should scan the output file. The Controller will process this field as a regular expression.</p>
Output File (for Exit Code Processing)	<p>Required if Output Type = File; path and file name of the output file that should be scanned for the text in the Scan Output For field.</p>
Exit Codes	<p>Required if Exit Code Processing = Success Exitcode Range or Failure Exitcode Range; range of exit codes. Format: Numeric. Use commas to list a series of exit codes; use hyphens to specify a range. Example: 1,5, 22-30.</p>
Automatic Output Retrieval	<p>Specifies whether you want the Controller to automatically retrieve any output from the job and attach it to the task instance record.</p> <p>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. • Standard Output/Error - Attach all standard output and standard error output.
Start Line	<p>If Automatic Output Retrieval = Standard Output, Standard Error, File, or Standard Output/Error; Instructs the Controller to retrieve data beginning at the line indicated.</p> <ul style="list-style-type: none"> • If a Start Line value is not specified, the default is 1. • If the Start Line value is -1, data will be retrieved starting at the end of the file.

Number of Lines	If Automatic Output Retrieval = Standard Output, Standard Error, File, or Standard Output/Error; 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 Controller system property.
Scan Text	If Automatic Output Retrieval = Standard Output, Standard Error, File, or Standard Output/Error; Regex pattern that the Controller will search for a match for in STDOUT/STDERR or a specified file. The Controller will include the Number of Lines above and below the first line matched. if the Regex pattern is not found, the following message is returned: OPSWISE WARNING - Scan text string not found.
Output File (for Automatic Output Retrieval)	Required if Automatic Output Retrieval = File; path and file name containing the output that you want automatically retrieved and attached to the task instance.
Retry Options	This section contains specifications for retrying the task.
Maximum Retries	User-defined; maximum number of times that the Controller should retry this task after it has started and gone to a failed state.
Retry Indefinitely	User-defined; indicates whether the Controller should continue trying indefinitely to run this task. If you enable this field, it overrides any value placed in the Maximum Retries field.
Retry Interval (Seconds)	User-defined; number of seconds between each retry.
Time Options	This section contains time-related specifications for the task.
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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.
Late Start Type	Required if Late Start is enabled. Options: <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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.
Late Start Duration	If Late Start Type = 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.

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: <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 = Time; Time after which the task finish time is considered late. Use HH:MM, 24-hour time.
Late Finish Duration	If Late Finish Type = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
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: <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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
User Estimated Duration	Estimated amount of time it should normally take to run this task. The Controller uses this information to calculate the User Estimated End Time on a task instance record.
Statistics	This section contains time-related statistics for task instances of the task.
First Time Ran	System-supplied; date and time this task first ran.
Last Time Ran	System-supplied; date and time the task last ran.
Last Instance Duration	System-supplied; amount of time the task took to run the last time it ran.

Lowest Instance Time	System-supplied; shortest amount of time this task has taken to run.
Average Instance Time	System-supplied; average amount of time this task takes to run.
Highest Instance Time	System-supplied; longest amount of time this task has taken to run.
Number of Instances	System-supplied; number of instances in the database for this task.
Buttons	This section identifies the buttons displayed above and below the Task Details that let you perform various actions.
Save	Saves a new task record in the Controller database.
Update	Saves updates to the record.
New	Displays empty (except for default values) Details for creating a new task.
Launch Task	Manually launches the task.
View Parents	Displays a list of any parent Workflow tasks for this task.
Copy	Creates a copy of this task, which you are prompted to rename.
Delete	<p>Deletes the current record.</p> <div style="background-color: #ffffcc; padding: 10px; margin: 10px 0;"> <p> Note You cannot delete a task if it is either:</p> <ul style="list-style-type: none"> Specified in an enabled Trigger. The only task specified in a disabled Trigger. </div>
Refresh	Refreshes any dynamic data displayed in the Details.
Close	For pop-up view only; closes the pop-up view of this task.
Tabs	This section identifies the tabs across the top of the Task Details that provide access to additional information about the task instance.
Variables	Lists all variables associated with this record.

Actions	<p>Allows you to specify actions that the Controller will take automatically based on events that occur during the execution of this task.</p> <p>Events are:</p> <ul style="list-style-type: none"> • Task instance status • Exit codes • Late start • Late finish • Early finish <p>Actions are:</p> <table border="1" data-bbox="370 478 1481 840"> <tr> <td data-bbox="370 478 509 548">Abort Action</td> <td data-bbox="509 478 1481 548">Abort the task if certain events occur. For details, see Abort Actions.</td> </tr> <tr> <td data-bbox="370 548 509 617">Email Notification</td> <td data-bbox="509 548 1481 617">Send an email if certain events occur. For details, see Email Notification Actions.</td> </tr> <tr> <td data-bbox="370 617 509 686">Set Variable</td> <td data-bbox="509 617 1481 686">Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow.</td> </tr> <tr> <td data-bbox="370 686 509 756">SNMP Notification</td> <td data-bbox="509 686 1481 756">Send an email if certain events occur. For details, see SNMP Notification Actions.</td> </tr> <tr> <td data-bbox="370 756 509 825">System Operation</td> <td data-bbox="509 756 1481 825">Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions.</td> </tr> </table>	Abort Action	Abort the task if certain events occur. For details, see Abort Actions .	Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .	Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .	SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .	System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .
Abort Action	Abort the task if certain events occur. For details, see Abort Actions .										
Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .										
Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .										
SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .										
System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .										
Virtual Resources	Lists all Virtual Resources to which this task is assigned.										
Mutually Exclusive	Lists all tasks that have been set to be mutually exclusive of this task.										
Instances	Lists all instances of the task.										
Triggers	Lists 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, the Controller 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 Triggers Overview .										
Notes	Lists all notes associated with this record.										
Versions	Stores copies of all previous versions of the current record. See Record Versioning .										

Viewing an SAP Task Instance

When an SAP task is launched, the Controller creates a task instance record of that task.

A task instance contains detailed information about a single execution of that task.

You can access a task instance from:

- **Instances tab** on the [SAP Task Details](#) for that task
- [Activity Monitor](#)
- [Task Instances list](#)

SAP Task Instance Details

The following SAP Task Instance Details contains information on the execution of the task shown in the [SAP Task Details](#).

SAP Task Instance Details: stonebranch-saptask-01
Update Re-run View Parent Retrieve Output... SAP Delete Refresh Close

SAP Task Instance
Virtual Resources Exclusive Requests Output Notes

General

Instance Name: Reference Id:

Task: Invoked By:

Task Description:

Member of Business Services: Execution User:

Virtual Resource Priority: Hold Resources on Failure:

Status

Status: Exit Code:

Status Description:

Queued Time:

Trigger Time: Start Time:

Launch Time: End Time:

Duration:

SAP Details

Utility Agent: Utility Agent Cluster:

Utility Agent Variable: Cluster Variable:

Utility Credentials: Utility Credentials Variable:

SAP Connection: SAP Credentials:

SAP Language: SAP Credentials Variable:

SAP Job Name: SAP Job ID:

SAP Process Chain Log ID:

SAP InfoPackage Request ID:

Command Group: Display Command:

SAP Job Name:

SAP Job ID:

From Date:

To Date:

SAP Command Options:

Runtime Directory:

Name	Value
No items to show.	

Exit Code Processing: Exit Codes:

Automatic Output Retrieval: Start Line: Number of Lines:

Scan Text:

Retry Options

Maximum Retries: Retry Indefinitely:

Retry Interval (Seconds): Current Retry Count:

Statistics

User Estimated End Time: Average Estimated End Time:



Shortest Estimated End Time: Longest Estimated End Time:



Update Re-run View Parent Retrieve Output... SAP Delete Refresh Close

SAP Task Instance Details Field Descriptions

The following table describes the fields, buttons, and tabs that display in SAP Task Instance Details.

Field Name	Description
General	This section contains general information about the task instance.
Instance Name	Name of this task instance.
Reference Id	System-supplied; the Controller increments this number each time the task is run.
Task	Name of the task that was run to create this task instance. Click the icon to display Task Details for the task.
Invoked by	System-supplied; how the task instance was launched. Options: <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 identify the user, check the Execution User column for that task instance on the Task Instances screen or, on most task instance screens, the Execution User field.
Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
Execution User	System-supplied; if the task was launched manually, the ID of the user who launched it.
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.
Status	This section contains information about the current status of the task instance.
Status	System-supplied; see Task Instance Statuses .
Exit Code	System-supplied; the exit code captured by the Agent when executing the task (for example, a command or script).
Status Description	System-supplied; additional information, if any, about the status of the task instance.
Queued Time	System-supplied; Date and time the task was queued for processing.

Trigger Time	System-supplied; Date and time the task instance was triggered.
Launch Time	System-supplied; Date and time the task instance was launched.
Start Time	System-supplied; Date and time the task instance started.
End Time	System-supplied; Date and time the task instance completed.
Duration	System-supplied; amount of time the task instance took to run.
SAP Details	This section contains assorted detailed information about the task instance.
Utility Agent	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 .
Utility Agent Variable	<p>If enabled, the Utility 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. Use the format: \${variable name}. The variable must be a supported type as described in Variables and Functions.</p> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Utility Agent Variable when updating multiple tasks, the Utility Agent field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Utility Agent Unresolved - into which you enter a variable, variable and text, or Utility Agent name. The Utility Agent field is not used.</p> </div>
Utility Agent Cluster	Group of Agents, one of which the Controller will choose to run this task. You can specify an agent cluster in addition to or in place of a specific Agent. If you specify an Agent and an agent cluster, the Controller first tries to run the task on the specific Agent. If the Agent is not available, the Controller reverts to the agent cluster. See Agent Clusters for more information.
Cluster Variable	<p>If enabled, the Utility 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. Use the format: \${variable name}. The variable must be a supported type as described in Variables and Functions.</p> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Utility Agent Cluster Variable when updating multiple tasks, the Utility Agent Cluster field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Utility Agent Cluster Unresolved - into which you enter a variable, variable and text, or Utility Agent Cluster name. The Utility Agent Cluster field is not used.</p> </div>
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	<p>If enabled, the Credentials field 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> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Credentials Variable when updating multiple tasks, the Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Credentials Unresolved - into which you enter a variable, variable and text, or Credentials name. The Credentials field is not used.</p> </div>
SAP Connection	Name of the SAP connection . The SAP connection specifies information about the SAP server. Select an existing SAP Connection from the drop-down list or click the icon to create a new SAP Connection.
SAP Language	<p>SAP logon language used when executing the SAP task. Valid values are:</p> <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	Login credentials that the Controller will use to access the SAP system. The credentials are stored in the Opwise Controller credentials table; see Credentials .
SAP Credentials Variable	<p>If enabled, the Credentials field 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> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Credentials Variable when updating multiple tasks, the Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Credentials Unresolved - into which you enter a variable, variable and text, or Credentials name. The Credentials field is not used.</p> </div>
SAP Job Name	Job name of the SAP job. Variables supported.
SAP Job ID	Job ID of the SAP job. Variables supported.
SAP Process Chain Log ID	Log ID for process chain instance to be monitored to completion.
SAP InfoPackage Request ID	Request ID of the InfoPackage that is to be monitored.
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	Directory from which the application should be executed. Variables supported.

<p>(Environment Variables)</p> <p>Name and Value</p>	<p>Allows you to enter environment variables needed by the program to run.</p> <p>To add a variable, click the + icon and enter a Name and Value. To delete a variable, select in the list of variables and click the - icon.</p> <p>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.</p>
<p>Exit Code Processing</p>	<p>Specifies how the Controller should determine whether the executed command failed or completed successfully.</p> <p>Options:</p> <ul style="list-style-type: none"> • Success Exitcode Range - Command is considered completed successfully if its exit code falls within the range specified in the Exit Codes field. • Failure Exitcode Range - Command is considered failed if its exit code falls within the range specified in the Exit Codes field. • Success Output Contains - Command is considered completed successfully if its output contains the text specified in the Scan Output For field. • Failure Output Contains - Command is considered failed if its output contains the text specified in the Scan Output For field. • Step Conditions (z/OS only) - 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).
<p>Output Type</p>	<p>Required if Exit Code Processing = Success Output Contains or Failure Output Contains; type of output.</p> <p>Options:</p> <ul style="list-style-type: none"> • Standard Output (STDOUT) • Standard Error (STDERR) • File
<p>Scan Output For</p>	<p>Required if Exit Code Processing = Success Output Contains or Failure Output Contains; text for which the Controller should scan the output file. The Controller will process this field as a regular expression.</p>
<p>Output File (for Exit Code Processing)</p>	<p>Required if Output Type = File; path and file name of the output file that should be scanned for the text in the Scan Output For field.</p>
<p>Exit Codes</p>	<p>Required if Exit Code Processing = Success Exitcode Range or Failure Exitcode Range; range of exit codes. Format: Numeric. Use commas to list a series of exit codes; use hyphens to specify a range. Example: 1,5, 22-30.</p>
<p>Automatic Output Retrieval</p>	<p>Specifies whether you want the Controller to automatically retrieve any output from the job and attach it to the task instance record.</p> <p>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. • Standard Output/Error - Attach all standard output and standard error output.
<p>Start Line</p>	<p>If Automatic Output Retrieval = Standard Output, Standard Error, File, or Standard Output/Error; Instructs the Controller to retrieve data beginning at the line indicated.</p> <ul style="list-style-type: none"> • If a Start Line value is not specified, the default is 1. • If the Start Line value is -1, data will be retrieved starting at the end of the file.

Number of Lines	If Automatic Output Retrieval = Standard Output, Standard Error, File, or Standard Output/Error; 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 Controller system property.
Scan Text	If Automatic Output Retrieval = Standard Output, Standard Error, File, or Standard Output/Error; Regex pattern that the Controller will search for a match for in STDOUT/STDERR or a specified file. The Controller will include the Number of Lines above and below the first line matched. if the Regex pattern is not found, the following message is returned: OPSWISE WARNING - Scan text string not found.
Output File (for Automatic Output Retrieval)	Required if Automatic Output Retrieval = File; path and file name containing the output that you want automatically retrieved and attached to the task instance.
Retry Options	This section contains specifications for retrying the task.
Maximum Retries	User-defined; maximum number of times that the Controller should retry this task after it has started and gone to a failed state.
Retry Indefinitely	User-defined; indicates whether the Controller should continue trying indefinitely to run this task. If you enable this field, it overrides any value placed in the Maximum Retries field.
Retry Interval (Seconds)	User-defined; number of seconds between each retry.
Current Retry Count	System-supplied; current number of times that the Controller has retried the task after it first went to failure status.
Time Options	This section contains time-related specifications for the task instance.
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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.
Started Late	System-supplied; this field is flagged if the task started later than the time specified in the Late Start fields.
Late Start Type	Required if Late Start is enabled. Options: <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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.

Late Start Duration	<p>If Late Start Type = 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.</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>
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>
Finished Late	<p>System-supplied; this field is flagged if the task finished later than the time or duration specified in the Late Finish fields.</p>
Late Finish Type	<p>Required if Late Finish is enabled.</p> <p>Options:</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 = Time; 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 = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.</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>
Finished Early	<p>System-supplied; this field is flagged if the task finished earlier than the time specified in the Early Finish fields.</p>
Early Finish Type	<p>Required if Early Finish is enabled.</p> <p>Options:</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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.</p>
Statistics	<p>This section contains time-related statistics for the task instance.</p>

User Estimated End Time	System-supplied; if the user entered information into the User Estimated Duration field in the task Details, the Controller uses this information to calculate an end time for the task instance, based on the date/time the task instance started.
Average Estimated End Time	System-supplied; average estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Shortest Estimated End Time	System-supplied; shortest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Longest Estimated End Time	System-supplied; longest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Buttons	This section identifies the buttons displayed above and below the Task Instance Details that let you perform various actions.
Update	Saves updates to the record.
Force Finish	See Force Finishing a Task .
Hold	Places the task instance on Hold (see Putting a Task on Hold).
Skip	For tasks loaded into the schedule that have not yet run; allows you to tell the Controller to skip this task. See Skipping a Task .
Re-run	See Re-running a Task .
View Parent	Displays the task instance Details for the parent Workflow of this task instance.
Retrieve Output	See Retrieving Output .
SAP	Displays an Action menu of SAP commands.
Delete	Deletes the current record.
Refresh	Refreshes any dynamic data displayed in the Details.
Close	For pop-up view only; closes the pop-up view of this task instance.
Tabs	This section identifies the tabs across the top of the Task Instance Details that provide access to additional information about the task instance.
Virtual Resources	Lists all Virtual Resources to which this task is assigned.
Exclusive Requests	Lists all records in the Exclusive Requests table (<code>ops_exclusive_order</code>) for this task instance.

Output	<p>Displays output generated from the process, if any, based on specifications provided by the user in the Automatic Output Retrieval fields in the task Details.</p> <p>If automatic output retrieval was not available or was not selected, output can be obtained by clicking the Retrieve Output button.</p>
Notes	Lists all notes associated with this record.

Output Redirection

An Agent 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"
```

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 in the SAP Task Details when that command is selected.

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

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 or File System**

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

- **Script**

If [Command](#) or [Script](#) = Script; Required. Name of the [script](#) in the Controller database that 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, the Controller 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.

- **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 Controller task.

<p>Run Process Chain</p>	<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.
<p>Run InfoPackage</p>	<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.
<p>Submit</p>	<p>Defines a new SAP job.</p>	<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 Scripts. • Script If Command or Script = Script; Required. Name of the script in the Controller database that 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, the Controller 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 the Controller 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 Controller 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 Scripts.• Script If Command or Script = Script; Required. Name of the script in the Controller database that 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: <pre>[host name]_[SAP System name]_[SAP System number]</pre> 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: <pre>hs0123_C11_55</pre>• Wait Specifies whether the Controller 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 Controller 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 restarted. • Wait Specifies whether the Controller 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 Controller 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 the Controller 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. <i>Variables</i> supported. • SAP Job ID Job ID of the SAP job. <i>Variables</i> 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 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 Controller 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. <i>Variables</i> supported. • SAP Job ID Job ID of the SAP job. <i>Variables</i> 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 spoollists. 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. <i>Variables</i> supported. • SAP Job ID Job ID of the SAP job. <i>Variables</i> 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. 																														
Display	Displays the data specified in the Display Command field. The data is written to standard output.	<ul style="list-style-type: none"> • Display Command One of the following: <table border="1" data-bbox="922 737 1479 1892"> <tr> <td data-bbox="927 743 1068 800">Job Log</td> <td data-bbox="1073 743 1474 800">Displays the job log for a specified SAP job.</td> </tr> <tr> <td data-bbox="927 806 1068 842">Spool List</td> <td data-bbox="1073 806 1474 842">Displays the spoolist for a job step.</td> </tr> <tr> <td data-bbox="927 848 1068 905">Status</td> <td data-bbox="1073 848 1474 905">Displays the current status for an SAP job.</td> </tr> <tr> <td data-bbox="927 911 1068 968">Variants</td> <td data-bbox="1073 911 1474 968">Displays the variants available for the specified ABAP program.</td> </tr> <tr> <td data-bbox="927 974 1068 1073">Variant</td> <td data-bbox="1073 974 1474 1073">Displays the contents of a specified variant. Note: Requires XBP interface 2.0 or greater.</td> </tr> <tr> <td data-bbox="927 1079 1068 1136">Job Definition</td> <td data-bbox="1073 1079 1474 1136">Displays the definition of the specified SAP job.</td> </tr> <tr> <td data-bbox="927 1142 1068 1241">Select</td> <td data-bbox="1073 1142 1474 1241">Displays a variety of attributes for a list of SAP jobs that match the specified criteria.</td> </tr> <tr> <td data-bbox="927 1247 1068 1346">System Log</td> <td data-bbox="1073 1247 1474 1346">Displays a portion of an SAP syslog that meets the specified date/time constraints.</td> </tr> <tr> <td data-bbox="927 1352 1068 1430">Intercept Table</td> <td data-bbox="1073 1352 1474 1430">Displays the contents of the job intercept criteria table for the connected SAP system.</td> </tr> <tr> <td data-bbox="927 1436 1068 1493">Intercepted Jobs</td> <td data-bbox="1073 1436 1474 1493">Displays intercepted jobs for the connected SAP system.</td> </tr> <tr> <td data-bbox="927 1499 1068 1556">Reports</td> <td data-bbox="1073 1499 1474 1556">Displays a list of ABAP reports that match the specified criteria.</td> </tr> <tr> <td data-bbox="927 1562 1068 1661">Commands</td> <td data-bbox="1073 1562 1474 1661">Displays a list of SAP external commands that match the specified criteria.</td> </tr> <tr> <td data-bbox="927 1667 1068 1724">Output Devices</td> <td data-bbox="1073 1667 1474 1724">Displays a list of SAP output devices that match the specified criteria.</td> </tr> <tr> <td data-bbox="927 1730 1068 1787">Print Formats</td> <td data-bbox="1073 1730 1474 1787">Displays a list of print formats that are available for the specified printer.</td> </tr> <tr> <td data-bbox="927 1793 1068 1850">Selection Screen</td> <td data-bbox="1073 1793 1474 1850">Displays information about the selection fields of an ABAP program.</td> </tr> </table>	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.
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.																															

		<table border="1"> <tr> <td data-bbox="922 128 1068 243">Event History</td> <td data-bbox="1068 128 1487 243">Displays a list of events that were logged in an SAP system's event history. The retrieved events can optionally be set to "Confirmed."</td> </tr> <tr> <td data-bbox="922 243 1068 338">Criteria Manager Profiles</td> <td data-bbox="1068 243 1487 338">Displays a list of Criteria Manager profiles.</td> </tr> <tr> <td data-bbox="922 338 1068 432">Criteria Manager Criteria</td> <td data-bbox="1068 338 1487 432">Displays the criteria hierarchy of a particular profile in XML format.</td> </tr> <tr> <td data-bbox="922 432 1068 527">Process Chains</td> <td data-bbox="1068 432 1487 527">Displays a list of process chains from the SAP system that meet the specified criteria.</td> </tr> <tr> <td data-bbox="922 527 1068 600">Process Chain</td> <td data-bbox="1068 527 1487 600">Displays the list of processes contained within the specified process chain.</td> </tr> <tr> <td data-bbox="922 600 1068 674">Process Chain Log</td> <td data-bbox="1068 600 1487 674">Displays the SAP log associated with the process chain.</td> </tr> <tr> <td data-bbox="922 674 1068 768">Process Chain Start Condition</td> <td data-bbox="1068 674 1487 768">Displays the SAP start condition for specified process chain.</td> </tr> <tr> <td data-bbox="922 768 1068 842">Process Chain Status</td> <td data-bbox="1068 768 1487 842">Displays the current status of the process chain.</td> </tr> <tr> <td data-bbox="922 842 1068 936">InfoPackages</td> <td data-bbox="1068 842 1487 936">Displays a list of InfoPackages on the SAP system that meet the specified criteria.</td> </tr> <tr> <td data-bbox="922 936 1068 1031">InfoPackage Status</td> <td data-bbox="1068 936 1487 1031">Displays the current status for the InfoPackage instance identified by the request ID.</td> </tr> </table>	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.
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 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. 																				
<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. <i>Variables</i> supported. SAP Job ID Job ID of the SAP job. <i>Variables</i> supported. 																				

Create CM Profile	Creates a new Criteria Manager profile.	<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 Scripts.• Script If Command or Script = Script; Required. Name of the script in the Controller database that 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 Scripts. • Script If Command or Script = Script; Required. Name of the script in the Controller database that 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](#)

Running an SAP Task

You can run an SAP task:

- Manually, by clicking the [Launch Task](#) or [Launch Task with Variables](#) button in the SAP tasks list or SAP Task Details [Action menu](#).
- As part of a [workflow](#).
- [Specify triggers](#) that run the task automatically based on times or events.

Monitoring Task Execution

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

File Transfer Task

- Overview
- Creating a File Transfer Task
 - File Transfer Types
 - FTP/SFTP File Transfer Task Details
 - FTP/SFTP File Transfer Task Details Field Descriptions
 - INFITRAN File Transfer Task Details
 - INFITRAN File Transfer Task Details Field Descriptions
- Viewing a File Transfer Task Instance
 - FTP/SFTP File Transfer Task Instance Details
 - FTP/SFTP File Transfer Task Instance Details Field Descriptions
 - INFITRAN File Transfer Task Instance Details
 - INFITRAN File Transfer Task Instance Details Field Descriptions
- Output Redirection
 - -uagstdio Examples
- Specifying When a Task Runs
- Running a File Transfer Task
- 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 Opwise Universal Agent for Linux/Unix, z/OS, or Windows 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 examples provide sample configurations for executing file transfers using a File Transfer task.

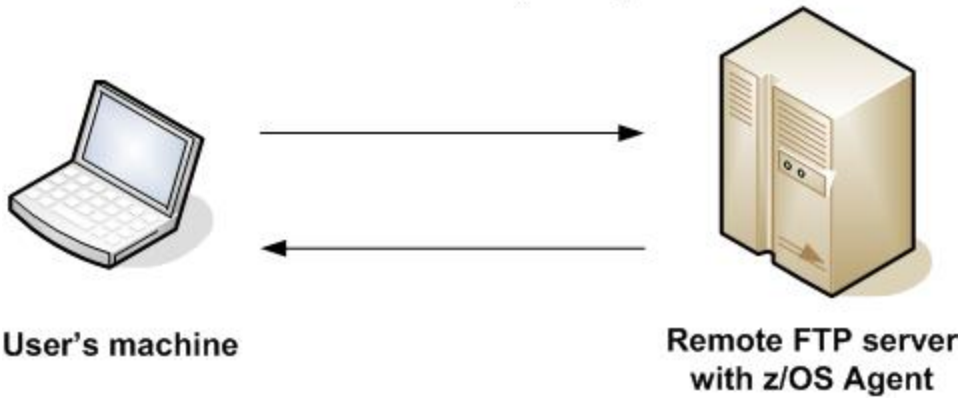
In the first example, the user wants to transfer a file from a remote File Transfer Server on a machine that does not have an Agent running on it. In this case, the File Transfer task definition provides an address and login credentials for the machine where the Agent is running as well as address and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

Example 1:
User wants to transfer a file from a remote machine that is not running an Agent.



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

Example 2:
User wants to transfer a file from a remote machine that is running an Agent.




Creating a File Transfer Task

Step 1 From the [Automation Center](#) navigation pane, select **Tasks > File Transfer Tasks**. The File Transfer Tasks list displays a list of all currently defined File Transfer tasks.

Below the list, File Transfer Task Details for a new File Transfer task displays. (You also can click the **New** button to display File Transfer Task Details for a new File Transfer task.)

The screenshot shows the 'File Transfer Tasks' management interface. At the top, there is a list of 5 tasks with columns for Task Name, Task Description, Transfer Type, Updated By, and Updated. Below the list is a 'File Transfer Task Details' form with several tabs: 'File Transfer Task', 'Variables', 'Actions', 'Virtual Resources', 'Mutually Exclusive', 'Instances', 'Triggers', 'Notes', and 'Versions'. The 'General' tab is active, showing fields for Task Name, Version (1), Task Description, Member of Business Services, Hold on Start, Virtual Resource Priority (10), and Hold Resources on Failure. The 'File Transfer Details' tab is also visible, showing fields for Transfer Type (FTP), Command (GET), Job Card (z/OS only), Agent, Agent Variable, Subcommands (z/OS only), Transfer Mode (Binary), and Transfer Method (Passive).

Task Name	Task Description	Transfer Type	Updated By	Updated
stonebranch-filetransfertask-01		FTP	ops.admin	2014-06-13 14:09:39 -0400
stonebranch-filetransfertask-02		INFINTRAN	ops.admin	2014-06-26 16:31:16 -0400
stonebranch-filetransfertask-03		FTP	ops.admin	2014-06-13 14:10:03 -0400
stonebranch-filetransfertask-04		FTP	ops.admin	2014-06-13 14:10:09 -0400
stonebranch-filetransfertask-05		FTP	ops.admin	2014-06-13 14:10:15 -0400

Step 2	<p>Enter/select Details for a new File Transfer task, using the field descriptions below as a guide.</p> <ul style="list-style-type: none"> • Required fields display in boldface. • Default values for fields, if available, display automatically. <p>To display more of the Details fields on the screen, you can temporarily hide the list.</p> <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note If you view File Transfer Task Details for an existing File Transfer task by clicking a task in the list, and then want to create a new File Transfer task, you must click the New button that displays above and below the Details.</p> </div>
Step 3	Click the Save button. The task is added to the database, and all buttons and tabs in the Task Details are enabled.

File Transfer Types

When you create a File Transfer task, you select a file transfer type: FTP, SFTP, or INFITRAN.

The Details for [FTP and SFTP transfer types](#) are the same; the Details for the [INFITRAN transfer types](#) differs considerably.

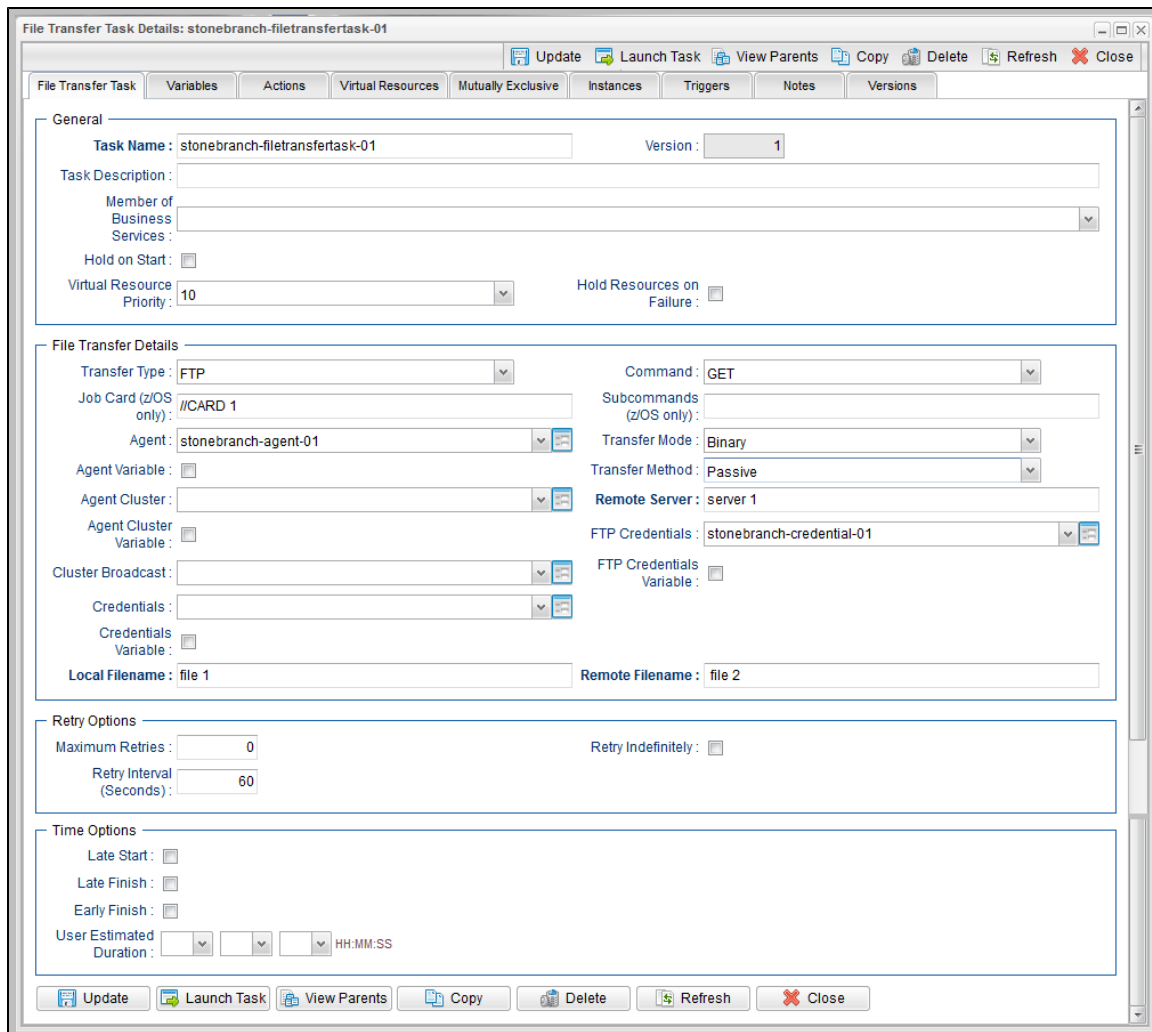
Using **SFTP** requires that you supply a valid [credential](#) that specifies the location of the SSL Private key on your Agent. In the Credentials Details, you supply the location for the private key in the field "Key Location (File Transfer only)". This location must exist on the Agent where you intend to run the SFTP task.

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

FTP/SFTP File Transfer Task Details

The following FTP/SFTP File Transfer Task Details is for an existing FTP/SFTP File Transfer task.

Depending on the values that you enter / select for these fields, and whether or not the FTP/SFTP File Transfer task has ever been launched, more (or less) fields may display. See the [field descriptions](#), below, for a description of all fields that may display in the FTP/SFTP File Transfer Task Details.







FTP/SFTP File Transfer Task Details Field Descriptions

The following table describes the fields, buttons, and tabs that display in FTP/SFTP File Transfer Task Details.


Field Name	Description
General	This section contains general information about the task.
Task Name	User-defined name of this task, containing a maximum 255 alphanumeric characters; variables supported. It is the responsibility of the user to develop a workable naming scheme for tasks.
Version	System-supplied; version number of the current record, which is incremented by the Controller every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
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.
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.
File Transfer Details	This section contains assorted detailed information about the task.
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.
Job Card (z/OS only)	For z/OS, the job card information for the JCL statement. Example: <pre>//File TransferJOB01 JOB (File Transfer,001),FANNY,MSGCLASS=X,MSGLEVEL=(1,1),NOTIFY=&SYSUID,CLASS=A</pre>
Subcommands (z/OS only)	For z/OS, any subcommands used in the JCL statement.
Agent	Name of the Agent resource that identifies the machine where the operation will run. If you do not specify an Agent, you must specify an Agent Cluster .

Agent Variable	<p>If enabled, 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. Use the format: <code>\${variable name}</code>. The variable must be a supported type as described in Variables and Functions.</p> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Agent Variable when updating multiple tasks, the Agent field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Agent Unresolved - into which you enter a variable, variable and text, or Agent name. The Agent field is not used.</p> </div>
Transfer Mode	<p>Mode of data transfer.</p> <p>Options:</p> <ul style="list-style-type: none"> • Binary • ASCII
Transfer Method	<p>Transfer method.</p> <p>Options:</p> <ul style="list-style-type: none"> • Active • Passive • Extended Passive
Agent Cluster	<p>Group of Agents, one of which the Controller will choose to run this task. You can specify an agent cluster in addition to or in place of a specific Agent. If you specify an Agent and an agent cluster, the Controller first tries to run the task on the specific agent. If the Agent is not available, the Controller reverts to the agent cluster. See Agent Clusters for more information.</p>
Agent Cluster Variable	<p>If enabled, 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. Use the format: <code>\${variable name}</code>. The variable must be a supported type as described in Variables and Functions.</p> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Agent Cluster Variable when updating multiple tasks, the Agent Cluster field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Agent Cluster Unresolved - into which you enter a variable, variable and text, or Agent Cluster name. The Agent Cluster field is not used.</p> </div>
Cluster Broadcast	<p>Group of Agents, all of which will run this task. You can specify a Cluster Broadcast in place of a specific Agent and/or agent cluster. Each instance of the task running on its own Agent becomes a separate task instance record in the database and displays separately on the Activity screen. See Agent Clusters for more information about defining agent clusters.</p>
Remote Server	<p>Required if Transfer Type = FTP or SFTP; name or IP address of the File Transfer server. This machine may or may not be the same as the 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 that the Agent will use to access the FTP or SFTP server machine. If the File Transfer server and Agent are running on the same machine, enter the same credentials as those you entered in the Credentials field.</p>

FTP Credentials Variable	<p>If enabled, the FTP Credentials field 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> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select FTP Credentials Variable when updating multiple tasks, the FTP Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - FTP Credentials Unresolved - into which you enter a variable, variable and text, or FTP Credentials name. The FTP Credentials field is not used.</p> </div>
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>
Credentials Variable	<p>If enabled, the Credentials field 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> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Credentials Variable when updating multiple tasks, the Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Credentials Unresolved - into which you enter a variable, variable and text, or Credentials name. The Credentials field is not used.</p> </div>
Local Filename	<p>Required if Transfer Type = FTP or SFTP; path and file name on the local server. That is, the "transfer from" file name.</p>
Remote Filename	<p>Required if Transfer Type = FTP or SFTP; path and file name on the remote server. That is, the "transfer to" file name.</p>
Retry Options	<p>This section contains specifications for retrying the task.</p>
Maximum Retries	<p>User-defined; maximum number of times that the Controller should retry this task after it has started and gone to a failed state.</p>
Retry Indefinitely	<p>User-defined; indicates whether the Controller should continue trying indefinitely to run this task. If you enable this field, it overrides any value placed in the Maximum Retries field.</p>
Retry Interval (Seconds)	<p>User-defined; number of seconds between each retry.</p>
Time Options	<p>This section contains time-related specifications for the task.</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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.</p>

Late Start Type	<p>Required if Late Start is enabled.</p> <p>Options:</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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.
Late Start Duration	<p>If Late Start Type = 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.</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>
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.</p> <p>Options:</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 = Time; Time after which the task finish time is considered late. Use HH:MM, 24-hour time.
Late Finish Duration	If Late Finish Type = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
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.</p> <p>Options:</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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
User Estimated Duration	Estimated amount of time it should normally take to run this task. The Controller uses this information to calculate the User Estimated End Time on a task instance record.
Statistics	This section contains time-related statistics for task instances of the task.
First Time Ran	System-supplied; date and time this task first ran.
Last Time Ran	System-supplied; date and time the task last ran.
Last Instance Duration	System-supplied; amount of time the task took to run the last time it ran.
Lowest Instance Time	System-supplied; shortest amount of time this task has taken to run.
Average Instance Time	System-supplied; average amount of time this task takes to run.
Highest Instance Time	System-supplied; longest amount of time this task has taken to run.
Number of Instances	System-supplied; number of instances in the database for this task.
Buttons	This section identifies the buttons displayed above and below the Task Details that let you perform various actions.
Save	Saves a new task record in the Controller database.
Update	Saves updates to the record.
New	Displays empty (except for default values) Details for creating a new task.
Launch Task	Manually launches the task.
View Parents	Displays a list of any parent Workflow tasks for this task.
Copy	Creates a copy of this task, which you are prompted to rename.
Delete	<p>Deletes the current record.</p> <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note You cannot delete a task if it is either:</p> <ul style="list-style-type: none"> Specified in an enabled Trigger. The only task specified in a disabled Trigger. </div>
Refresh	Refreshes any dynamic data displayed in the Details.
Close	For pop-up view only; closes the pop-up view of this task.

Tabs	This section identifies the tabs across the top of the Task Details that provide access to additional information about the task.										
Variables	Lists all variables associated with this record.										
Actions	<p>Allows you to specify actions that the Controller will take automatically based on events that occur during the execution of this task.</p> <p>Events are:</p> <ul style="list-style-type: none"> • Task instance status • Exit codes • Late start • Late finish • Early finish <p>Actions are:</p> <table border="1"> <tr> <td>Abort Action</td> <td>Abort the task if certain events occur. For details, see Abort Actions.</td> </tr> <tr> <td>Email Notification</td> <td>Send an email if certain events occur. For details, see Email Notification Actions.</td> </tr> <tr> <td>Set Variable</td> <td>Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow.</td> </tr> <tr> <td>SNMP Notification</td> <td>Send an email if certain events occur. For details, see SNMP Notification Actions.</td> </tr> <tr> <td>System Operation</td> <td>Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions.</td> </tr> </table>	Abort Action	Abort the task if certain events occur. For details, see Abort Actions .	Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .	Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .	SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .	System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .
Abort Action	Abort the task if certain events occur. For details, see Abort Actions .										
Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .										
Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .										
SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .										
System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .										
Virtual Resources	Lists all Virtual Resources to which this task is assigned.										
Mutually Exclusive	Lists all tasks that have been set to be mutually exclusive of this task.										
Instances	Lists all instances of the task.										
Triggers	Lists 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, the Controller 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 Triggers Overview .										
Notes	Lists all notes associated with this record.										
Versions	Stores copies of all previous versions of the current record. See Record Versioning .										

INFITRAN File Transfer Task Details


The following INFITRAN File Transfer Task Details is for an existing INFITRAN File Transfer task.



Depending on the values that you enter / select for these fields, and whether or not the INFITRAN File Transfer task has ever been launched, more (or less) fields may display. See the [field descriptions](#), below, for a description of all fields that may display in the INFITRAN File Transfer Task Details.

INFITRAN File Transfer Task Details Field Descriptions



The following table describes the fields, buttons, and tabs that display in INFITRAN File Transfer Task Details.

Field Name	Description
General	This section contains general information about the task.

Task Name	User-defined name of this task, containing a maximum 255 alphanumeric characters; variables supported. It is the responsibility of the user to develop a workable naming scheme for tasks.
Version	System-supplied; version number of the current record, which is incremented by the Controller every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
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.
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.
File Transfer Details	This section contains assorted detailed information about the task.
Transfer Type	Type of File Transfer server. Options: <ul style="list-style-type: none"> • FTP • SFTP • INFITRAN
Utility Agent	Name of the Windows or Linux/Unix Agent resource that will communicate with the Universal Command Agent. If you do not specify an Agent, you must specify an agent cluster .
Utility Agent Variable	If enabled, the Utility 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. Use the format: <code>\${variable name}</code> . The variable must be a supported type as described in Variables and Functions . <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Utility Agent Variable when updating multiple tasks, the Utility Agent field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Utility Agent Unresolved - into which you enter a variable, variable and text, or Utility Agent name. The Utility Agent field is not used.</p> </div>

Utility Agent Cluster	Group of Agents, one of which the Controller will choose to run this task. You can specify an agent cluster in addition to or in place of a specific Agent. If you specify an Agent and an agent cluster, the Controller first tries to run the task on the specific Agent. If the Agent is not available, the Controller reverts to the agent cluster. See Agent Clusters for more information.
Utility Agent Cluster Variable	<p>If enabled, the Utility 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. Use the format: <code>\$(variable name)</code>. The variable must be a supported type as described in Variables and Functions.</p> <div data-bbox="310 426 1446 606" style="background-color: #ffffcc; padding: 10px;"> <p> Note If you select Utility Agent Cluster Variable when updating multiple tasks, the Utility Agent Cluster field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Utility Agent Cluster Unresolved - into which you enter a variable, variable and text, or Utility Agent Cluster name. The Utility Agent Cluster field is not used.</p> </div>
Utility Credentials	Login credentials that the Agent will use to access the Universal Command server machine.
Utility Credentials Variable	<p>If enabled, the Utility Credentials field 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> <div data-bbox="310 888 1446 1068" style="background-color: #ffffcc; padding: 10px;"> <p> Note If you select Utility Credentials Variable when updating multiple tasks, the Utility Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Utility Credentials Unresolved - into which you enter a variable, variable and text, or Utility Credentials name. The Utility Credentials field is not used.</p> </div>
Transfer Mode	<p>Mode of data transfer.</p> <p>Options:</p> <ul style="list-style-type: none"> • Binary • Text
Encrypt	<p>The method of encryption that the Controller will use in the transfer.</p> <p>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

Compress	<p>The type of data compression used in the transfer, if any.</p> <p>Options:</p> <ul style="list-style-type: none"> • YES • NO • ZLIB • HASP
Codepage	<p>Options:</p> <p>(See Code Pages, below.)</p>
File Creation Option	<p>Specifies whether the transferred file should be created (new), appended, or replace any existing file.</p> <p>Options:</p> <ul style="list-style-type: none"> • None • APPEND • NEW • REPLACE
Trim Trailing Spaces	<p>If enabled, specifies that the Controller should trim trailing spaces from lines on a text transfer.</p>
Network Fault Tolerant	<p>Enable if the session is network fault tolerant.</p>
Runtime Directory	<p>Directory from which the application should be executed. Variables supported.</p>
Source Filename(s)	<p>Required; Path and file name on the source Infitran server.</p>
Destination Filename(s)	<p>Required; Path and file name on the destination Infitran server.</p>
Source File System	<p>Type of file system on the source server.</p> <p>Options:</p> <ul style="list-style-type: none"> • None • DSN • HFS • LIB
Destination File System	<p>Type of file system on the destination server.</p> <p>Options:</p> <ul style="list-style-type: none"> • None • DSN • HFS • LIB
Source Infitran Agent	<p>Required; Name of the Agent resource defined in the Controller that describes the source Infitran Agent machine (primary transfer server).</p>

Destination Infitran Agent	Required; Name of the Agent resource defined in the Controller that provides details about the destination Infitran 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 - Source Agent is an Infitran Agent defined in the Controller. • Infitran Agent Variable - Source Agent will be defined by setting the variable in the Source Infitran Agent field. • Infitran Agent Hostname - Source Agent runs on the host name 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 - Destination Agent is an Infitran Agent defined in the Controller. • Infitran Agent Variable - Destination Agent will be defined by setting the variable in the Destination Infitran Agent field. • Infitran Agent Hostname - Destination Agent runs on the host name 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	If enabled, the Source Credentials field (see above) converts from a reference field (where you browse and select a credentials record from the Opwise Controller 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 . <div style="background-color: #ffffcc; padding: 10px;"> <p> Note If you select Source Credentials Variable when updating multiple tasks, the Source Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Source Credentials Unresolved - into which you enter a variable, variable and text, or Source Credentials name. The Source Credentials field is not used.</p> </div>
Destination Credentials Variable	If enabled, the Destination Credentials field (see above) converts from a reference field (where you browse and select a credentials record from the Opwise Controller 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 . <div style="background-color: #ffffcc; padding: 10px;"> <p> Note If you select Destination Credentials Variable when updating multiple tasks, the Destination Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Destination Credentials Unresolved - into which you enter a variable, variable and text, or Destination Credentials name. The Destination Credentials field is not used.</p> </div>
Append Source Open Options	Any additional free form open command options for the source (primary) transfer server.
Append Destination Open Option	Any additional free form open command options for the destination (secondary) transfer server.

Append Infitran Options	Any additional free form Universal Data Mover command options.
Exit Code Processing	Specifies how the Controller should determine whether the executed command failed or completed successfully. Options: <ul style="list-style-type: none"> • Success Exitcode Range - Command is considered completed successfully if its exit code falls within the range specified in the Exit Codes field. • Failure Exitcode Range - Command is considered failed if its exit code falls within the range specified in the Exit Codes field. • Success Output Contains - Command is considered completed successfully if its output contains the text specified in the Scan Output For field. • Failure Output Contains - Command is considered failed if its output contains the text specified in the Scan Output For field. • Step Conditions (z/OS only) - 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	Required if Exit Code Processing = Success Output Contains or Failure Output Contains; 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 Exitcode Range; range of exit codes. Format: Numeric. Use commas to list a series of exit codes; 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; text for which the Controller should scan the output file. The Controller will process this field as a regular expression.
Output File-Exit Codes	Required if Output Type = File; path and file name of the output file that should be scanned for the text in the Scan Output For field.
Retry Options	This section contains specifications for retrying the task.
Retry Exit Codes	Exit code range for which an auto-retry of tasks in FAILED status will occur. Exit code ranges must be in the same format as ranges specified in the Exit Codes field. Maximum Retries must be greater than 0. If this field is empty, any exit code potentially will cause a retry.
Maximum Retries	User-defined; maximum number of times that the Controller should retry this task after it has started and gone to a failed state.
Retry Indefinitely	User-defined; indicates whether the Controller should continue trying indefinitely to run this task. If you enable this field, it overrides any value placed in the Maximum Retries field.
Retry Interval (Seconds)	User-defined; number of seconds between each retry.
Time Options	This section contains time-related specifications for the task.
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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.

Late Start Type	<p>Required if Late Start is enabled.</p> <p>Options:</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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.
Late Start Duration	<p>If Late Start Type = 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.</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>
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.</p> <p>Options:</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 = Time; Time after which the task finish time is considered late. Use HH:MM, 24-hour time.
Late Finish Duration	If Late Finish Type = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
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.</p> <p>Options:</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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
User Estimated Duration	Estimated amount of time it should normally take to run this task. The Controller uses this information to calculate the User Estimated End Time on a task instance record.
Statistics	This section contains time-related statistics for task instances of the task.
First Time Ran	System-supplied; date and time this task first ran.
Last Time Ran	System-supplied; date and time the task last ran.
Last Instance Duration	System-supplied; amount of time the task took to run the last time it ran.
Lowest Instance Time	System-supplied; shortest amount of time this task has taken to run.
Average Instance Time	System-supplied; average amount of time this task takes to run.
Highest Instance Time	System-supplied; longest amount of time this task has taken to run.
Number of Instances	System-supplied; number of instances in the database for this task.
Buttons	This section identifies the buttons displayed above and below the Task Details that let you perform various actions.
Save	Saves a new task record in the Controller database.
Update	Saves updates to the record.
New	Displays empty (except for default values) Details for creating a new task.
Launch Task	Manually launches the task.
View Parents	Displays a list of any parent Workflow tasks for this task.
Copy	Creates a copy of this task, which you are prompted to rename.
Delete	Deletes the current record.
Refresh	Refreshes any dynamic data displayed in the Details.
Close	For pop-up view only; closes the pop-up view of this task.
Tabs	This section identifies the tabs across the top of the Task Details that provide access to additional information about the task.
Variables	Lists all variables associated with this record.

Actions	<p>Allows you to specify actions that the Controller will take automatically based on events that occur during the execution of this task.</p> <p>Events are:</p> <ul style="list-style-type: none"> • Task instance status • Exit codes • Late start • Late finish • Early finish <p>Actions are:</p> <table border="1" data-bbox="272 478 1481 835"> <tr> <td data-bbox="272 478 412 548">Abort Action</td> <td data-bbox="417 478 1481 548">Abort the task if certain events occur. For details, see Abort Actions.</td> </tr> <tr> <td data-bbox="272 554 412 623">Email Notification</td> <td data-bbox="417 554 1481 623">Send an email if certain events occur. For details, see Email Notification Actions.</td> </tr> <tr> <td data-bbox="272 630 412 699">Set Variable</td> <td data-bbox="417 630 1481 699">Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow.</td> </tr> <tr> <td data-bbox="272 705 412 774">SNMP Notification</td> <td data-bbox="417 705 1481 774">Send an email if certain events occur. For details, see SNMP Notification Actions.</td> </tr> <tr> <td data-bbox="272 781 412 850">System Operation</td> <td data-bbox="417 781 1481 850">Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions.</td> </tr> </table>	Abort Action	Abort the task if certain events occur. For details, see Abort Actions .	Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .	Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .	SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .	System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .
Abort Action	Abort the task if certain events occur. For details, see Abort Actions .										
Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .										
Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .										
SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .										
System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .										
Virtual Resources	Lists all Virtual Resources to which this task is assigned.										
Mutually Exclusive	Lists all tasks that have been set to be mutually exclusive of this task.										
Instances	Lists all instances of the task.										
Triggers	Lists 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, the Controller 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 Triggers Overview .										
Notes	Lists all notes associated with this record.										
Versions	Stores copies of all previous versions of the current record. See Record Versioning .										

Viewing a File Transfer Task Instance

When a File Transfer task is launched, the Controller creates a task instance record of that task.

A task instance contains detailed information about a single execution of that task.

You can access a task instance from:

- **Instances tab** on the [FTP/SFTP](#) or [INFITRAN](#) File Transfer Task Details for that task
- [Activity Monitor](#)
- [Task Instances list](#)

FTP/SFTP File Transfer Task Instance Details


The following FTP/SFTP File Transfer Task Instance Details contains information on the execution of the task shown in the [FTP/SFTP File Transfer Task Details](#).



FTP/SFTP File Transfer Task Instance Details Field Descriptions


The following table describes the fields, buttons, and tabs that display in FTP/SFTP File Transfer Task Instance Details.

Field Name	Description
General	This section contains general information about the task instance.
Instance Name	Name of this task instance.
Reference Id	System-supplied; the Controller increments this number each time the task is run.
Task	Name of the task that was run to create this task instance. Click the icon to display Task Details for the task.

Invoked by	<p>System-supplied; how the task instance was launched.</p> <p>Options:</p> <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 identify the user, check the Execution User column for that task instance on the Task Instances screen or, on most task instance screens, the Execution User field.
Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
Execution User	System-supplied; if the task was launched manually, the ID of the user who launched it.
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.
Status	This section contains information about the current status of the task instance.
Status	System-supplied; see Task Instance Statuses .
Exit Code	System-supplied; the exit code captured by the Agent when executing the task (for example, a command or script).
Status Description	System-supplied; additional information, if any, about the status of the task instance.
Queued Time	System-supplied; Date and time the task was queued for processing.
Trigger Time	System-supplied; Date and time the task instance was triggered.
Launch Time	System-supplied; Date and time the task instance was launched.
Start Time	System-supplied; Date and time the task instance started.
End Time	System-supplied; Date and time the task instance completed.
Duration	System-supplied; amount of time the task instance took to run.
File Transfer Details	This section contains assorted detailed information about the task instance.


Transfer Type	<p>Type of File Transfer server.</p> <p>Options:</p> <ul style="list-style-type: none"> • FTP • SFTP • INFITRAN
Command	<p>File Transfer command being executed.</p> <p>Options:</p> <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.
Job Card (z/OS only)	<p>For z/OS, the job card information for the JCL statement. Example:</p> <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <pre>//File TransferJOB01 JOB (File Transfer,001),FANNY,MSGCLASS=X,MSGLEVEL=(1,1),NOTIFY=&SYSUID,CLASS=A</pre> </div>
Subcommands (z/OS only)	<p>For z/OS, any subcommands used in the JCL statement.</p>
Agent	<p>Name of the Agent resource that identifies the machine where the operation will run. If you do not specify an Agent, you must specify an Agent Cluster.</p>
Agent Variable	<p>If enabled, 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. Use the format: <code>\${variable name}</code>. The variable must be a supported type as described in Variables and Functions.</p> <div style="background-color: #ffffcc; padding: 10px; margin: 10px 0;"> <p> Note If you select Agent Variable when updating multiple tasks, the Agent field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Agent Unresolved - into which you enter a variable, variable and text, or Agent name. The Agent field is not used.</p> </div>
Transfer Mode	<p>Mode of data transfer.</p> <p>Options:</p> <ul style="list-style-type: none"> • Binary • ASCII

Transfer Method	<p>Transfer method.</p> <p>Options:</p> <ul style="list-style-type: none"> • Active • Passive • Extended Passive
Agent Cluster	<p>Group of Agents, one of which the Controller will choose to run this task. You can specify an agent cluster in addition to or in place of a specific Agent. If you specify an Agent and an agent cluster, the Controller first tries to run the task on the specific agent. If the Agent is not available, the Controller reverts to the agent cluster. See Agent Clusters for more information.</p>
Agent Cluster Variable	<p>If enabled, 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. Use the format: <code>\${variable name}</code>. The variable must be a supported type as described in Variables and Functions.</p> <div data-bbox="339 663 1446 848" style="background-color: #ffffcc; padding: 10px;"> <p> Note If you select Agent Cluster Variable when updating multiple tasks, the Agent Cluster field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Agent Cluster Unresolved - into which you enter a variable, variable and text, or Agent Cluster name. The Agent Cluster field is not used.</p> </div>
Cluster Broadcast	<p>Group of Agents, all of which will run this task. You can specify a Cluster Broadcast in place of a specific Agent and/or agent cluster. Each instance of the task running on its own Agent becomes a separate task instance record in the database and displays separately on the Activity screen. See Agent Clusters for more information about defining agent clusters.</p>
Remote Server	<p>Required if Transfer Type = FTP or SFTP; name or IP address of the File Transfer server. This machine may or may not be the same as the 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 that the Agent will use to access the FTP or SFTP server machine. If the File Transfer server and Agent are running on the same machine, enter the same credentials as those you entered in the Credentials field.</p>
FTP Credentials Variable	<p>If enabled, the FTP Credentials field 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> <div data-bbox="339 1509 1446 1694" style="background-color: #ffffcc; padding: 10px;"> <p> Note If you select FTP Credentials Variable when updating multiple tasks, the FTP Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - FTP Credentials Unresolved - into which you enter a variable, variable and text, or FTP Credentials name. The FTP Credentials field is not used.</p> </div>
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>

Credentials Variable	<p>If enabled, the Credentials field 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> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Credentials Variable when updating multiple tasks, the Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Credentials Unresolved - into which you enter a variable, variable and text, or Credentials name. The Credentials field is not used.</p> </div>
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.
Retry Options	This section contains specifications for retrying the task.
Maximum Retries	User-defined; maximum number of times that the Controller should retry this task after it has started and gone to a failed state.
Retry Indefinitely	User-defined; indicates whether the Controller should continue trying indefinitely to run this task. If you enable this field, it overrides any value placed in the Maximum Retries field.
Retry Interval	User-defined; number of seconds between each retry.
Current Retry Count	System-supplied; current number of times that the Controller has retried the task after it first went to failure status.
Time Options	This section contains time-related statistics for the task instance.
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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.
Started Late	System-supplied; this field is flagged if the task started later than the time specified in the Late Start fields.
Late Start Type	<p>Required if Late Start is enabled.</p> <p>Options:</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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.

Late Start Duration	<p>If Late Start Type = 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.</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>
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>
Finished Late	<p>System-supplied; this field is flagged if the task finished later than the time or duration specified in the Late Finish fields.</p>
Late Finish Type	<p>Required if Late Finish is enabled.</p> <p>Options:</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 = Time; 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 = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.</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>
Finished Early	<p>System-supplied; this field is flagged if the task finished earlier than the time specified in the Early Finish fields.</p>
Early Finish Type	<p>Required if Early Finish is enabled.</p> <p>Options:</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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.</p>

Finished Early	System-supplied; this field is flagged if the task finished earlier than the time specified in the Early Finish fields.
Statistics	This section contains time-related statistics for the task instance.
User Estimated End Time	System-supplied; if the user entered information into the User Estimated Duration field in the task Details, the Controller uses this information to calculate an end time for the task instance, based on the date/time the task instance started.
Average Estimated End Time	System-supplied; average estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Shortest Estimated End Time	System-supplied; shortest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Longest Estimated End Time	System-supplied; longest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Buttons	This section identifies the buttons displayed above and below the Task Instance Details that let you perform various actions.
Update	Saves updates to the record.
Force Finish	See Force Finishing a Task .
Hold	Places the task instance on Hold (see Putting a Task on Hold).
Skip	For tasks loaded into the schedule that have not yet run; allows you to tell the Controller to skip this task. See Skipping a Task .
Re-run	See Re-running a Task .
View Parent	Displays the task instance Details for the parent Workflow of this task instance.
Delete	Deletes the current record.
Refresh	Refreshes any dynamic data displayed in the Details.
Close	For pop-up view only; closes the pop-up view of this task instance.
Tabs	This section identifies the tabs across the top of the Task Instance Details that provide access to additional information about the task instance.
Virtual Resources	Lists all Virtual Resources to which this task is assigned.
Exclusive Requests	Lists all records in the Exclusive Requests table (<code>ops_exclusive_order</code>) for this task instance.

Output	Displays output generated from the process. <div data-bbox="339 205 1445 336" style="background-color: #ffffcc; padding: 10px;"> Note For File Transfer task instances, output always is automatically retrieved and is available from the Output tab.</div>
Notes	Lists all notes associated with this record.

INFITRAN File Transfer Task Instance Details

The following INFITRAN File Transfer Task Instance Details contains information on the execution of the task shown in the [INFITRAN File Transfer Task Details](#).

The screenshot displays the 'File Transfer Task Instance Details' window for 'stonebranch-filetransfertask-02'. The window is divided into several sections:



- General:** Instance Name: stonebranch-filetransfertask-02, Reference Id: 2, Task: stonebranch-filetransfertask-02, Invoked By: Manually Launched, Task Description: Test Entry, Member of Business Services: [dropdown], Execution User: stonebranch-user-01, Virtual Resource Priority: 10, Hold Resources on Failure: [checkbox].
- Status:** Status: Success, Exit Code: 0, Status Description: [empty], Queued Time: 2015-02-06 14:45:37 -0500, Trigger Time: [empty], Start Time: 2015-02-06 14:45:41 -0500, Launch Time: 2015-02-06 14:45:37 -0500, End Time: 2014-06-20 13:09:49 -0400, Duration: 0 Seconds.
- File Transfer Details:** Transfer Type: INFITRAN, agentId: [empty], Transfer Mode: Text, Agent Variable: [checkbox], Encrypt: NO, Agent Cluster: [empty], Compress: NO, Agent Cluster Variable: [checkbox], Codepage: -- None --, Credentials: [empty], Credentials Variable: [checkbox], Runtime Directory: [empty], Form or Script: Script, Script: Sample Infitran Script (testfile.txt), Source Credentials: [empty], Destination Credentials: QA NIX qatest, Source Credentials Variable: [checkbox], Destination Credentials Variable: [checkbox], Append Infitran Options: [empty], Exit Code Processing: Success Exitcode Range, Exit Codes: 0.
- Retry Options:** Retry Exit Codes: [empty], Maximum Retries: 0, Retry Indefinitely: [checkbox], Retry Interval (Seconds): 60, Current Retry Count: 0.
- Statistics:** User Estimated End Time: [empty], Average Estimated End Time: 2014-07-22 11:09:33 -0400, Shortest Estimated End Time: 2014-07-22 11:09:33 -0400, Longest Estimated End Time: 2014-07-22 11:09:34 -0400.


INFITRAN File Transfer Task Instance Details Field Descriptions

The following table describes the fields, buttons, and tabs that display in INFITRAN File Transfer Task Instance Details.



Field Name	Description
General	This section contains general information about the task instance.

Instance Name	Name of this task instance.
Reference Id	System-supplied; the Controller increments this number each time the task is run.
Task	Name of the task that was run to create this task instance. Click the icon to display Task Details for the task.
Invoked by	System-supplied; how the task instance was launched. Options: <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 identify the user, check the Execution User column for that task instance on the Task Instances screen or, on most task instance screens, the Execution User field.
Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
Execution User	System-supplied; if the task was launched manually, the ID of the user who launched it.
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.
Status	This section contains information about the current status of the task instance.
Status	System-supplied; see Task Instance Statuses .
Exit Code	System-supplied; the exit code captured by the Agent when executing the task (for example, a command or script).
Status Description	System-supplied; additional information, if any, about the status of the task instance.
Queued Time	System-supplied; Date and time the task was queued for processing.
Trigger Time	System-supplied; Date and time the task instance was triggered.
Launch Time	System-supplied; Date and time the task instance was launched.
Start Time	System-supplied; Date and time the task instance started.

End Time	System-supplied; Date and time the task instance completed.
Duration	System-supplied; amount of time the task instance took to run.
File Transfer Details	This section contains assorted detailed information about the task instance.
Transfer Type	Type of File Transfer server. Options: <ul style="list-style-type: none"> • FTP • SFTP • INFITRAN
Utility Agent	Name of the Windows or Linux/Unix Agent resource that will communicate with the Universal Command Agent. If you do not specify an Agent, you must specify an agent cluster .
Utility Agent Variable	If enabled, the Utility 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. Use the format: <code>\${variable name}</code> . The variable must be a supported type as described in Variables and Functions . <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Utility Agent Variable when updating multiple tasks, the Utility Agent field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Utility Agent Unresolved - into which you enter a variable, variable and text, or Utility Agent name. The Utility Agent field is not used.</p> </div>
Utility Agent Cluster	Group of Agents, one of which the Controller will choose to run this task. You can specify an agent cluster in addition to or in place of a specific Agent. If you specify an Agent and an agent cluster, the Controller first tries to run the task on the specific Agent. If the Agent is not available, the Controller reverts to the agent cluster. See Agent Clusters for more information.
Utility Agent Cluster Variable	If enabled, the Utility 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. Use the format: <code>\${variable name}</code> . The variable must be a supported type as described in Variables and Functions . <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Utility Agent Cluster Variable when updating multiple tasks, the Utility Agent Cluster field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Utility Agent Cluster Unresolved - into which you enter a variable, variable and text, or Utility Agent Cluster name. The Utility Agent Cluster field is not used.</p> </div>
Utility Credentials	Login credentials that the Agent will use to access the Universal Command server machine.


Utility Credentials Variable	<p>If enabled, the Utility Credentials field 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> <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note If you select Utility Credentials Variable when updating multiple tasks, the Utility Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Utility Credentials Unresolved - into which you enter a variable, variable and text, or Utility Credentials name. The Utility Credentials field is not used.</p> </div>
Transfer Mode	<p>Mode of data transfer.</p> <p>Options:</p> <ul style="list-style-type: none"> • Binary • ASCII
Encrypt	<p>The method of encryption that the Controller will use in the transfer.</p> <p>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
Compress	<p>The type of data compression used in the transfer, if any.</p> <p>Options:</p> <ul style="list-style-type: none"> • YES • NO • ZLIB • HASP
Codepage	<p>Options: (see Code Pages, below)</p>
File Creation Option	<p>Specifies whether the transferred file should be created (new), appended, or replace any existing file.</p> <p>Options:</p> <ul style="list-style-type: none"> • None • APPEND • NEW • REPLACE
Trim Trailing Spaces	<p>If enabled, specifies that the Controller should trim trailing spaces from lines on a text transfer.</p>
Network Fault Tolerant	<p>Enable if the session is network fault tolerant.</p>

Runtime Directory	Directory from which the application should be executed. Variables supported.
Source Filename	Required; Path and file name on the source Infitran server.
Destination Filename	Required; 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; Name of the Agent resource defined in the Controller that describes the source Infitran Agent machine (primary transfer server).
Destination Infitran Agent	Required; Name of the Agent resource defined in the Controller that provides details about the destination Infitran 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 - Source Agent is an Infitran Agent defined in the Controller. • Infitran Agent Variable - Source Agent will be defined by setting the variable in the Source Infitran Agent field. • Infitran Agent Hostname - Source Agent runs on the host name 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 - Destination Agent is an Infitran Agent defined in the Controller. • Infitran Agent Variable - Destination Agent will be defined by setting the variable in the Destination Infitran Agent field. • Infitran Agent Hostname - Destination Agent runs on the host name 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.

<p>Source Credentials Variable</p>	<p>If enabled, the Source Credentials field (see above) converts from a reference field (where you browse and select a credentials record from the Opswise Controller credentials table) 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> <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note If you select Source Credentials Variable when updating multiple tasks, the Source Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Source Credentials Unresolved - into which you enter a variable, variable and text, or Source Credentials name. The Source Credentials field is not used.</p> </div>
<p>Destination Credentials Variable</p>	<p>If enabled, the Destination Credentials field (see above) converts from a reference field (where you browse and select a credentials record from the Opswise Controller credentials table) 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> <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note If you select Destination Credentials Variable when updating multiple tasks, the Destination Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Destination Credentials Unresolved - into which you enter a variable, variable and text, or Destination Credentials name. The Destination Credentials field is not used.</p> </div>
<p>Append Source Open Options</p>	<p>Any additional free form open command options for the source (primary) transfer server.</p>
<p>Append Destination Open Option</p>	<p>Any additional free form open command options for the destination (secondary) transfer server.</p>
<p>Append Infitran Options</p>	<p>Any additional free form Universal Data Mover command options.</p>
<p>Exit Code Processing</p>	<p>Specifies how the Controller should determine whether the executed command failed or completed successfully.</p> <p>Options:</p> <ul style="list-style-type: none"> • Success Exitcode Range - Command is considered completed successfully if its exit code falls within the range specified in the Exit Codes field. • Failure Exitcode Range - Command is considered failed if its exit code falls within the range specified in the Exit Codes field. • Success Output Contains - Command is considered completed successfully if its output contains the text specified in the Scan Output For field. • Failure Output Contains - Command is considered failed if its output contains the text specified in the Scan Output For field. • Step Conditions (z/OS only) - 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).
<p>Output Type-Exit Code</p>	<p>Required if Exit Code Processing = Success Output Contains or Failure Output Contains; type of output.</p> <p>Options:</p> <ul style="list-style-type: none"> • Standard Output (STDOUT) • Standard Error (STDERR) • File
<p>Exit Codes</p>	<p>Required if Exit Code Processing = Success Exitcode Range or Failure Exitcode Range; range of exit codes. Format: Numeric. Use commas to list a series of exit codes; use hyphens to specify a range. Example: 1,5, 22-30.</p>

Scan Output For	Required if Exit Code Processing = Success Output Contains or Failure Output Contains; text for which the Controller should scan the output file. The Controller will process this field as a regular expression.
Output File-Exit Codes	Required if Output Type = File; path and file name of the output file that should be scanned for the text in the Scan Output For field.
Retry Options	This section contains specifications for retrying the task.
Retry Exit Codes	Exit code range for which an auto-retry of tasks in FAILED status will occur. Exit code ranges must be in the same format as ranges specified in the Exit Codes field. Maximum Retries must be greater than 0. If this field is empty, any exit code potentially will cause a retry.
Maximum Retries	User-defined; maximum number of times that the Controller should retry this task after it has started and gone to a failed state.
Retry Indefinitely	User-defined; indicates whether the Controller should continue trying indefinitely to run this task. If you enable this field, it overrides any value placed in the Maximum Retries field.
Retry Interval	User-defined; number of seconds between each retry.
Current Retry Count	System-supplied; current number of times that the Controller has retried the task after it first went to failure status.
Time Options	This section contains time-related specifications for the task instance.
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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.
Started Late	System-supplied; this field is flagged if the task started later than the time specified in the Late Start fields.
Late Start Type	Required if Late Start is enabled. Options: <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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.
Late Start Duration	If Late Start Type = 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.

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.
Finished Late	System-supplied; this field is flagged if the task finished later than the time or duration specified in the Late Finish fields.
Late Finish Type	Required if Late Finish is enabled. Options: <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 = Time; Time after which the task finish time is considered late. Use HH:MM, 24-hour time.
Late Finish Duration	If Late Finish Type = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
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.
Finished Early	System-supplied; this field is flagged if the task finished earlier than the time specified in the Early Finish fields.
Early Finish Type	Required if Early Finish is enabled. Options: <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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Statistics	This section contains time-related statistics for the task instances of this task.
User Estimated End Time	System-supplied; if the user entered information into the User Estimated Duration field in the task Details, the Controller uses this information to calculate an end time for the task instance, based on the date/time the task instance started.
Average Estimated End Time	System-supplied; average estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.

Shortest Estimated End Time	System-supplied; shortest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Longest Estimated End Time	System-supplied; longest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Buttons	This section identifies the buttons displayed above and below the Task Instance Details that let you perform various actions.
Update	Saves updates to the record.
Force Finish	See Force Finishing a Task .
Hold	Places the task instance on Hold (see Putting a Task on Hold).
Skip	For tasks loaded into the schedule that have not yet run; allows you to tell the Controller to skip this task. See Skipping a Task .
Re-run	See Re-running a Task .
View Parent	Displays the task instance Details for the parent Workflow of this task instance.
Delete	Deletes the current record.
Refresh	Refreshes any dynamic data displayed in the Details.
Close	For pop-up view only; closes the pop-up view of this task instance.
Tabs	This section identifies the tabs across the top of the Task Instance Details that provide access to additional information about the task instance.
Virtual Resources	Lists all Virtual Resources to which this task is assigned.
Exclusive Requests	Lists all records in the Exclusive Requests table (<code>ops_exclusive_order</code>) for this task instance.
Output	Displays output generated from the process. <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note For File Transfer task instances, output always is automatically retrieved and is available from the Output tab.</p> </div>
Notes	Lists all notes associated with this record.

Output Redirection

An Agent processes File Transfer (INFITRAN), Indesca, and SAP tasks differently than Windows and Linux/Unix tasks. 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 tasks. 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 tasks).

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

For a File Transfer (INFITRAN) task, 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:\INFITRANAOUT\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"
```

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](#).

Running a File Transfer Task

You can run a File Transfer task:

- Manually, by clicking the [Launch Task](#) or [Launch Task with Variables](#) button in the File Transfer tasks list or File Transfer Task Details [Action menu](#).
- As part of a [workflow](#).
- [Specify triggers](#) that run the task automatically based on times or events.

Monitoring Task Execution

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

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	cp862	IBM Portugal 1140
ISO8859-13	cp863	IBM German 1141
ISO8859-14	cp864	IBM Danish 1142
ISO8859-15	cp865	IBM Finish 1143
	cp866	IBM Italian 1144
	cp869	IBM Spanish 1145
	cp874	IBM UK 1146
	cp1250	IBM Swiss 1148
	cp1251	IBM Greek 4971
	cp1252	
	cp1253	
	cp1254	
	cp1255	
	cp1256	
	cp1257	
	cp1258	

Manual Task

- Overview
- Creating a Manual Task
 - Manual Task Details
- Manual Task Field Descriptions
- Viewing a Manual Task Instance
 - Manual Task Instance Details
 - Manual Task Instance Details Field Descriptions
- Running a Manual Task
- Monitoring Task Execution

Overview

Manual tasks are used to create a pause in a Workflow during which the user must take some action.

The processing of a Manual task within a Workflow is described here:

Step 1	While its upstream tasks are running, a Manual task instance remains in WAITING status. When the Manual task instance launches, it goes immediately into ACTION REQUIRED status, meaning you must perform some manual activity. Opwise Controller also sets the Started Time in the Manual task instance to the time that the task instance went into the ACTION REQUIRED status.
Step 2	Optionally, you can re-set the Started Time of the Manual task by issuing the Set Started command. Either: <ol style="list-style-type: none"> 1. On the Activity Monitor, right-click the Manual task and select Set Started. 2. On the Workflow Monitor, right-click the Manual task and select Commands > Set Started.
Step 3	When you have completed the activities called for in the Manual task, you must indicate that the task is completed and that the Workflow can continue. Either: <ol style="list-style-type: none"> 1. On the Activity Monitor, right-click the Manual task and select Set Completed. 2. On the Workflow Monitor, right-click the Manual task and select Commands > Set Completed.
Step 4	The Manual task goes into SUCCESS status, the End Time is set, and the Workflow continues. If the Manual task is not completed but you still want the Workflow to continue, select Force Finish .



Note

You also can set a Manual task to STARTED or COMPLETED status from the [Command Line Interface \(CLI\)](#).

Creating a Manual Task

Step 1 From the [Automation Center](#) navigation pane, select **Tasks > Manual Tasks**. The Manual Tasks list displays a list of all currently defined Manual tasks.

Below the list, Manual Task Details for a new Manual task displays. (You also can click the **New** button to display Manual Task Details for a new Manual task.)

The screenshot shows the 'Manual Tasks' interface. At the top, there is a 'Manual Tasks' tab and a '5 Manual Tasks' dropdown. Below this is a table with columns: Task Name, Task Description, Updated By, and Updated. The table contains five rows of tasks, all updated by 'ops.admin' on '2014-06-23'.

Task Name	Task Description	Updated By	Updated
stonebranch-manualtask-01		ops.admin	2014-06-23 17:17:37 -0400
stonebranch-manualtask-02		ops.admin	2014-06-23 17:18:01 -0400
stonebranch-manualtask-03		ops.admin	2014-06-23 17:17:55 -0400
stonebranch-manualtask-04		ops.admin	2014-06-23 17:17:51 -0400
stonebranch-manualtask-05		ops.admin	2014-06-23 17:17:43 -0400

Below the table is the 'Manual Task Details' form. It has tabs for 'Manual Task', 'Variables', 'Actions', 'Virtual Resources', 'Mutually Exclusive', 'Instances', 'Triggers', 'Notes', and 'Versions'. The 'Manual Task' tab is active, showing a 'General' section with fields for Task Name, Task Description, Member of Business Services, Hold on Start, Virtual Resource Priority, and Hold Resources on Failure. There is also a 'Time Options' section with fields for Late Start, Late Finish, Early Finish, and User Estimated Duration.

Step 2 Enter/select Details for a new Manual task, using the field descriptions below as a guide.

- Required fields display in **boldface**.
- Default values for fields, if available, display automatically.

To display more of the Details fields on the screen, you can temporarily [hide the list](#).



Note

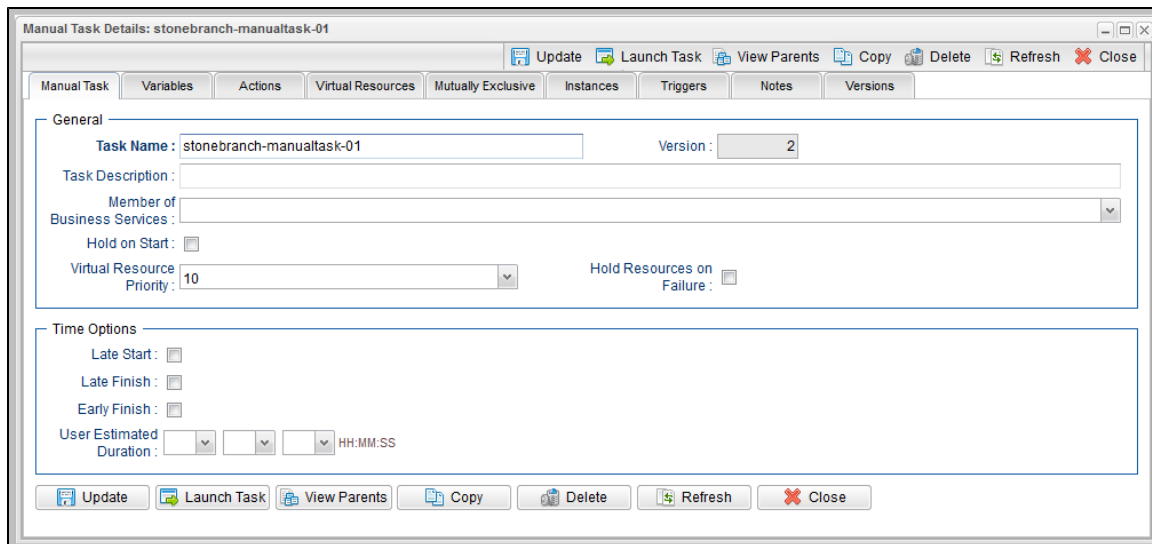
If you view [Manual Task Details](#) for an existing Manual task by clicking a task in the list, and then want to create a new Manual task, you must click the **New** button that displays above and below the Details.

Step 3 Click the **Save** button. The task is added to the database, and all buttons and tabs in the Task Details are enabled.

Manual Task Details

The following Manual Task Details is for an existing Manual task.

Depending on the values that you enter / select for these fields, and whether or not the Manual task has ever been launched, more (or less) fields may display. See the [field descriptions](#), below, for a description of all fields that may display in the Manual Task Details.




Manual Task Field Descriptions

The following table describes the fields, buttons, and tabs that display in the Manual Task Details.

Field Name	Description
General	This section contains general information about the task.
Task Name	User-defined name of this task, containing a maximum 255 alphanumeric characters; variables supported. It is the responsibility of the user to develop a workable naming scheme for tasks.
Version	System-supplied; version number of the current record, which is incremented by the Controller every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
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.
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.

Time Options	This section contains time-related specifications for the task.
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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.
Late Start Type	Required if Late Start is enabled. Options: <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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.
Late Start Duration	If Late Start Type = 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.
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: <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 = Time; Time after which the task finish time is considered late. Use HH:MM, 24-hour time.
Late Finish Duration	If Late Finish Type = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
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.</p> <p>Options:</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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
User Estimated Duration	Estimated amount of time it should normally take to run this task. The Controller uses this information to calculate the User Estimated End Time on a task instance record.
Statistics	This section contains time-related statistics for task instances of the task.
First Time Ran	System-supplied; date and time this task first ran.
Last Time Ran	System-supplied; date and time the task last ran.
Last Instance Duration	System-supplied; amount of time the task took to run the last time it ran.
Lowest Instance Time	System-supplied; shortest amount of time this task has taken to run.
Average Instance Time	System-supplied; average amount of time this task takes to run.
Highest Instance Time	System-supplied; longest amount of time this task has taken to run.
Number of Instances	System-supplied; number of instances in the database for this task.
Buttons	This section identifies the buttons displayed above and below the Details that let you perform various actions.
Save	Saves a new task record in the Controller database.
Update	Saves updates to the record.
New	Displays empty (except for default values) Details for creating a new task.
Launch Task	Manually launches the task.
View Parents	Displays a list of any parent Workflow tasks for this task.
Copy	Creates a copy of this task, which you are prompted to rename.

Delete	<p>Deletes the current record.</p> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note You cannot delete a task if it is either:</p> <ul style="list-style-type: none"> Specified in an enabled Trigger. The only task specified in a disabled Trigger. </div>										
Refresh	Refreshes any dynamic data displayed in the Details.										
Close	For pop-up view only; closes the pop-up view of this task.										
Tabs	This section identifies the tabs across the top of the Details that provide access to additional information about the task.										
Variables	Lists all variables associated with this record.										
Actions	<p>Allows you to specify actions that the Controller will take automatically based on events that occur during the execution of this task.</p> <p>Events are:</p> <ul style="list-style-type: none"> Task instance status Exit codes Late start Late finish Early finish <p>Actions are:</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="background-color: #f2f2f2;">Abort Action</td> <td>Abort the task if certain events occur. For details, see Abort Actions.</td> </tr> <tr> <td style="background-color: #f2f2f2;">Email Notification</td> <td>Send an email if certain events occur. For details, see Email Notification Actions.</td> </tr> <tr> <td style="background-color: #f2f2f2;">Set Variable</td> <td>Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow.</td> </tr> <tr> <td style="background-color: #f2f2f2;">SNMP Notification</td> <td>Send an email if certain events occur. For details, see SNMP Notification Actions.</td> </tr> <tr> <td style="background-color: #f2f2f2;">System Operation</td> <td>Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions.</td> </tr> </table>	Abort Action	Abort the task if certain events occur. For details, see Abort Actions .	Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .	Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .	SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .	System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .
Abort Action	Abort the task if certain events occur. For details, see Abort Actions .										
Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .										
Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .										
SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .										
System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .										
Virtual Resources	Lists all Virtual Resources to which this task is assigned.										
Mutually Exclusive	Lists all tasks that have been set to be mutually exclusive of this task.										
Instances	Lists all instances of the task.										
Triggers	Lists 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, the Controller 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 Triggers Overview .										
Notes	Lists all notes associated with this record.										

Versions	Stores copies of all previous versions of the current record. See Record Versioning .
-----------------	---

Viewing a Manual Task Instance

When a Manual task is launched, the Controller creates a task instance record of that task.

A task instance contains detailed information about a single execution of that task.

You can access a task instance from:

- **Instances tab** on the [Manual Task Details](#) for that task
- [Activity Monitor](#)
- [Task Instances list](#)

Manual Task Instance Details

The following Manual Task Instance Details contains information on the execution of the task shown in the [Manual Task Details](#).

Manual Task Instance Details Field Descriptions

The following table describes the fields, buttons, and tabs that display in Manual Task Instance Details.

Field Name	Description
General	This section contains general information about the task instance.
Instance Name	Name of this task instance.
Reference Id	System-supplied; the Controller increments this number each time the task is run.

Task	Name of the task that was run to create this task instance. Click the icon to display Task Details for the task.
Invoked by	System-supplied; how the task instance was launched. Options: <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 identify the user, check the Execution User column for that task instance on the Task Instances screen or, on most task instance screens, the Execution User field.
Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
Execution User	System-supplied; if the task was launched manually, the ID of the user who launched it.
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.
Status	This section contains information about the current status of the task instance.
Status	System-supplied; see Task Instance Statuses .
Status Description	System-supplied; additional information, if any, about the status of the task instance.
Queued Time	System-supplied; Date and time the task was queued for processing.
Trigger Time	System-supplied; Date and time the task instance was triggered.
Launch Time	System-supplied; Date and time the task instance was launched.
Start Time	System-supplied; Date and time the task instance started.
End Time	System-supplied; Date and time the task instance completed.
Duration	System-supplied; amount of time the task instance took to run.
Time Options	This section contains time-related specifications for the task instance.

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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.
Started Late	System-supplied; this field is flagged if the task started later than the time specified in the Late Start fields.
Late Start Type	Required if Late Start is enabled. Options: <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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.
Late Start Duration	If Late Start Type = 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.
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.
Finished Late	System-supplied; this field is flagged if the task finished later than the time or duration specified in the Late Finish fields.
Late Finish Type	Required if Late Finish is enabled. Options: <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 = Time; Time after which the task finish time is considered late. Use HH:MM, 24-hour time.
Late Finish Duration	If Late Finish Type = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
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.

Finished Early	System-supplied; this field is flagged if the task finished earlier than the time specified in the Early Finish fields.
Early Finish Type	Required if Early Finish is enabled. Options: <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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Statistics	This section contains time-related statistics for the task instance.
User Estimated End Time	System-supplied; if the user entered information into the User Estimated Duration field in the task Details, the Controller uses this information to calculate an end time for the task instance, based on the date/time the task instance started.
Average Estimated End Time	System-supplied; average estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Shortest Estimated End Time	System-supplied; shortest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Longest Estimated End Time	System-supplied; longest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Buttons	This section identifies the buttons displayed above and below the Task Instance Details that let you perform various actions.
Update	Saves updates to the record.
Force Finish	See Force Finishing a Task .
Hold	Places the task instance on Hold (see Putting a Task on Hold).
Set Started	Sets the task instance to the Started status.
Set Completed	Sets the task instance to the Success status.
Skip	For tasks loaded into the schedule that have not yet run; allows you to tell the Controller to skip this task. See Skipping a Task .
Re-run	See Re-running a Task .
View Parent	Displays the task instance Details for the parent Workflow of this task instance.

Retrieve Output	See Retrieving Output .
Delete	Deletes the current record.
Refresh	Refreshes any dynamic data displayed in the Details.
Close	For pop-up view only; closes the pop-up view of this task instance.
Tabs	This section identifies the tabs across the top of the Task Instance Details that provide access to additional information about the task instance.
Virtual Resources	Lists all Virtual Resources to which this task is assigned.
Exclusive Requests	Lists all records in the Exclusive Requests table (<code>ops_exclusive_order</code>) for this task instance.
Notes	Lists all notes associated with this record.

Running a Manual Task

You can run a Manual task:

- Manually, by clicking the [Launch Task](#) or [Launch Task with Variables](#) button in the Manual Tasks list or Manual Task Details [Action menu](#).
- As part of a [workflow](#).
- [Specify triggers](#) that run the task automatically based on times or events.

Monitoring Task Execution

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

Timer Task

- Overview
- Creating a Timer Task
 - Timer Task Details
 - Timer Task Details Field Descriptions
- Viewing a Timer Task Instance
 - Timer Task Instance Details
 - Timer Task Instance Details Field Descriptions
- Running a Timer Task
- Monitoring Task Execution

Overview

The Timer task allows you to execute a timer 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 Timer Task

Step 1 From the Automation Center navigation pane, select **Tasks > Timer Tasks**. The Timer Tasks list displays a list of all currently defined Timer tasks.

Below the list, Timer Task Details for a new Timer task displays. (You also can click the **New** button to display Timer Task Details for a new Timer task.)

The screenshot shows the 'Timer Tasks' management interface. At the top, there is a list of 5 timer tasks:

Task Name	Task Description	Timer Type	Updated By	Updated
stonebranch-timertask-01		Seconds	ops.admin	2014-06-13 14:12:54 -0400
stonebranch-timertask-02		Seconds	ops.admin	2014-06-13 14:13:03 -0400
stonebranch-timertask-03		Seconds	ops.admin	2014-06-13 14:13:08 -0400
stonebranch-timertask-04		Seconds	ops.admin	2014-06-13 14:13:13 -0400
stonebranch-timertask-05		Seconds	ops.admin	2014-06-13 14:13:17 -0400


Below the list is the 'Timer Task Details' form, which includes the following sections:

- General:** Task Name (text input), Version (input field with value 1), Task Description (text input), Member of Business Services (dropdown menu), Hold on Start (checkbox), Virtual Resource Priority (input field with value 10), Hold Resources on Failure (checkbox).
- Timer Details:** Timer Type (dropdown menu with value Seconds), Wait Time in Seconds (input field).
- Time Options:** Late Start (checkbox).

Step 2 Enter/select Details for a new Timer task, using the [field descriptions](#) below as a guide.

- Required fields display in **boldface**.
- Default values for fields, if available, display automatically.

To display more of the Details fields on the screen, you can temporarily [hide the list](#).

 **Note**
 If you view [Timer Task Details](#) for an existing Timer task by clicking a task in the list, and then want to create a new Timer task, you must click the **New** button that displays above and below the Details.

Step 3 Click the **Save** button. The task is added to the database, and all buttons and tabs in the Task Details are enabled.

Timer Task Details

The following Linux/Unix Task Details is for an existing Linux/Unix task.

Depending on the values that you enter / select for these fields, and whether or not the Timer task has ever been launched, more (or less) fields may display. See the [field descriptions](#), below, for a description of all fields that may display in the Timer Task Details.

Timer Task Details Field Descriptions


The following table describes the fields, buttons, and tabs that display in the Timer Task Details.

Field Name	Description

General	This section contains general information about the task.
Task Name	User-defined name of this task, containing a maximum 255 alphanumeric characters; variables supported. It is the responsibility of the user to develop a workable naming scheme for tasks.
Version	System-supplied; version number of the current record, which is incremented by the Controller every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
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.
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.
Timer Details	This section contains assorted detailed information about the task.
Timer Type	User-supplied; the type of Timer command you want to execute. Options: <ul style="list-style-type: none"> • Time - Use the Wait Until Time field to specify the time of day that you want the Timer task to be completed. • Duration - Use the Wait Duration field to specify the number of hours, minutes, and/or seconds that the Timer task will run. • Seconds - Use the Wait Time in Seconds field to specify the number of seconds that the Timer task will run.
Wait Until Time	If Timer Type = Time; Time of day (in 24-hour time) that the Timer task should go to a completed status.
Wait Duration	If Timer Type = Duration; Number of hours, minutes, and/or seconds the Timer task will run.
Wait Time in Seconds	If Timer Type = Seconds; Number of seconds the Timer task will run.
Time Options	This section contains time-related specifications for the task.

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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.
Late Start Type	Required if Late Start is enabled. Options: <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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.
Late Start Duration	If Late Start Type = 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.
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: <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 = Time; Time after which the task finish time is considered late. Use HH:MM, 24-hour time.
Late Finish Duration	If Late Finish Type = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
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: <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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
User Estimated Duration	Estimated amount of time it should normally take to run this task. The Controller uses this information to calculate the User Estimated End Time on a task instance record.
Statistics	This section contains time-related statistics for the task instances of the task.
First Time Ran	System-supplied; date and time this task first ran.
Last Time Ran	System-supplied; date and time the task last ran.
Last Instance Duration	System-supplied; amount of time the task took to run the last time it ran.
Lowest Instance Time	System-supplied; shortest amount of time this task has taken to run.
Average Instance Time	System-supplied; average amount of time this task takes to run.
Highest Instance Time	System-supplied; longest amount of time this task has taken to run.
Number of Instances	System-supplied; number of instances in the database for this task.
Buttons	This section identifies the buttons displayed above and below the Task Details that let you perform various actions.
Save	Saves a new task record in the Controller database.
Update	Saves updates to the record.
New	Displays empty (except for default values) Details for creating a new task.
Launch Task	Manually launches the task.
View Parents	Displays a list of any parent Workflow tasks for this task.
Copy	Creates a copy of this task, which you are prompted to rename.

Delete button	<p>Deletes the current record.</p> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note You cannot delete a task if it is either:</p> <ul style="list-style-type: none"> Specified in an enabled Trigger. The only task specified in a disabled Trigger. </div>										
Refresh	Refreshes any dynamic data displayed in the Details.										
Close	For pop-up view only; closes the pop-up view of this task.										
Tabs	This section identifies the tabs across the top of the Task Details that provide access to additional information about the task.										
Variables	Lists all variables associated with this record.										
Actions	<p>Allows you to specify actions that the Controller will take automatically based on events that occur during the execution of this task.</p> <p>Events are:</p> <ul style="list-style-type: none"> Task instance status Exit codes Late start Late finish Early finish <p>Actions are:</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="background-color: #e0e0e0;">Abort Action</td> <td>Abort the task if certain events occur. For details, see Abort Actions.</td> </tr> <tr> <td style="background-color: #e0e0e0;">Email Notification</td> <td>Send an email if certain events occur. For details, see Email Notification Actions.</td> </tr> <tr> <td style="background-color: #e0e0e0;">Set Variable</td> <td>Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow.</td> </tr> <tr> <td style="background-color: #e0e0e0;">SNMP Notification</td> <td>Send an email if certain events occur. For details, see SNMP Notification Actions.</td> </tr> <tr> <td style="background-color: #e0e0e0;">System Operation</td> <td>Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions.</td> </tr> </table>	Abort Action	Abort the task if certain events occur. For details, see Abort Actions .	Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .	Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .	SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .	System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .
Abort Action	Abort the task if certain events occur. For details, see Abort Actions .										
Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .										
Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .										
SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .										
System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .										
Virtual Resources	Lists all Virtual Resources to which this task is assigned.										
Mutually Exclusive	Lists all tasks that have been set to be mutually exclusive of this task.										
Instances	Lists all instances of this Workflow.										
Triggers	Lists 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, the Controller 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 Triggers Overview .										
Notes	Lists all notes associated with this record.										

Viewing a Timer Task Instance

When a Timer task is launched, the Controller creates a task instance record of that task.

A task instance contains detailed information about a single execution of that task.

You can access a task instance from:

- **Instances tab** on the [Timer Task Details](#) for that task
- [Activity list](#)
- [Task Instances list](#)

Timer Task Instance Details

The following Timer Task Instance Details contains information on the execution of the task shown in the [Timer Task Details](#).

Timer Task Instance Details Field Descriptions

The following table describes the fields, buttons, and tabs that display in Timer Task Instance Details.

Field Name	Description
General	This section contains general information about the task instance.
Instance Name	Name of this task instance.
Reference Id	System-supplied; the Controller increments this number each time the task is run.
Task	Name of the task that was run to create this task instance. Click the icon to display Task Details for the task.

Invoked by	<p>System-supplied; how the task instance was launched.</p> <p>Options:</p> <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 identify the user, check the Execution User column for that task instance on the Task Instances screen or, on most task instance screens, the Execution User field.
Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
Execution User	System-supplied; if the task was launched manually, the ID of the user who launched it.
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.
Status	This section contains information about the current status of the task instance.
Status	System-supplied; see Task Instance Statuses .
Exit Code	System-supplied; the exit code captured by the Agent when executing the task (for example, a command or script).
Status Description	System-supplied; additional information, if any, about the status of the task instance.
Queued Time	System-supplied; Date and time the task was queued for processing.
Trigger Time	System-supplied; Date and time the task instance was triggered.
Launch Time	System-supplied; Date and time the task instance was launched.
Start Time	System-supplied; Date and time the task instance started.
End Time	System-supplied; Date and time the task instance completed.
Duration	System-supplied; amount of time the task instance took to run.

CPU Time	System-supplied; amount of CPU time the task took to run.
Timer Details	This section contains assorted detailed information about the task instance.
Timer Type	Type of Timer command that was executed. Options: <ul style="list-style-type: none"> • Time • Duration • Seconds
Wait Until Time	If Timer Type = Time; Time of day (in 24-hour time) that the Timer task should go to a completed status.
Wait Duration	If Timer Type = Duration; Number of hours, minutes, and/or seconds the Timer task will run.
Wait Time in Seconds	If Timer Type = Seconds; Number of seconds the Timer task will run.
Time Options	This section contains time-related specifications for the task instance.
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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.
Started Late	System-supplied; this field is flagged if the task started later than the time specified in the Late Start fields.
Late Start Type	Required if Late Start is enabled. Options: <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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.
Late Start Duration	If Late Start Type = 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.
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.

Finished Late	System-supplied; this field is flagged if the task finished later than the time or duration specified in the Late Finish fields.
Late Finish Type	Required if Late Finish is enabled. Options: <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 = Time; Time after which the task finish time is considered late. Use HH:MM, 24-hour time.
Late Finish Duration	If Late Finish Type = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
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.
Finished Early	System-supplied; this field is flagged if the task finished earlier than the time specified in the Early Finish fields.
Early Finish Type	Required if Early Finish is enabled. Options: <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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Statistics	This section contains time-related statistics for the task instance.
User Estimated End Time	System-supplied; if the user entered information into the User Estimated Duration field in the task Details, the Controller 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	System-supplied; shortest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Average Estimated End Time	System-supplied; average estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Longest Estimated End Time	System-supplied; longest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.

Buttons	This section identifies the buttons displayed above and below the Task Instance Details that let you perform various actions.
Update	Saves updates to the record.
Force Finish	See Force Finishing a Task .
Hold	Places the task instance on Hold (see Putting a Task on Hold).
Skip	For tasks loaded into the schedule that have not yet run; allows you to tell the Controller to skip this task. See Skipping a Task .
Re-run	See Re-running a Task .
View Parent	Displays the task instance Details for the parent Workflow of this task instance.
Delete	Deletes the current record.
Refresh	Refreshes any dynamic data displayed in the Details.
Close	For pop-up view only; closes the pop-up view of this task instance.
Tabs	This section identifies the tabs across the top of the Task Instance Details that provide access to additional information about the task instance.
Virtual Resources	Lists all Virtual Resources to which this task is assigned.
Exclusive Requests	Lists all records in the Exclusive Requests table (<code>ops_exclusive_order</code>) for this task instance.
Notes	Lists all notes associated with this record.

Running a Timer Task

You can run a Timer task:

- Manually, by clicking the [Launch Task](#) or [Launch Task with Variables](#) button in the Timer tasks list or Timer Task Details [Action menu](#).
- As part of a [workflow](#).
- [Specify triggers](#) that run the task automatically based on times or events.

Monitoring Task Execution

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

SQL Task

- Overview
- Built-In Variables
- Creating a SQL Task
 - SQL Task Details
 - SQL Task Details Field Descriptions
- Viewing a SQL Task Instance
 - SQL Task Instance Details
 - SQL Task Instance Details Field Descriptions
- Running a SQL Task
- 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 following [built-in variables](#) can be used in a SQL task to pass data where appropriate:

- Task and Task Instance variable
- SQL and Stored Procedure variables

Creating a SQL Task

Step 1 From the Automation Center navigation pane, select **Tasks > SQL Tasks**. The SQL Tasks list displays a list of all currently defined SQL tasks.

Below the list, SQL Task Details for a new SQL task displays. (You also can click the **New** button to display SQL Task Details for a new SQL task.)

The screenshot shows the 'SQL Tasks' interface. At the top, there is a list of 5 SQL tasks with columns for Task Name, Task Description, Updated By, and Updated. Below the list is the 'SQL Task Details' form, which is currently in a disabled state. The form has tabs for 'SQL Task', 'Variables', 'Actions', 'Virtual Resources', 'Mutually Exclusive', 'Instances', 'Triggers', 'Notes', and 'Versions'. The 'General' section includes fields for Task Name, Version (1), Task Description, Member of Business Services, Hold on Start, Virtual Resource Priority (10), and Hold Resources on Failure. The 'SQL Details' section includes Database Connection, Credentials, Database Connection Variable, Maximum Rows, Credentials Variable, and Auto Cleanup.

Step 2 Enter/select Details for a new SQL task, using the field descriptions below as a guide.

- Required fields display in **boldface**.
- Default values for fields, if available, display automatically.

To display more of the Details fields on the screen, you can temporarily [hide the list](#).



Note

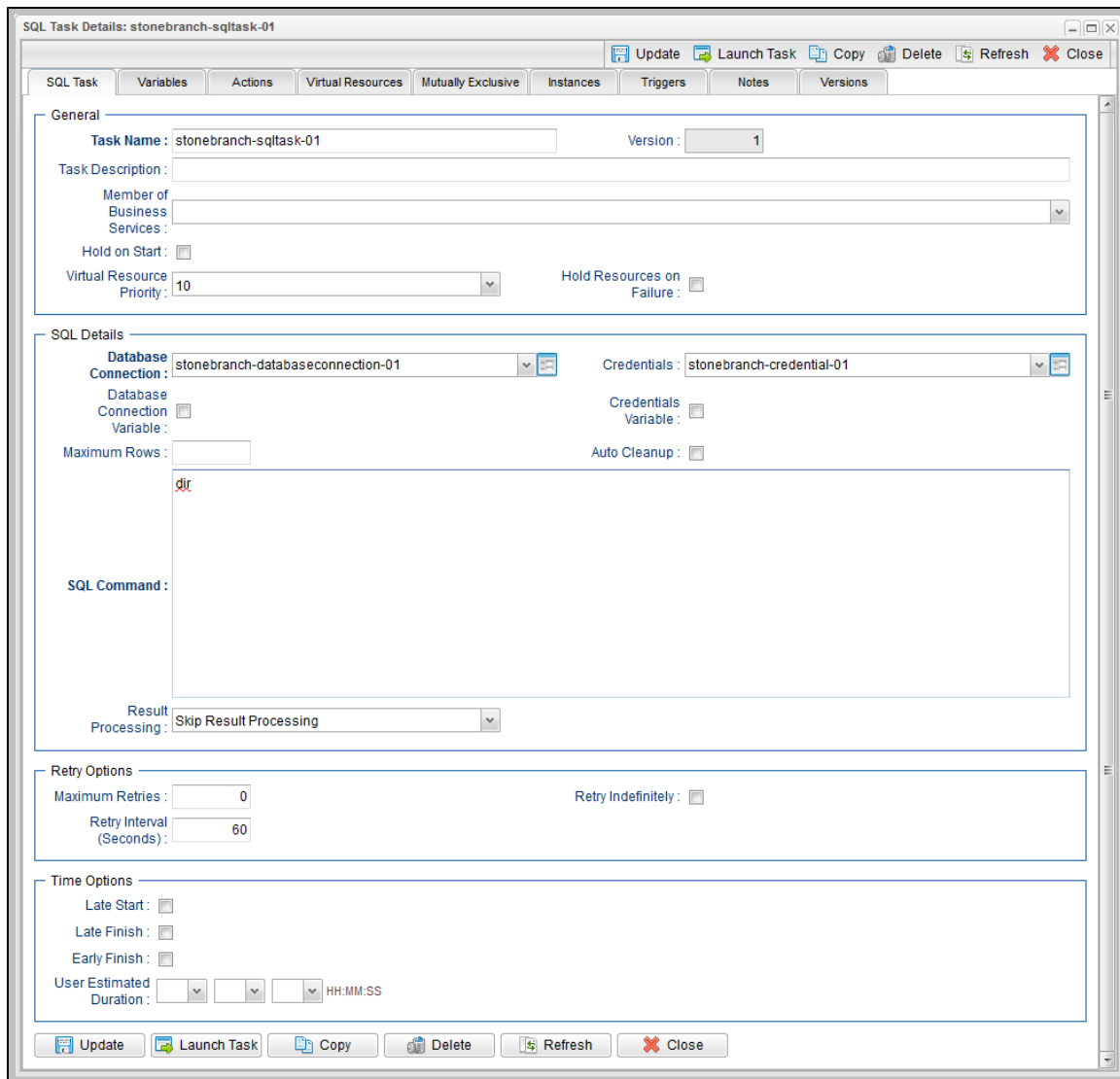
If you view [Windows Task Details](#) for an existing SQL task by clicking a task in the list, and then want to create a new SQL task, you must click the **New** button that displays above and below the Details.

Step 3 Click the **Save** button. The task is added to the database, and all buttons and tabs in the Task Details are enabled.

SQL Task Details

The following SQL Task Details is for an existing SQL task.


Depending on the values that you enter / select for these fields, and whether or not the SQL task has ever been launched, more (or less) fields may display. See the [field descriptions](#), below, for a description of all fields that may display in the SQL Task Details.



SQL Task Details Field Descriptions


The following table describes the fields, buttons, and tabs that display in the SQL Task Details.

Field Name	Description
General	This section contains general information about the task.
Task Name	User-defined name of this task, containing a maximum 255 alphanumeric characters; variables supported. It is the responsibility of the user to develop a workable naming scheme for tasks.
Version	System-supplied; version number of the current record, which is incremented by the Controller every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.

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.
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.
SQL Details	This section contains assorted detailed information about the task.
Database Connection	Name of the Opwise Controller database connection that defines the database. Select a database from the drop-down list or click the icon to create a new database connection.
Database Connection Variable	If enabled, the Database Connection field 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	If enabled, the Credentials field 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 . <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Credentials Variable when updating multiple tasks, the Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Credentials Unresolved - into which you enter a variable, variable and text, or Credentials name. The Credentials field is not used.</p> </div>
Maximum Rows	If necessary, specifies 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 .
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).
SQL Command	SQL command being executed against the database. Variables supported.

Result Processing	<p>Specifies how the Controller should determine whether the SQL command failed or completed successfully.</p> <p>Options:</p> <ul style="list-style-type: none"> • Skip Result Processing • Success Exitcode Range - The SQL command is considered completed successfully if its exit code falls within the range specified in the Exit Codes field. • Failure Exitcode Range - The SQL command is considered failed if its exit code falls within the range specified in the Exit Codes field. • 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). • 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).
Exit Codes	<p>Required if Result 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.</p>
Column Name	<p>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.</p>
Operator	<p>Operator being used for the comparison. Options: =, !=, >, >=, <, <=, regex.</p>
Value	<p>Value being compared, using the operator specified.</p>
Retry Options	<p>This section contains specifications for retrying the task.</p>
Maximum Retries	<p>User-defined; maximum number of times that the Controller should retry this task after it has started and gone to a failed state.</p>
Retry Indefinitely	<p>User-defined; indicates whether the Controller should continue trying indefinitely to run this task. If you enable this field, it overrides any value placed in the Maximum Retries field.</p>
Retry Interval (Seconds)	<p>User-defined; number of seconds between each retry.</p>
Time Options	<p>This section contains time-related specifications for the task.</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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.</p>
Late Start Type	<p>Required if Late Start is enabled.</p> <p>Options:</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>If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.</p>

Late Start Duration	<p>If Late Start Type = 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.</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>
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.</p> <p>Options:</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 = Time; 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 = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.</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.</p> <p>Options:</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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.</p>
User Estimated Duration	<p>Estimated amount of time it should normally take to run this task. The Controller uses this information to calculate the User Estimated End Time on a task instance record.</p>
Statistics	<p>This section contains time-related statistics for task instances of the task.</p>
First Time Ran	<p>System-supplied; date and time this task first ran.</p>

Last Time Ran	System-supplied; date and time the task last ran.
Last Instance Duration	System-supplied; amount of time the task took to run the last time it ran.
Lowest Instance Time	System-supplied; shortest amount of time this task has taken to run.
Average Instance Time	System-supplied; average amount of time this task takes to run.
Highest Instance Time	System-supplied; longest amount of time this task has taken to run.
Number of Instances	System-supplied; number of instances in the database for this task.
Buttons	This section identifies the buttons displayed above and below the Task Details that let you perform various actions.
Save	Saves a new task record in the Controller database.
Update button	Saves updates to the record.
New	Displays empty (except for default values) Details for creating a new task.
Launch Task	Manually launches the task.
View Parents	Displays a list of any parent Workflow tasks for this task.
Copy	Creates a copy of this task, which you are prompted to rename.
Delete	<p>Deletes the current record.</p> <div style="background-color: #ffffcc; padding: 10px; margin: 10px 0;"> <p> Note You cannot delete a task if it is either:</p> <ul style="list-style-type: none"> Specified in an enabled Trigger. The only task specified in a disabled Trigger. </div>
Refresh	Refreshes any dynamic data displayed in the Details.
Close	For pop-up view only; closes the pop-up view of this task.
Tabs	This section identifies the tabs across the top of the Task Details that provide access to additional information about the task.
Variables	Lists all variables associated with this record.

Actions	<p>Allows you to specify actions that the Controller will take automatically based on events that occur during the execution of this task.</p> <p>Events are:</p> <ul style="list-style-type: none"> • Task instance status • Exit codes • Late start • Late finish • Early finish <p>Actions are:</p> <table border="1" data-bbox="266 478 1484 840"> <tr> <td data-bbox="266 478 412 548">Abort Action</td> <td data-bbox="412 478 1484 548">Abort the task if certain events occur. For details, see Abort Actions.</td> </tr> <tr> <td data-bbox="266 548 412 617">Email Notification</td> <td data-bbox="412 548 1484 617">Send an email if certain events occur. For details, see Email Notification Actions.</td> </tr> <tr> <td data-bbox="266 617 412 686">Set Variable</td> <td data-bbox="412 617 1484 686">Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow.</td> </tr> <tr> <td data-bbox="266 686 412 756">SNMP Notification</td> <td data-bbox="412 686 1484 756">Send an email if certain events occur. For details, see SNMP Notification Actions.</td> </tr> <tr> <td data-bbox="266 756 412 825">System Operation</td> <td data-bbox="412 756 1484 825">Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions.</td> </tr> </table>	Abort Action	Abort the task if certain events occur. For details, see Abort Actions .	Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .	Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .	SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .	System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .
Abort Action	Abort the task if certain events occur. For details, see Abort Actions .										
Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .										
Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .										
SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .										
System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .										
Virtual Resources	Lists all Virtual Resources to which this task is assigned.										
Mutually Exclusive	Lists all tasks that have been set to be mutually exclusive of this task.										
Instances	Lists all instances of the task.										
Triggers	Lists 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, the Controller 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 Triggers Overview .										
Notes	Lists all notes associated with this record.										
Versions	Stores copies of all previous versions of the current record. See Record Versioning .										

Viewing a SQL Task Instance

When a SQL task is launched, the Controller creates a task instance record of that task.

A task instance contains detailed information about a single execution of that task.

You can access a task instance from:

- **Instances tab** on the [SQL Task Details](#) for that task
- [Activity Monitor](#)
- [Task Instances list](#)

SQL Task Instance Details

The following SQL Task Instance Details contains information on the execution of a SQL task.

The screenshot shows the 'SQL Task Instance Details' window for a task named 'stonebranch-sqltask-01'. The window is divided into several sections:

- General:** Instance Name (stonebranch-sqltask-01), Reference Id (2), Task (stonebranch-sqltask-01), Invoked By (Manually Launched), Task Description, Member of Business Services, Execution User (stonebranch-user-01), Virtual Resource Priority (10), and Hold Resources on Failure (checkbox).
- Status:** Status (Success), Status Description, Queued Time (2015-02-06 14:45:37 -0500), Trigger Time, Start Time (2015-02-06 14:45:41 -0500), Launch Time (2015-02-06 14:45:37 -0500), End Time (2014-06-20 13:09:49 -0400), Duration (0 Seconds), SQL State, Rows Retrieved (150), and SQL Error Message.
- SQL Details:** Database Connection (stonebranch-databaseconnection-01), Credentials (stonebranch-credential-02), Database Connection Variable (checkbox), Credentials Variable (checkbox), Maximum Rows, Auto Cleanup (checkbox), SQL Command (select * from task_rstrt_criteria;), and Result Processing (Skip Result Processing).
- Retry Options:** Maximum Retries (0), Retry Interval (Seconds) (60), Retry Indefinitely (checkbox), and Current Retry Count (0).
- Statistics:** User Estimated End Time, Average Estimated End Time (2014-06-15 11:13:53 -0400), Shortest Estimated End Time, and Longest Estimated End Time.


At the top and bottom of the window are buttons for Update, Re-run, Delete, Refresh, and Close.

SQL Task Instance Details Field Descriptions

The following table describes the fields, buttons, and tabs that display in the SQL Task Instance Details.

Field Name	Description
General	This section contains general information about the task instance.
Instance Name	Name of this task instance.
Reference Id	System-supplied; the Controller increments this number each time the task is run.

Task	Name of the task that was run to create this task instance. Click the icon to display Task Details for the task.
Invoked by	System-supplied; how the task instance was launched. Options: <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 identify the user, check the Execution User column for that task instance on the Task Instances screen or, on most task instance screens, the Execution User field.
Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
Execution User	System-supplied; if the task was launched manually, the ID of the user who launched it.
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.
Status	This section contains information about the current status of the task instance.
Status	System-supplied; see Task Instance Statuses .
Exit Code	System-supplied; the exit code captured by the Agent when executing the task (for example, a command or script).
Status Description	System-supplied; additional information, if any, about the status of the task instance.
Queued Time	System-supplied; Date and time the task was queued for processing.
Trigger Time	System-supplied; Date and time the task instance was triggered.
Launch Time	System-supplied; Date and time the task instance was launched.
Start Time	System-supplied; Date and time the task instance started.
End Time	System-supplied; Date and time the task instance completed.

Duration	System-supplied; amount of time the task instance took to run.
SQL State	System-supplied; resolves to a return code that indicates the outcome of the most recently executed SQL statement.
Rows Retrieved	System-supplied; number of rows retrieved by the SQL procedure.
SQL Error Message	System-supplied; any error messages returned by the SQL procedure.
SQL Details	This section contains assorted detailed information about the task instance.
Database Connection	Name of the Opwise Controller database connection that defines the database. Select a database from the drop-down list or click the icon to create a new database connection.
Database Connection Variable	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: \${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	If enabled, the Credentials field 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 . <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note If you select Credentials Variable when updating multiple tasks, the Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Credentials Unresolved - into which you enter a variable, variable and text, or Credentials name. The Credentials field is not used.</p> </div>
Maximum Rows	If necessary, specifies 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 .
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).
SQL Command	SQL command being executed against the database. Variables supported.

Result Processing	<p>Specifies how the Controller should determine whether the SQL command failed or completed successfully.</p> <p>Options:</p> <ul style="list-style-type: none"> • Skip Result Processing • Success Exitcode Range - The SQL command is considered completed successfully if its exit code falls within the range specified in the Exit Codes field. • Failure Exitcode Range - The SQL command is considered failed if its exit code falls within the range specified in the Exit Codes field. • 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). • 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).
Exit Codes	<p>Required if Result 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.</p>
Column Name	<p>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.</p>
Operator	<p>Operator being used for the comparison. Options: =, !=, >, >=, <, <=, regex.</p>
Value	<p>Value being compared, using the operator specified.</p>
Retry Options	<p>This section contains specifications for retrying the task.</p>
Maximum Retries	<p>User-defined; maximum number of times that the Controller should retry this task after it has started and gone to a failed state.</p>
Retry Indefinitely	<p>User-defined; indicates whether the Controller should continue trying indefinitely to run this task. If you enable this field, it overrides any value placed in the Maximum Retries field.</p>
Retry Interval (Seconds)	<p>User-defined; number of seconds between each retry.</p>
Current Retry Count	<p>System-supplied; current number of times that the Controller has retried the task after it first went to failure status.</p>
Time Options	<p>This section contains time-related specifications for the task instance.</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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.</p>
Started Late	<p>System-supplied; this field is flagged if the task started later than the time specified in the Late Start fields.</p>

Late Start Type	<p>Required if Late Start is enabled.</p> <p>Options:</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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.
Late Start Duration	<p>If Late Start Type = 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.</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>
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.
Finished Late	System-supplied; this field is flagged if the task finished later than the time or duration specified in the Late Finish fields.
Late Finish Type	<p>Required if Late Finish is enabled.</p> <p>Options:</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 = Time; Time after which the task finish time is considered late. Use HH:MM, 24-hour time.
Late Finish Duration	If Late Finish Type = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
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.
Finished Early	System-supplied; this field is flagged if the task finished earlier than the time specified in the Early Finish fields.
Early Finish Type	<p>Required if Early Finish is enabled.</p> <p>Options:</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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Statistics	This section contains time-related statistics for the task instance.
User Estimated End Time	System-supplied; if the user entered information into the User Estimated Duration field in the task Details, the Controller uses this information to calculate an end time for the task instance, based on the date/time the task instance started.
Average Estimated End Time	System-supplied; average estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Shortest Estimated End Time	System-supplied; shortest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Longest Estimated End Time	System-supplied; longest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Buttons	This section identifies the buttons displayed above and below the Task Instance Details that let you perform various actions.
Update	Saves updates to the record.
Force Finish	See Force Finishing a Task .
Hold	Places the task instance on Hold (see Putting a Task on Hold).
Skip	For tasks loaded into the schedule that have not yet run; allows you to tell the Controller to skip this task. See Skipping a Task .
Re-run	See Re-running a Task .
View Parent	Displays the task instance Details for the parent Workflow of this task instance.
Retrieve Output	See Retrieving Output .
Refresh	Refreshes any dynamic data displayed in the Details.
Close	For pop-up view only; closes the pop-up view of this task instance.
Tabs	This section identifies the tabs across the top of the Task Instance Details that provide access to additional information about the task instance.
SQL Results	Stores results of executed SQL statements, if any.
SQL Warnings	Warnings returned by executed SQL statements, if any.

Virtual Resources	Lists all Virtual Resources to which this task is assigned.
Exclusive Requests	Lists all records in the Exclusive Requests table (<code>ops_exclusive_order</code>) for this task instance.
Notes	Lists all notes associated with this record.

Running a SQL Task

You can run a SQL task:

- Manually, by clicking the [Launch Task](#) or [Launch Task with Variables](#) button in the SQL tasks list or SQL Task Details [Action](#) menu.
- As part of a [workflow](#).
- [Specify triggers](#) that run the task automatically based on times or events.

Monitoring Task Execution

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

Stored Procedure Task

- Overview
- Built-In Variables
- Creating a Stored Procedure Task
 - Stored Procedure Task Details
 - Stored Procedure Task Details Field Descriptions
- Viewing a Stored Procedure Task Instance
 - Stored Procedure Task Instance Details
 - Stored Procedure Task Instance Details Field Descriptions
- Adding Stored Procedure Parameters
 - Adding a Parameter
 - Stored Procedure Parameter Field Descriptions
 - Deleting a Parameter
- Running a Stored Procedure Task
- 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 following [built-in variables](#) 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 Stored Procedure Task

Step 1 From the [Automation Center](#) navigation pane, select **Tasks > Stored Procedure Tasks**. The Stored Procedure Tasks list displays a list of all currently defined Stored Procedure tasks.

Below the list, Stored Procedure Task Details for a new Stored Procedure task displays. (You also can click the **New** button to display Stored Procedure Task Details for a new Stored Procedure task.)

The screenshot displays the 'Stored Procedure Tasks' interface. At the top, there is a list of five tasks, each with a task name, description, updated by user, and updated timestamp. Below the list, the 'Stored Procedure Task Details' form is shown, which includes tabs for 'General', 'Stored Procedure Parameters', 'Variables', 'Actions', 'Virtual Resources', 'Mutually Exclusive', 'Instances', 'Triggers', 'Notes', and 'Versions'. The 'General' tab is active, showing fields for 'Task Name', 'Version', 'Task Description', 'Member of Business Services', 'Hold on Start', 'Virtual Resource Priority', 'Hold Resources on Failure', 'Database Connection', 'Credentials', 'Database Connection Variable', 'Maximum Rows', and 'Auto Cleanup'.

Task Name	Task Description	Updated By	Updated
stonebranch-storedproceduretask-01		ops.admin	2014-06-13 14:15:21 -0400
stonebranch-storedproceduretask-02		ops.admin	2014-06-13 14:15:30 -0400
stonebranch-storedproceduretask-03		ops.admin	2014-06-13 14:15:34 -0400
stonebranch-storedproceduretask-04		ops.admin	2014-06-13 14:15:38 -0400
stonebranch-storedproceduretask-05		ops.admin	2014-06-13 14:15:43 -0400

Step 2 Enter/select Details for a new Stored Procedure task, using the field descriptions below as a guide.

- Required fields display in **boldface**.
- Default values for fields, if available, display automatically.

To display more of the Details fields on the screen, you can temporarily [hide the list](#).



Note

If you view [Stored Procedure Task Details](#) for an existing Stored Procedure task by clicking a task in the list, and then want to create a new Stored Procedure task, you must click the **New** button that displays above and below the Details.

Step 3 Click the **Save** button. The task is added to the database, and all buttons and tabs in the Task Details are enabled.

Stored Procedure Task Details


The following Stored Procedure Task Details is for an existing Stored Procedure task.

Depending on the values that you enter / select for these fields, and whether or not the Stored Procedure task has ever been launched, more (or less) fields may display. See the [field descriptions](#), below, for a description of all fields that may display in the Stored Procedure Task Details.

Stored Procedure Task Details Field Descriptions


The following table describes the fields, buttons, and tabs that display in the Stored Procedure Task Details.

Field Name	Description
General	This section contains general information about the task.
Task Name	User-defined name of this task, containing a maximum 255 alphanumeric characters; variables supported. It is the responsibility of the user to develop a workable naming scheme for tasks.
Version	System-supplied; version number of the current record, which is incremented by the Controller every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
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.

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	<p>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.</p>
Stored Procedure Details	<p>This section contains assorted detailed information about the task.</p>
Database Connection	<p>Name of the Opwise Controller database connection that defines the database. Select a database from the drop-down list or click the icon to create a new database connection.</p>
Database Connection Variable	<p>If enabled, the Database Connection field 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>
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>
Credentials Variable	<p>If enabled, the Credentials field 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> <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note If you select Credentials Variable when updating multiple tasks, the Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Credentials Unresolved - into which you enter a variable, variable and text, or Credentials name. The Credentials field is not used.</p> </div>
Maximum Rows	<p>If necessary, specifies 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.</p>
Auto Cleanup	<p>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).</p>
Stored Procedure Name	<p>Name of the file containing the stored procedure being executed against the database. Variables supported.</p>

Result Processing	<p>Specifies how the Controller should determine whether the Stored Procedure failed or completed successfully.</p> <p>Options:</p> <ul style="list-style-type: none"> • Skip Result Processing. • Success Exitcode Range - The Stored Procedure is considered completed successfully if its exit code falls within the range specified in the Exit Codes field. • Failure Exitcode Range - The Stored Procedure is considered failed if its exit code falls within the range specified in the Exit Codes field. • 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). • 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). • 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.
Exit Codes	Required if Result 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.
Parameter Position	If Result Processing = Success Output Parameter or Failure Output Parameter 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.
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.
Retry Options	This section contains specifications for retrying the task.
Maximum Retries	User-defined; maximum number of times that the Controller should retry this task after it has started and gone to a failed state.
Retry Indefinitely	User-defined; indicates whether the Controller should continue trying indefinitely to run this task. If you enable this field, it overrides any value placed in the Maximum Retries field.
Retry Interval (Seconds)	User-defined; number of seconds between each retry.
Time Options	This section contains time-related specifications for the task.
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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.

Late Start Type	<p>Required if Late Start is enabled.</p> <p>Options:</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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.
Late Start Duration	<p>If Late Start Type = 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.</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>
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.</p> <p>Options:</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 = Time; Time after which the task finish time is considered late. Use HH:MM, 24-hour time.
Late Finish Duration	If Late Finish Type = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
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.</p> <p>Options:</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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
User Estimated Duration	Estimated amount of time it should normally take to run this task. The Controller uses this information to calculate the User Estimated End Time on a task instance record.
Statistics	This section contains time-related statistics for task instances of the task.
First Time Ran	System-supplied; date and time this task first ran.
Last Time Ran	System-supplied; date and time the task last ran.
Last Instance Duration	System-supplied; amount of time the task took to run the last time it ran.
Lowest Instance Time	System-supplied; shortest amount of time this task has taken to run.
Average Instance Time	System-supplied; average amount of time this task takes to run.
Highest Instance Time	System-supplied; longest amount of time this task has taken to run.
Number of Instances	System-supplied; number of instances in the database for this task.
Buttons	This section identifies the buttons displayed above and below the Task Details that let you perform various actions.
Save	Saves a new task record in the Controller database.
Update	Saves updates to the record.
New	Displays empty (except for default values) Details for creating a new task.
Launch Task	Manually launches the task.
View Parents	Displays a list of any parent Workflow tasks for this task.
Copy	Creates a copy of this task, which you are prompted to rename.
Delete	<p>Deletes the current record.</p> <div style="background-color: #ffffcc; padding: 10px; margin: 10px 0;"> <p> Note You cannot delete a task if it is either:</p> <ul style="list-style-type: none"> Specified in an enabled Trigger. The only task specified in a disabled Trigger. </div>
Refresh	Refreshes any dynamic data displayed in the Details.
Close	For pop-up view only; closes the pop-up view of this task.

Tabs	This section identifies the tabs across the top of the Task Details that provide access to additional information about the task instance.										
Stored Procedure Parameters	See Adding Stored Procedure Parameters , below.										
Variables	Lists all variables associated with this record.										
Actions	<p>Allows you to specify actions that the Controller will take automatically based on events that occur during the execution of this task.</p> <p>Events are:</p> <ul style="list-style-type: none"> • Task instance status • Exit codes • Late start • Late finish • Early finish <p>Actions are:</p> <table border="1"> <tr> <td>Abort Action</td> <td>Abort the task if certain events occur. For details, see Abort Actions.</td> </tr> <tr> <td>Email Notification</td> <td>Send an email if certain events occur. For details, see Email Notification Actions.</td> </tr> <tr> <td>Set Variable</td> <td>Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow.</td> </tr> <tr> <td>SNMP Notification</td> <td>Send an email if certain events occur. For details, see SNMP Notification Actions.</td> </tr> <tr> <td>System Operation</td> <td>Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions.</td> </tr> </table>	Abort Action	Abort the task if certain events occur. For details, see Abort Actions .	Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .	Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .	SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .	System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .
Abort Action	Abort the task if certain events occur. For details, see Abort Actions .										
Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .										
Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .										
SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .										
System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .										
Virtual Resources	Lists all Virtual Resources to which this task is assigned.										
Mutually Exclusive	Lists all tasks that have been set to be mutually exclusive of this task.										
Instances	Lists all instances of the task.										
Triggers	Lists 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, the Controller 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 Triggers Overview .										
Notes	Lists all notes associated with this record.										
Versions	Stores copies of all previous versions of the current record. See Record Versioning .										

Viewing a Stored Procedure Task Instance

When a Stored Procedure task is launched, the Controller creates a task instance record of that task.

A task instance contains detailed information about a single execution of that task.

You can access a task instance from:

- **Instances tab** on the [Stored Procedure Task Details](#) for that task
- [Activity Monitor](#)
- [Task Instances list](#)

Stored Procedure Task Instance Details


The following Stored Procedure Task Instance Details contains information on the execution of the task shown in the [Stored Procedure Task Details](#).

Stored Procedure Task Instance Details Field Descriptions

The following table describes the fields, buttons, and tabs that display in [Stored Procedure Task Instance Details](#).

Field Name	Description
General	This section contains general information about the task instance.
Instance Name	Name of this task instance.

Reference Id	System-supplied; the Controller increments this number each time the task is run.
Task	Name of the task that was run to create this task instance. Click the icon to display Task Details for the task.
Invoked by	System-supplied; how the task instance was launched. Options: <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 identify the user, check the Execution User column for that task instance on the Task Instances screen or, on most task instance screens, the Execution User field.
Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
Execution User	System-supplied; if the task was launched manually, the ID of the user who launched it.
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.
Status	This section contains information about the current status of the task instance.
Status	System-supplied; see Task Instance Statuses .
Status Description	System-supplied; additional information, if any, about the status of the task instance.
Queued Time	System-supplied; Date and time the task was queued for processing.
Trigger Time	System-supplied; Date and time the task instance was triggered.
Launch Time	System-supplied; Date and time the task instance was launched.
Start Time	System-supplied; Date and time the task instance started.
End Time	System-supplied; Date and time the task instance completed.

Duration	System-supplied; amount of time the task instance took to run.
SQL State	System-supplied; resolves to a return code that indicates the outcome of the most recently executed SQL statement.
Rows Retrieved	System-supplied; number of rows retrieved by the SQL procedure.
SQL Error Message	System-supplied; any error messages returned by the SQL procedure.
Stored Procedure Details	This section contains assorted detailed information about the task instance.
Database Connection	Name of the Opwise Controller database connection that defines the database. Select a database from the drop-down list or click the icon to create a new database connection.
Database Connection Variable	If enabled, the Database Connection field 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	If enabled, the Credentials field 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 . <div data-bbox="315 1087 1446 1270" style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note If you select Credentials Variable when updating multiple tasks, the Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Credentials Unresolved - into which you enter a variable, variable and text, or Credentials name. The Credentials field is not used.</p> </div>
Maximum Rows	If necessary, specifies 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 .
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).
Stored Procedure Name	Name of the file containing the stored procedure being executed against the database. Variables supported.

Result Processing	<p>Specifies how the Controller should determine whether the Stored Procedure failed or completed successfully.</p> <p>Options:</p> <ul style="list-style-type: none"> • Skip Result Processing. • Success Exitcode Range - The Stored Procedure is considered completed successfully if its exit code falls within the range specified in the Exit Codes field. • Failure Exitcode Range - The Stored Procedure is considered failed if its exit code falls within the range specified in the Exit Codes field. • 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). • 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). • 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	If Result Processing = Success Output Parameter or Failure Output Parameter 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	Required if Result 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.
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.
Retry Options	This section contains specifications for retrying the task.
Maximum Retries	User-defined; maximum number of times that the Controller should retry this task after it has started and gone to a failed state.
Retry Indefinitely	User-defined; indicates whether the Controller should continue trying indefinitely to run this task. If you enable this field, it overrides any value placed in the Maximum Retries field.
Retry Interval (Seconds)	User-defined; number of seconds between each retry.
Current Retry Count	System-supplied; current number of times that the Controller has retried the task after it first went to failure status.
Time Options	This section contains time-related specifications for the task instance.

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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.
Started Late	System-supplied; this field is flagged if the task started later than the time specified in the Late Start fields.
Late Start Type	Required if Late Start is enabled. Options: <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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.
Late Start Duration	If Late Start Type = 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.
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.
Finished Late	System-supplied; this field is flagged if the task finished later than the time or duration specified in the Late Finish fields.
Late Finish Type	Required if Late Finish is enabled. Options: <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 = Time; Time after which the task finish time is considered late. Use HH:MM, 24-hour time.
Late Finish Duration	If Late Finish Type = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
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.

Finished Early	System-supplied; this field is flagged if the task finished earlier than the time specified in the Early Finish fields.
Early Finish Type	Required if Early Finish is enabled. Options: <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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Statistics	This section contains time-related statistics for the task instance.
User Estimated End Time	System-supplied; if the user entered information into the User Estimated Duration field in the task Details, the Controller uses this information to calculate an end time for the task instance, based on the date/time the task instance started.
Average Estimated End Time	System-supplied; average estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Shortest Estimated End Time	System-supplied; shortest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Longest Estimated End Time	System-supplied; longest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Buttons	This section identifies the buttons displayed above and below the Task Instance Details that let you perform various actions.
Update	Saves updates to the record.
Force Finish	See Force Finishing a Task .
Hold	Places the task instance on Hold (see Putting a Task on Hold).
Skip	For tasks loaded into the schedule that have not yet run; allows you to tell the Controller to skip this task. See Skipping a Task .
Re-run	See Re-running a Task .
View Parent	Displays the task instance Details for the parent Workflow of this task instance.
Retrieve Output	See Retrieving Output .

Delete	Deletes the current record.
Refresh	Refreshes any dynamic data displayed in the Details.
Close	For pop-up view only; closes the pop-up view of this task instance.
Tabs	This section identifies the tabs across the top of the Task Instance Details that provide access to additional information about the task instance.
Stored Procedure Parameters	See Adding Stored Procedure Parameters , below.
SQL Results	Stores results of executed SQL statements, if any.
SQL Warnings	Warnings returned by executed SQL statements, if any.
Virtual Resources	Lists all Virtual Resources to which this task is assigned.
Exclusive Requests	Lists all records in the Exclusive Requests table (<code>ops_exclusive_order</code>) for this task instance.
Output	Displays output generated from the process, if any, based on specifications provided by the user in the Automatic Output Retrieval fields in the task Details. If automatic output retrieval was not available or was not selected, output can be obtained by clicking the Retrieve Output button.
Notes	Lists all notes associated with this record.

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 the Stored Procedure Parameters tab. The Stored Procedure Parameters list displays a list of all currently defined Stored Procedure parameters.

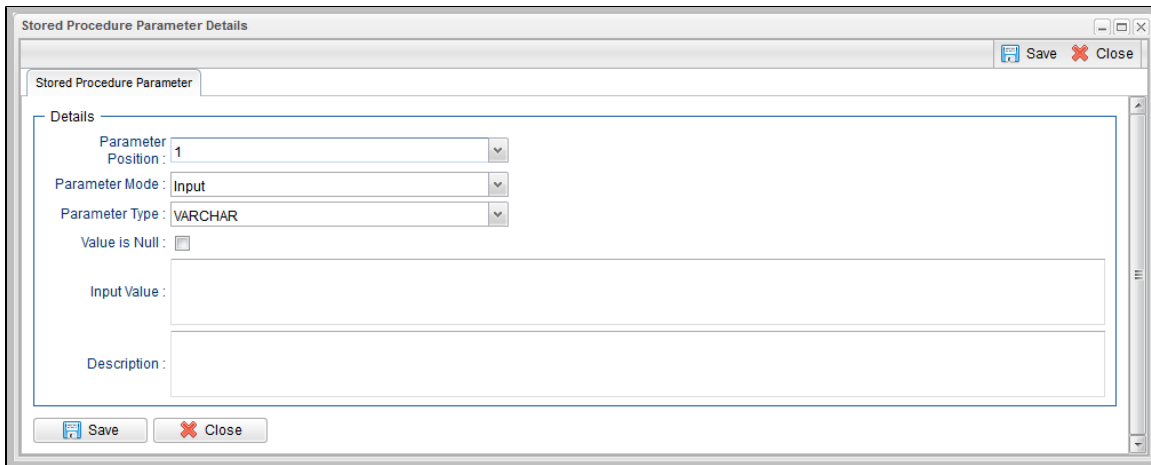
Stored Procedure Task Details: stonebranch-storedprocedurerecord-01

Stored Procedure Task | **Stored Procedure Parameters** | Variables | Actions | Virtual Resources | Mutually Exclusive | Instances | Triggers | Notes | Versions

New ↻

Parameter Position	Parameter Mode	Parameter Type	Input Value	Variable Name	Description	Updated By	Updated
1	Input	VARCHAR	123			ops.admin	2014-06-27 10:45:24 -0400
2	Input	TIME	00:10			ops.admin	2014-06-27 10:45:57 -0400

Step 3 Click the **New** button to display Stored Procedures Parameter Details for a new parameter.



Step 4 Use the field descriptions, below, to define the parameter.

Step 5 Click the **Save** button.

Stored Procedure Parameter Field Descriptions

Field Name	Description
Parameter Position	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"> • NUMERIC • DECIMAL • INTEGER • SMALLINT • FLOAT • REAL • DOUBLE • VARCHAR • BOOLEAN • DATE • TIME • TIMESTAMP • BIGINT • VARBINARY
Value is Null	If Parameter Mode = Input or Input/Output; 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	If Parameter Mode = Input or Input/Output; Input value of the parameter, if any.
Description	Description of this parameter.

Variable Scope	<p>If Parameter Mode = Output or Input/Output; applies to parameters associated with a task in a workflow. Scope of the variable to assign the Output value.</p> <p>Options:</p> <ul style="list-style-type: none"> • SELF • PARENT • TOP_LEVEL_PARENT
Variable Name	If Parameter Mode = Output or Input/Output; name of variable to assign the Output value.

Deleting a Parameter

To delete a single parameter, either:

- Right-click the parameter on the Stored Procedure Parameters list and click **Delete** on the [Action menu](#).
- Open the Stored Procedure Parameter record and click the **Delete** button.

Running a Stored Procedure Task

You can run a Stored Procedure task:

- Manually, by clicking the [Launch Task](#) or [Launch Task with Variables](#) button in the Stored Procedure tasks list or Stored Procedure Task Details [Action menu](#).
- As part of a [workflow](#).
- [Specify triggers](#) that run the task automatically based on times or events.

Monitoring Task Execution

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

Email Task

- Overview
- Creating an Email Task
 - Email Task Details
 - Email Task Details Field Descriptions
- Viewing an Email Task Instance
 - Email Task Instance Details
 - Email Task Instance Details Field Descriptions
- Running an Email Task
- 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 an Email Task

Step 1 From the Automation Center navigation pane, select **Tasks > Email Tasks**. The Email Tasks list displays a list of all currently defined Email tasks.

Below the list, Email Task Details for a new Email task displays. (You also can click the **New** button to display Email Task Details for a new Email task.)

The screenshot shows the 'Email Tasks' interface. At the top, there is a list of 5 Email Tasks with columns for Task Name, Task Description, Updated By, and Updated. Below the list, the 'Email Task Details' form is displayed, showing fields for Task Name, Task Description, Member of Business Services, Hold on Start, Virtual Resource Priority, Hold Resources on Failure, Email Template, Reply-To, and To.

Task Name	Task Description	Updated By	Updated
stonebranch-emailtask-01		ops.admin	2014-06-13 14:17:06 -0400
stonebranch-emailtask-02		ops.admin	2014-06-13 14:17:19 -0400
stonebranch-emailtask-03		ops.admin	2014-06-13 14:17:24 -0400
stonebranch-emailtask-04		ops.admin	2014-06-13 14:17:27 -0400
stonebranch-emailtask-05		ops.admin	2014-06-13 14:17:32 -0400

Email Task Details

General

Task Name : Version :

Task Description :

Member of Business Services :

Hold on Start :


Virtual Resource Priority : Hold Resources on Failure :

Email Details

Email Template : Email Connection :

Reply-To :

To :

Step 2	<p>Enter/select Details for a new Email task, using the field descriptions below as a guide.</p> <ul style="list-style-type: none"> • Required fields display in boldface. • Default values for fields, if available, display automatically. <p>To display more of the Details fields on the screen, you can temporarily hide the list.</p> <div style="background-color: #ffffcc; padding: 10px; margin: 10px 0;"> <p> Note If you view Email Task Details for an existing Email task by clicking a task in the list, and then want to create a new Email task, you must click the New button that displays above and below the Details.</p> </div>
Step 3	<p>Click the Save button. The task is added to the database, and all buttons and tabs in the Task Details are enabled.</p>

Email Task Details

The following Email Task Details is for an existing Email task.

Depending on the values that you enter / select for these fields, and whether or not the Email task has ever been launched, more (or less) fields may display. See the [field descriptions](#), below, for a description of all fields that may display in the Email Task Details.

The screenshot shows the 'Email Task Details' window for 'stonebranch-emailtask-01'. The interface includes several tabs: 'Email Task', 'Variables', 'Actions', 'Virtual Resources', 'Mutually Exclusive', 'Instances', 'Triggers', 'Notes', and 'Versions'. The 'General' section contains fields for 'Task Name' (stonebranch-emailtask-01), 'Version' (1), 'Task Description' (email to stonebranch), 'Member of Business Services' (stonebranchbusinessservice 01), 'Hold on Start' (checkbox), 'Virtual Resource Priority' (10), and 'Hold Resources on Failure' (checkbox). The 'Email Details' section includes 'Email Template' (stonebranch-emailtemplate-01), 'Email Connection' (stonebranch-emailconnection-01), and fields for 'Reply-To', 'To' (stonebranch@email.com), 'Cc', 'Bcc', 'Subject', and 'Body'. The 'Time Options' section has checkboxes for 'Late Start', 'Late Finish', and 'Early Finish', along with a 'User Estimated Duration' field in HH:MM:SS format. At the bottom, there are buttons for 'Update', 'Launch Task', 'View Parents', 'Copy', 'Delete', 'Refresh', and 'Close'.


Email Task Details Field Descriptions

The following table describes the fields, buttons, and tabs that display in the Email Task Details.

Field Name	Description
General	This section contains general information about the task.
Task Name	User-defined name of this task, containing a maximum 255 alphanumeric characters; variables supported. It is the responsibility of the user to develop a workable naming scheme for tasks.
Version	System-supplied; version number of the current record, which is incremented by the Controller every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
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.
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.
Email Details	This section contains assorted detailed information about the task.
Email Template	Name of an Email template defined in an Email Template Details. An Email template allows you to specify standard recipients and text for outgoing emails. Enter the name of an existing Email template, select an Email template from the drop-down list, or click the Details icon to create a new Email template. Every Email template specifies an Email connection. If you do not specify an Email template in this field, you must specify an Email connection in the Email Connection field. If you specify both an Email template (in this field) and an Email Connection , the Email server specified in the Email Connection field overrides the Email server specified in this field. (Any information specified in an Email task overrides what is specified in an Email template.)
Email Connection	Required if an Email template is not specified in the Email Template field; name of an Email connection defined via the Email Connection definition screen. An email connection specifies information about an email server. Enter the name of an existing Email connection, or click the magnifying glass to browse for an existing Email connection or create a new Email connection. If you specify both an Email Template and an Email connection (in this field), the Email connection specified in this field overrides the Email connection specified in the Email Template field.
Reply-To	Email address of the sender. Use commas to separate multiple recipients. Variables and functions supported.

To	Email address of the recipient. Use commas to separate multiple recipients. Variables and functions supported.
CC	Email address of the party being sent a copy of the email, if any. Use commas to separate multiple recipients. Variables and functions supported.
BCC	Email address of the party being sent a blind (hidden) copy of the email, if any. Use commas to separate multiple recipients. Variables and functions supported.
Subject	Subject line of the email. Variables and functions supported.
Body	Text of the email message. Variables and functions supported. If both the email template and the email task contain text in the body, the text is appended.
Time Options	This section contains time-related specifications for the task.
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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.
Late Start Type	Required if Late Start is enabled. Options: <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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.
Late Start Duration	If Late Start Type = 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.
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: <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 = Time; Time after which the task finish time is considered late. Use HH:MM, 24-hour time.
Late Finish Duration	If Late Finish Type = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
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: <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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
User Estimated Duration	Estimated amount of time it should normally take to run this task. The Controller uses this information to calculate the User Estimated End Time on a task instance record.
Statistics	This section contains time-related statistics for task instances of the task.
First Time Ran	System-supplied; date and time this task first ran.
Last Time Ran	System-supplied; date and time the task last ran.
Last Instance Duration	System-supplied; amount of time the task took to run the last time it ran.
Lowest Instance Time	System-supplied; shortest amount of time this task has taken to run.
Average Instance Time	System-supplied; average amount of time this task takes to run.
Highest Instance Time	System-supplied; longest amount of time this task has taken to run.
Number of Instances	System-supplied; number of instances in the database for this task.
Buttons	This section identifies the buttons displayed above and below the Task Details that let you perform various actions.
Save	Saves a new Linux/Unix task record in the Controller database.

Update	Saves updates to the record.										
New	Displays empty (except for default values) Details for creating a new task.										
Launch Task	Manually launches the task.										
View Parents	Displays a list of any parent Workflow tasks for this task.										
Copy	Creates a copy of this task, which you are prompted to rename.										
Delete	<p>Deletes the current record.</p> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note You cannot delete a task if it is either:</p> <ul style="list-style-type: none"> Specified in an enabled Trigger. The only task specified in a disabled Trigger. </div>										
Refresh	Refreshes any dynamic data displayed in the Details.										
Close	For pop-up view only; closes the pop-up view of this task.										
Tabs	This section identifies the tabs across the top of the Task Details that provide access to additional information about the task.										
Variables	Lists all variables associated with this record.										
Actions	<p>Allows you to specify actions that the Controller will take automatically based on events that occur during the execution of this task.</p> <p>Events are:</p> <ul style="list-style-type: none"> Task instance status Exit codes Late start Late finish Early finish <p>Actions are:</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 20%;">Abort Action</td> <td>Abort the task if certain events occur. For details, see Abort Actions.</td> </tr> <tr> <td>Email Notification</td> <td>Send an email if certain events occur. For details, see Email Notification Actions.</td> </tr> <tr> <td>Set Variable</td> <td>Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow.</td> </tr> <tr> <td>SNMP Notification</td> <td>Send an email if certain events occur. For details, see SNMP Notification Actions.</td> </tr> <tr> <td>System Operation</td> <td>Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions.</td> </tr> </table>	Abort Action	Abort the task if certain events occur. For details, see Abort Actions .	Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .	Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .	SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .	System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .
Abort Action	Abort the task if certain events occur. For details, see Abort Actions .										
Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .										
Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .										
SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .										
System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .										
Virtual Resources	Lists all Virtual Resources to which this task is assigned.										

Mutually Exclusive	Lists all tasks that have been set to be mutually exclusive of this task.
Instances	Lists all instances of the task.
Triggers	Lists 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, the Controller 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 Triggers Overview .
Notes	Lists all notes associated with this record.
Versions	Stores copies of all previous versions of the current record. See Record Versioning .

Viewing an Email Task Instance

When an Email task is launched, the Controller creates a task instance record of that task.

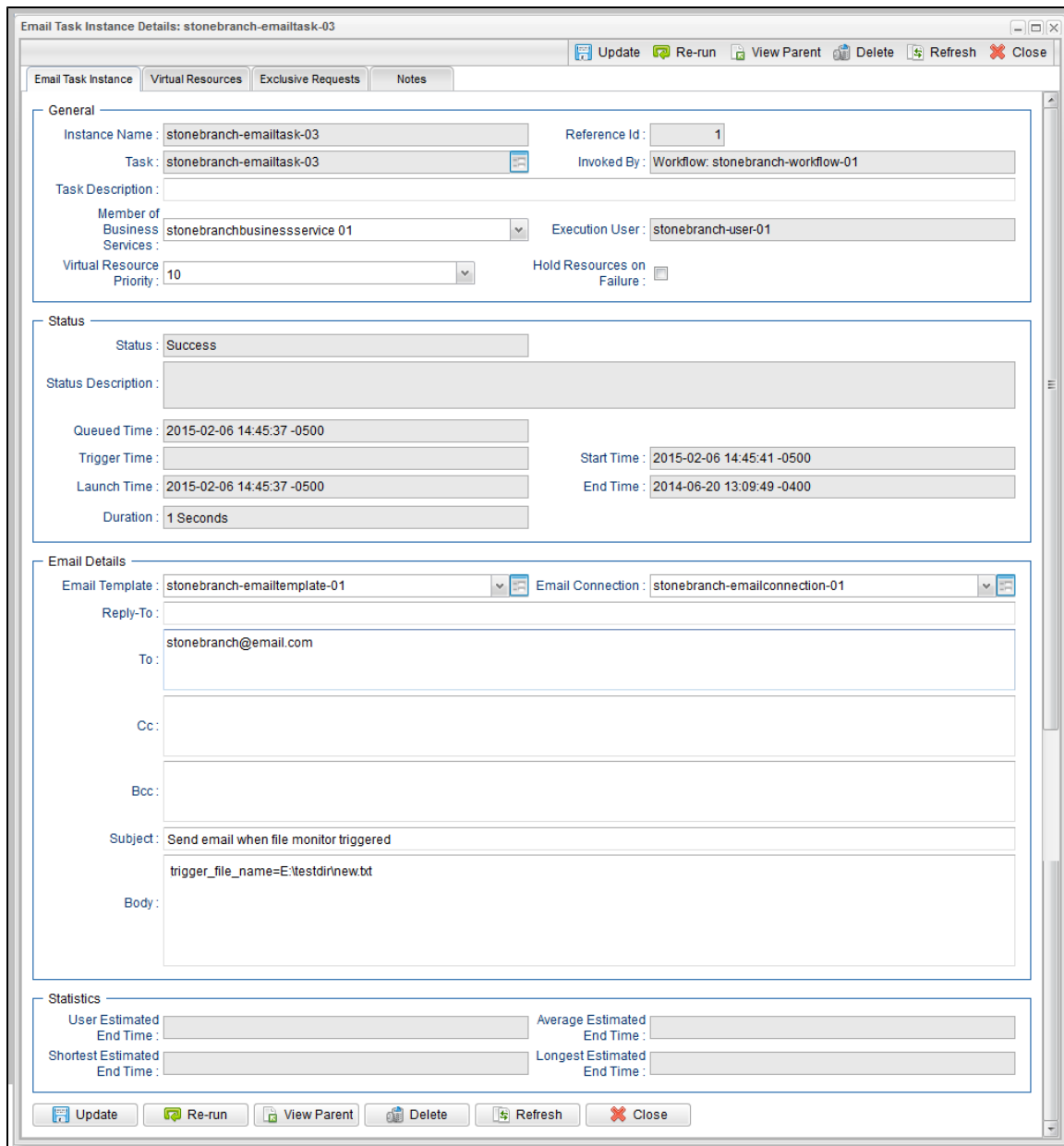
A task instance contains detailed information about a single execution of that task.

You can access a task instance from:

- **Instances tab** on the [Email Task Details](#) for that task
- [Activity Monitor](#)
- [Task Instances list](#)

Email Task Instance Details

The following Email Task Instance Details contains information on the execution of the task shown in the [Email Task Details](#).



Email Task Instance Details Field Descriptions

The following table describes the fields, buttons, and tabs that display in Email Task Instance Details.

Field Name	Description
General	This section contains general information about the task instance.
Instance Name	Name of this task instance.
Reference Id	System-supplied; the Controller increments this number each time the task is run.
Task	Name of the task that was run to create this task instance. Click the icon to display Task Details for the task.

Invoked by	<p>System-supplied; how the task instance was launched.</p> <p>Options:</p> <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 identify the user, check the Execution User column for that task instance on the Task Instances screen or, on most task instance screens, the Execution User field.
Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
Execution User	System-supplied; if the task was launched manually, the ID of the user who launched it.
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.
Status	This section contains information about the current status of the task instance.
Status	System-supplied; see Task Instance Statuses .
Status Description	System-supplied; additional information, if any, about the status of the task instance.
Queued Time	System-supplied; Date and time the task was queued for processing.
Trigger Time	System-supplied; Date and time the task instance was triggered.
Launch Time	System-supplied; Date and time the task instance was launched.
Start Time	System-supplied; Date and time the task instance started.
End Time	System-supplied; Date and time the task instance completed.
Duration	System-supplied; amount of time the task instance took to run.
Email Details	This section contains assorted detailed information about the task instance.

Email Template	<p>Name of an Email template defined in an Email Template Details. An Email template allows you to specify standard recipients and text for outgoing emails. Enter the name of an existing Email template, select an Email template from the drop-down list, or click the Details icon to create a new Email template.</p> <p>Every Email template specifies an Email connection. If you do not specify an Email template in this field, you must specify an Email connection in the Email Connection field.</p> <p>If you specify both an Email template (in this field) and an Email Connection, the Email server specified in the Email Connection field overrides the Email server specified in this field.</p> <p>(Any information specified in an Email task overrides what is specified in an Email template.)</p>
Email Connection	<p>Required if an Email template is not specified in the Email Template field; name of an Email connection defined via the Email Connection definition screen. An email connection specifies information about an email server. Enter the name of an existing Email connection, or click the magnifying glass to browse for an existing Email connection or create a new Email connection.</p> <p>If you specify both an Email Template and an Email connection (in this field), the Email connection specified in this field overrides the Email connection specified in the Email Template field.</p>
Reply-To	Email address of the sender. Use commas to separate multiple recipients. Variables and functions supported.
To	Email address of the recipient. Use commas to separate multiple recipients. Variables and functions supported.
CC	Email address of the party being sent a copy of the email, if any. Use commas to separate multiple recipients. Variables and functions supported.
BCC	Email address of the party being sent a blind (hidden) copy of the email, if any. Use commas to separate multiple recipients. Variables and functions supported.
Subject	Subject line of the email. Variables and functions supported.
Body	Text of the email message. Variables and functions supported. If both the email template and the email task contain text in the body, the text is appended.
Time Options	This section contains time-related specifications for the task instance.
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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.
Started Late	System-supplied; this field is flagged if the task started later than the time specified in the Late Start fields.
Late Start Type	<p>Required if Late Start is enabled.</p> <p>Options:</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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.
Late Start Duration	<p>If Late Start Type = 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.</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>
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.
Finished Late	System-supplied; this field is flagged if the task finished later than the time or duration specified in the Late Finish fields.
Late Finish Type	<p>Required if Late Finish is enabled.</p> <p>Options:</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 = Time; Time after which the task finish time is considered late. Use HH:MM, 24-hour time.
Late Finish Duration	If Late Finish Type = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
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.
Finished Early	System-supplied; this field is flagged if the task finished earlier than the time specified in the Early Finish fields.
Early Finish Type	<p>Required if Early Finish is enabled.</p> <p>Options:</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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.

Statistics	This section contains time-related statistics for the task instance.
User Estimated End Time	System-supplied; if the user entered information into the User Estimated Duration field in the task Details, the Controller uses this information to calculate an end time for the task instance, based on the date/time the task instance started.
Average Estimated End Time	System-supplied; average estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Shortest Estimated End Time	System-supplied; shortest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Longest Estimated End Time	System-supplied; longest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Buttons	This section identifies the buttons displayed above and below the Task Instance Details that let you perform various actions.
Update	Saves updates to the record.
Force Finish	See Force Finishing a Task .
Hold	Places the task instance on Hold (see Putting a Task on Hold).
Skip	For tasks loaded into the schedule that have not yet run; allows you to tell the Controller to skip this task. See Skipping a Task .
Re-run	See Re-running a Task .
View Parent	Displays the task instance Details for the parent Workflow of this task instance.
Retrieve Output	See Retrieving Output .
Delete	Deletes the current record.
Refresh	Refreshes any dynamic data displayed in the Details.
Close	For pop-up view only; closes the pop-up view of this task instance.
Tabs	This section identifies the tabs across the top of the Task Instance Details that provide access to additional information about the task instance.
Virtual Resources	Lists all Virtual Resources to which this task is assigned.
Exclusive Requests	Lists all records in the Exclusive Requests table (<code>ops_exclusive_order</code>) for this task instance.
Notes	Lists all notes associated with this record.

Running an Email Task

You can run an Email task:

- Manually, by clicking the [Launch Task](#) or [Launch Task with Variables](#) button in the Email Tasks list or Email Task Details [Action](#) menu.
- As part of a [workflow](#).
- [Specify triggers](#) that run the task automatically based on times or events.

Monitoring Task Execution

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

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 Task Monitor Task](#)
 - [Task Monitor Task Details](#)
 - [Task Monitor Task Details Field Descriptions](#)
- [Viewing a Task Monitor Task Instance](#)
 - [Task Monitor Task Instance Details](#)
 - [Task Monitor Task Instance Details 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
- Specific task
- Task type, such as a Windows task
- Group of tasks based on the name, such as all tasks whose name contains the string **DEV**

You also can 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
- 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 following [built-in variables](#) can be used in a Task Monitor task 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.



Note

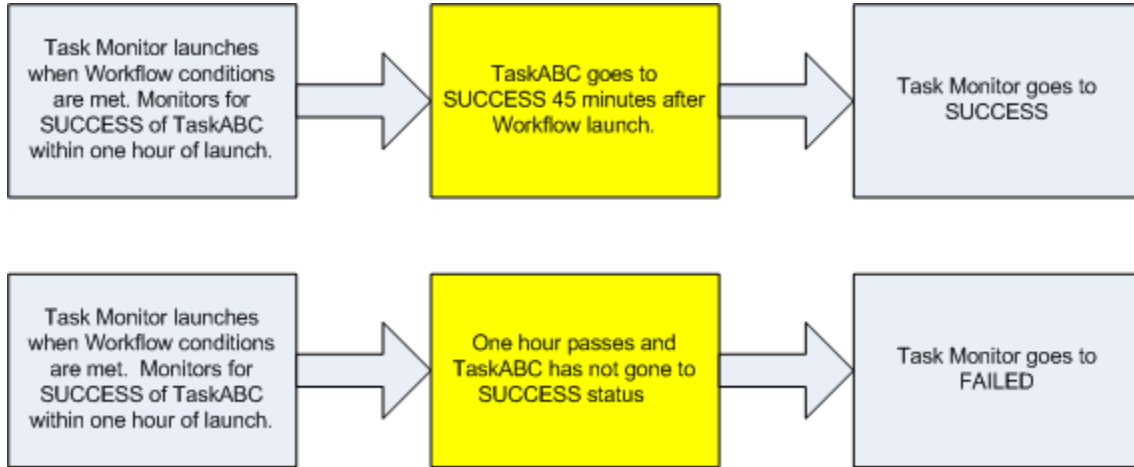
Any changes made to a Task Monitor task are not recognized by its respective Triggers until those Triggers are disabled and re-enabled.

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 Controller 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.

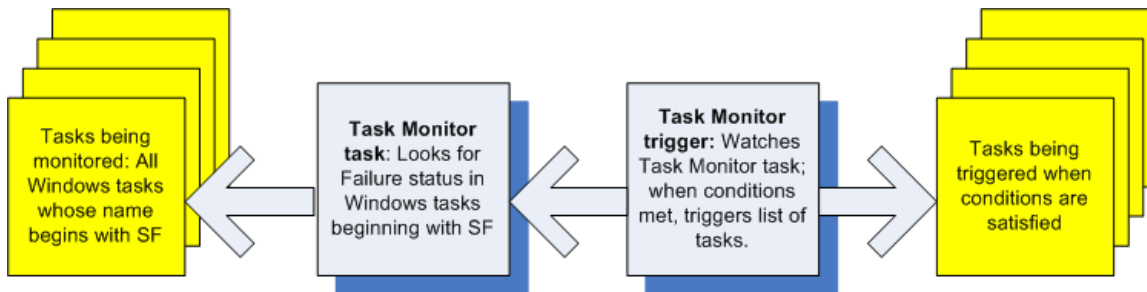
The following diagram illustrates how a Task Monitor might go to SUCCESS and FAILED status within a workflow.



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 the Controller 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 Task Monitor Task

Step 1 From the [Automation Center](#) navigation pane, select **Tasks > Task Monitor Tasks**. The Task Monitor Tasks list displays a list of all currently defined Task Monitor tasks.

Below the list, Task Monitor Task Details for a new Task Monitor task displays. (You also can click the **New** button to display Task Monitor Task Details for a new Task Monitor task.)

The screenshot shows the 'Task Monitors' interface. At the top, there is a 'Task Monitors' tab and a '5 Task Monitors' dropdown. Below this is a table with columns: Task Name, Task Description, Monitoring Type, Status To Monitor, Updated By, and Updated. The table contains five rows of tasks, all with 'Specific Task' monitoring type and 'Success' status. Below the table is the 'Task Monitor Details' form, which has tabs for 'Task Monitor', 'Variables', 'Actions', 'Virtual Resources', 'Mutually Exclusive', 'Instances', 'Task Monitor Triggers', 'Notes', and 'Versions'. The 'Task Monitor' tab is active, showing a 'General' section with fields for Task Name, Task Description, Member of Business Services, Hold on Start, Virtual Resource Priority, and Hold Resources on Failure. The 'Monitor Details' section has fields for Status To Monitor, Monitoring Type (set to 'Specific Task'), Workflow Name, Condition, Time Scope, and Task To Monitor.

Task Name	Task Description	Monitoring Type	Status To Monitor	Updated By	Updated
stonebranch-taskmonitor-01		Specific Task	Success	ops.admin	2014-06-13 14:18:45 -0400
stonebranch-taskmonitor-02		Specific Task	Success	ops.admin	2014-06-13 14:18:54 -0400
stonebranch-taskmonitor-03		Specific Task	Success	ops.admin	2014-06-13 14:18:59 -0400
stonebranch-taskmonitor-04		Specific Task	Success	ops.admin	2014-06-13 14:19:04 -0400
stonebranch-taskmonitor-05		Specific Task	Success	ops.admin	2014-06-13 14:19:08 -0400

Step 2 Enter/select Details for a new Task Monitor task, using the [field descriptions](#) below as a guide.

- Required fields display in **boldface**.
- Default values for fields, if available, display automatically.

To display more of the Details fields on the screen, you can temporarily [hide the list](#).



Note

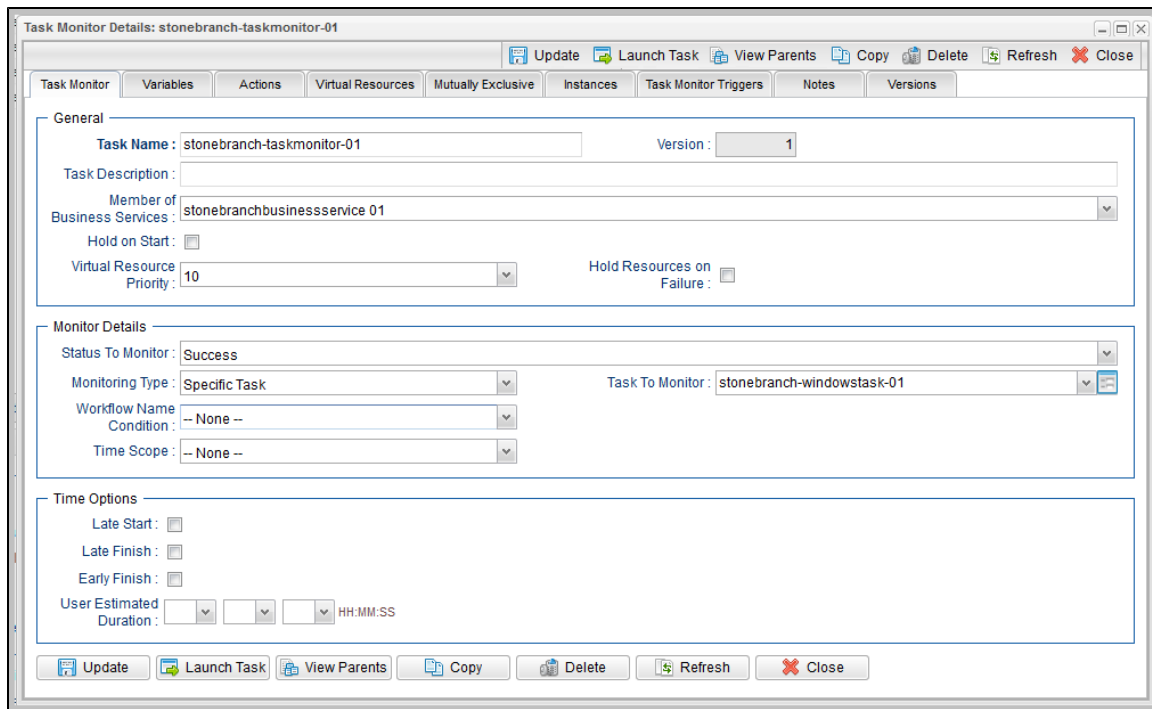
If you view [Task Monitor Task Details](#) for an existing Task Monitor task by clicking a task in the list, and then want to create a new Task Monitor task, you must click the **New** button that displays above and below the Details.

Step 3 Click the **Save** button. The task is added to the database, and all buttons and tabs in the Task Details are enabled.

Task Monitor Task Details

The following Task Monitor Task Details is for an existing Task Monitor task.

Depending on the values that you enter / select for these fields, and whether or not the Task Monitor task has ever been launched, more (or less) fields may display. See the [field descriptions](#), below, for a description of all fields that may display in the Task Monitor Task Details.




Task Monitor Task Details Field Descriptions


The following table describes the fields, buttons, and tabs that display in the Task Monitor Task Details.

Field Name	Description
General	This section contains general information about the task.
Task Name	User-defined name of this task, containing a maximum 255 alphanumeric characters; variables supported. It is the responsibility of the user to develop a workable naming scheme for tasks.
Version	System-supplied; version number of the current record, which is incremented by the Controller every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
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.
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.
Task Monitor Details	This section contains assorted detailed information about the task.
Status To Monitor	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	Specifies which task or tasks are being monitored. Options: <ul style="list-style-type: none"> • Specific Task - One task is being monitored. Use the Task to Monitor field to specify the task name. • General Tasks - Allows you to specify selection parameters that determine which task or tasks to be monitored. Use the Task Name Condition and Task Type to Monitor fields to create your selection parameters.
Task to Monitor	If Monitoring Type = Specific Task; specifies the task to monitor. Enter a task name or select a task from the drop-down list. To display details about a task on the list, select it and then click the Task To Monitor icon.
Task Name Condition	If Monitoring Type = General Task(s); specifies a type of condition for the name of tasks being monitored for. If you select a condition type, a corresponding field displays that allows you to enter a value for that condition. Only tasks meeting the specified condition value will be monitored for. Options: <ul style="list-style-type: none"> • None • Starts With • Contains • Ends With
Task Type to Monitor	If Monitoring Type = General Tasks; 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.
Task Name Starts With	If Task Name Condition = Starts With; character string at the start of the name of a task or tasks being monitored for.
Task Name Contains	If Task Name Condition = Contains; character string in the name of a task or tasks being monitored for.
Task Name Ends With	If Task Name Condition = Ends With; character string at the end of the name of a task or tasks being monitored for.
Workflow Name Condition	Specifies a type of condition for the name of a workflow or workflows containing the task being monitored for. If you select a condition type, a corresponding field displays that allows you to enter a value for that condition. Only tasks in workflows meeting the specified condition value will be monitored for. Options: <ul style="list-style-type: none"> • None • Equals • Starts With • Contains • Ends With
Workflow Name Equals	If Workflow Name Condition = Equals; exact name of a workflow or workflows containing the task being monitored for.

Workflow Name Starts With	If Workflow Name Condition = Starts With; character string at the start of the name of a workflow or workflows containing the task being monitored for.
Workflow Name Contains	If Workflow Name Condition = Contains; character string in the name of a workflow or workflows containing the task being monitored for.
Workflow Name Ends With	If Workflow Name Condition = Ends With; character string at the end of the name of a workflow or workflows containing the task being monitored for.
Time Scope	<p>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.</p> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note The task being monitored still must be in the Activity Monitor 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 All Task Instances table (ops_exec) for the specified event. If it locates the event, the Task Monitor is satisfied immediately.</p> </div>
From [+/-]hh:mm:	<p>If Time Scope = Relative; used for Task Monitor tasks not associated with a trigger. Together with the To 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.</p> <p>If you specify a past time in the this field, as soon as the Task Monitor task launches, the Controller 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, the Controller continues monitoring until the conditions are met or until the To time has passed.</p>
To [+/-]hh:mm:	<p>If Time Scope = Relative; 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.</p> <p>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.</p>
Time Options	This section contains time-related specifications for task instances of the task.
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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.
Late Start Type	<p>Required if Late Start is enabled.</p> <p>Options:</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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.

Late Start Duration	<p>If Late Start Type = 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.</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>
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.</p> <p>Options:</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 = Time; 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 = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.</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.</p> <p>Options:</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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.</p>
User Estimated Duration	<p>Estimated amount of time it should normally take to run this task. The Controller uses this information to calculate the User Estimated End Time on a task instance record.</p>
Statistics	<p>This section contains time-related statistics for task instances of the task.</p>
First Time Ran	<p>System-supplied; date and time this task first ran.</p>

Last Time Ran	System-supplied; date and time the task last ran.
Last Instance Duration	System-supplied; amount of time the task took to run the last time it ran.
Lowest Instance Time	System-supplied; shortest amount of time this task has taken to run.
Average Instance Time	System-supplied; average amount of time this task takes to run.
Highest Instance Time	System-supplied; longest amount of time this task has taken to run.
Number of Instances	System-supplied; number of instances in the database for this task.
Buttons	This section identifies the buttons displayed above and below the Task Details that let you perform various actions.
Save	Saves a new task record in the Controller database.
Update	Saves updates to the record.
New	Displays empty (except for default values) Details for creating a new task.
Launch Task	Manually launches the task.
View Parents	Displays a list of any parent Workflow tasks for this task.
Copy	Creates a copy of this task, which you are prompted to rename.
Delete	<p>Deletes the current record.</p> <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note You cannot delete a task if it is either:</p> <ul style="list-style-type: none"> Specified in an enabled Trigger. The only task specified in a disabled Trigger. </div>
Refresh	Refreshes any dynamic data displayed in the Details.
Close	For pop-up view only; closes the pop-up view of this task.
Tabs	This section identifies the tabs across the top of the Task Details that provide access to additional information about the task.
Variables	Lists all variables associated with this record.

Actions	<p>Allows you to specify actions that the Controller will take automatically based on events that occur during the execution of this task.</p> <p>Events are:</p> <ul style="list-style-type: none"> • Task instance status • Exit codes • Late start • Late finish • Early finish <p>Actions are:</p> <table border="1" data-bbox="266 478 1484 840"> <tr> <td data-bbox="266 478 410 548">Abort Action</td> <td data-bbox="410 478 1484 548">Abort the task if certain events occur. For details, see Abort Actions.</td> </tr> <tr> <td data-bbox="266 548 410 617">Email Notification</td> <td data-bbox="410 548 1484 617">Send an email if certain events occur. For details, see Email Notification Actions.</td> </tr> <tr> <td data-bbox="266 617 410 686">Set Variable</td> <td data-bbox="410 617 1484 686">Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow.</td> </tr> <tr> <td data-bbox="266 686 410 756">SNMP Notification</td> <td data-bbox="410 686 1484 756">Send an email if certain events occur. For details, see SNMP Notification Actions.</td> </tr> <tr> <td data-bbox="266 756 410 825">System Operation</td> <td data-bbox="410 756 1484 825">Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions.</td> </tr> </table>	Abort Action	Abort the task if certain events occur. For details, see Abort Actions .	Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .	Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .	SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .	System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .
Abort Action	Abort the task if certain events occur. For details, see Abort Actions .										
Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .										
Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .										
SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .										
System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .										
Virtual Resources	Lists all Virtual Resources to which this task is assigned.										
Mutually Exclusive	Lists all tasks that have been set to be mutually exclusive of this task.										
Instances	Lists all instances of this Workflow.										
Task Monitor Triggers	Lists 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, the Controller 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 Triggers Overview .										
Notes	Lists all notes associated with this record.										
Versions	Stores copies of all previous versions of the current record. See Record Versioning .										

Viewing a Task Monitor Task Instance

When a Task Monitor task is launched, the Controller creates a task instance record of that task.

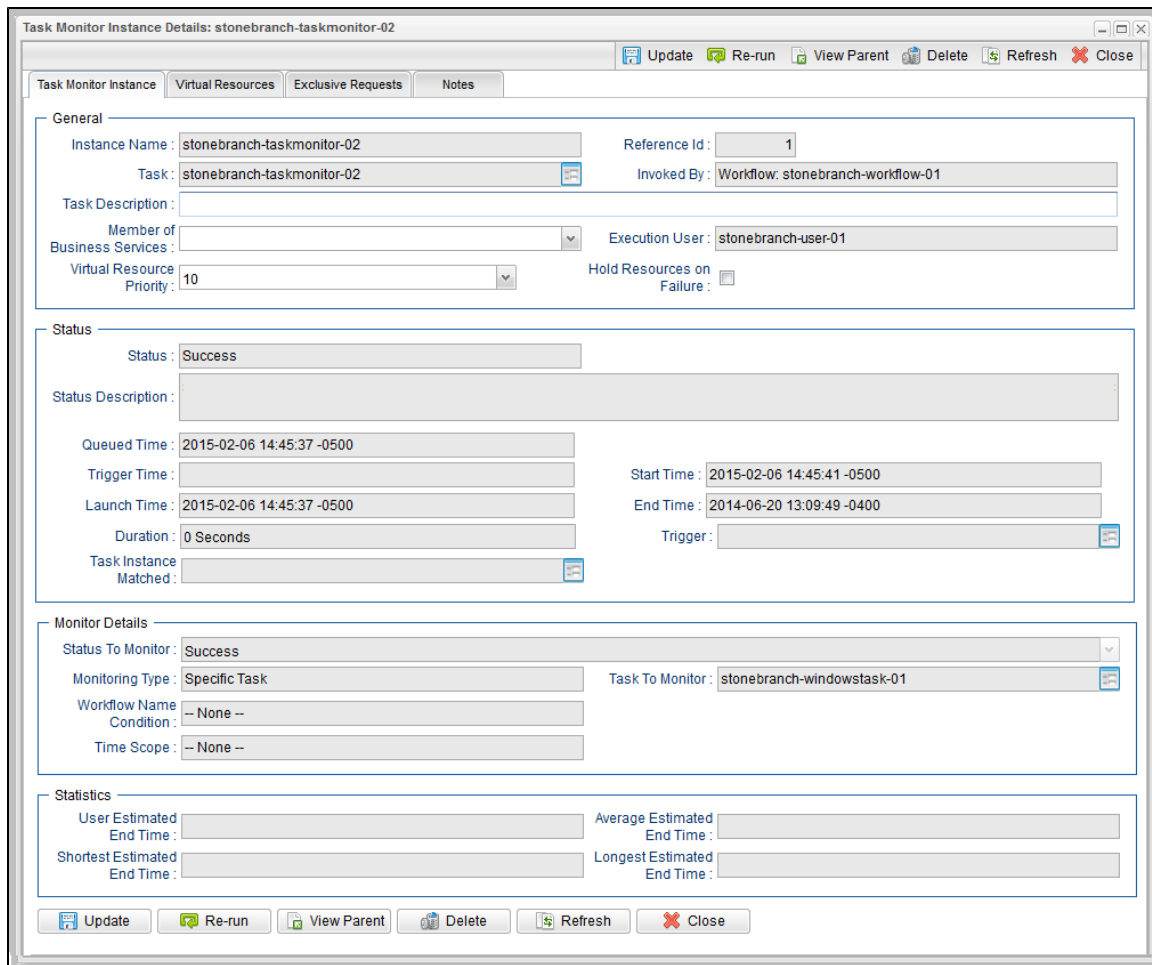
A task instance contains detailed information about a single execution of that task.

You can access a task instance from:

- **Instances tab** on the [Task Monitor Task Details](#) for that task
- [Activity Monitor](#)
- [Task Instances list](#)

Task Monitor Task Instance Details

The following Task Monitor Task Instance Details contains information on the execution of the task shown in the [Task Monitor Task Details](#).




Task Monitor Task Instance Details Field Descriptions

The following table describes the fields, buttons, and tabs that display in Task Monitor Task Instance Details.

Field Name	Description
General	This section contains general information about the task instance.
Instance Name	Name of this task instance.
Reference Id	System-supplied; the Controller increments this number each time the task is run.
Task	Name of the task that was run to create this task instance. Click the icon to display Task Details for the task.
Invoked by	System-supplied; how the task instance was launched. Options: <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 identify the user, check the Execution User column for that task instance on the Task Instances screen or, on most task instance screens, the Execution User field.
Task Description	User-defined; description of this record. (Maximum = 200 characters.)

Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
Execution User	System-supplied; if the task was launched manually, the ID of the user who launched it.
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.
Status	This section contains information about the current status of the task instance.
Status	System-supplied; see Task Instance Statuses .
Status Description	System-supplied; additional information, if any, about the status of the task instance.
Queued Time	System-supplied; Date and time the task was queued for processing.
Trigger Time	System-supplied; Date and time the task instance was triggered.
Launch Time	System-supplied; Date and time the task instance was launched.
Start Time	System-supplied; Date and time the task instance started.
End Time	System-supplied; Date and time the task instance completed.
Duration	System-supplied; amount of time the task instance took to run.
Trigger	Trigger, if any, on whose behalf the Task Monitor task is monitoring other tasks.
Task Instance Matched	Last task that matched the specifications of the task(s) being monitored.
Task Monitor Details	This section contains assorted detailed information about the task instance.
Status To Monitor	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	<p>Specifies which task or tasks are being monitored.</p> <p>Options:</p> <ul style="list-style-type: none"> • Specific Task - One task is being monitored. Use the Task to Monitor field to specify the task name. • General Tasks - Allows you to specify selection parameters that determine which task or tasks to be monitored. Use the Task Name Condition and Task Type to Monitor fields to create your selection parameters.
Task to Monitor	<p>If Monitoring Type = Specific Task; specifies the task to monitor. Enter a task name or select a task from the drop-down list. To display details about a task on the list, select it and then click the Task To Monitor icon.</p>
Task Name Condition	<p>If Monitoring Type = General Task(s); specifies a type of condition for the name of tasks being monitored for. If you select a condition type, a corresponding field displays that allows you to enter a value for that condition. Only tasks meeting the specified condition value will be monitored for.</p> <p>Options:</p> <ul style="list-style-type: none"> • None • Starts With • Contains • Ends With
Task Name Starts With	<p>If Task Name Condition = Starts With; character string at the start of the name of a task or tasks being monitored for.</p>
Task Name Contains	<p>If Task Name Condition = Contains; character string in the name of a task or tasks being monitored for.</p>
Task Name Ends With	<p>If Task Name Condition = Ends With; character string at the end of the name of a task or tasks being monitored for.</p>
Workflow Name Condition	<p>Specifies a type of condition for the name of a workflow or workflows containing the task being monitored for. If you select a condition type, a corresponding field displays that allows you to enter a value for that condition.</p> <p>Only tasks in workflows meeting the specified condition value will be monitored for.</p> <p>Options:</p> <ul style="list-style-type: none"> • None • Equals • Starts With • Contains • Ends With
Workflow Name Equals	<p>If Workflow Name Condition = Equals; exact name of a workflow or workflows containing the task being monitored for.</p>
Workflow Name Starts With	<p>If Workflow Name Condition = Starts With; character string at the start of the name of a workflow or workflows containing the task being monitored for.</p>
Workflow Name Contains	<p>If Workflow Name Condition = Contains; character string in the name of a workflow or workflows containing the task being monitored for.</p>
Workflow Name Ends With	<p>If Workflow Name Condition = Ends With; character string at the end of the name of a workflow or workflows containing the task being monitored for.</p>

Time Scope	<p>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.</p> <div data-bbox="305 310 1446 489" style="background-color: #ffffcc; padding: 10px;"> <p> Note The task being monitored still must be in the Activity Monitor 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 All Task Instances table (ops_exec) for the specified event. If it locates the event, the Task Monitor is satisfied immediately.</p> </div>
From [+/-]hh:mm:	<p>If Time Scope = Relative; used for Task Monitor tasks not associated with a trigger. Together with the To 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.</p> <p>If you specify a past time in the this field, as soon as the Task Monitor task launches, the Controller 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, the Controller continues monitoring until the conditions are met or until the To time has passed.</p>
To [+/-]hh:mm:	<p>If Time Scope = Relative; 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.</p> <p>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.</p>
Time Options	This section contains time-related specifications for the task instance.
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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.</p>
Started Late	System-supplied; this field is flagged if the task started later than the time specified in the Late Start fields.
Late Start Type	<p>Required if Late Start is enabled.</p> <p>Options:</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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.
Late Start Duration	<p>If Late Start Type = 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.</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>

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.
Finished Late	System-supplied; this field is flagged if the task finished later than the time or duration specified in the Late Finish fields.
Late Finish Type	Required if Late Finish is enabled. Options: <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 = Time; Time after which the task finish time is considered late. Use HH:MM, 24-hour time.
Late Finish Duration	If Late Finish Type = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
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.
Finished Early	System-supplied; this field is flagged if the task finished earlier than the time specified in the Early Finish fields.
Early Finish Type	Required if Early Finish is enabled. Options: <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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Statistics	This section contains time-related statistics for the task instance.
User Estimated End Time	System-supplied; if the user entered information into the User Estimated Duration field in the task Details, the Controller uses this information to calculate an end time for the task instance, based on the date/time the task instance started.
Average Estimated End Time	System-supplied; average estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.

Shortest Estimated End Time	System-supplied; shortest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Longest Estimated End Time	System-supplied; longest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Buttons	This section identifies the buttons displayed above and below the Task Instance Details that let you perform various actions.
Update	Saves updates to the record.
Force Finish	See Force Finishing a Task .
Hold	Places the task instance on Hold (see Putting a Task on Hold).
Skip	For tasks loaded into the schedule that have not yet run; allows you to tell the Controller to skip this task. See Skipping a Task .
Re-run	See Re-running a Task .
View Parent	Displays the task instance Details for the parent Workflow of this task instance.
Retrieve Output	See Retrieving Output .
Delete	Deletes the current record.
Refresh	Refreshes any dynamic data displayed in the Details.
Close	For pop-up view only; closes the pop-up view of this task instance.
Tabs	This section identifies the tabs across the top of the Task Instance Details that provide access to additional information about the task instance.
Virtual Resources	Lists all Virtual Resources to which this task is assigned.
Exclusive Requests	Lists all records in the Exclusive Requests table (<code>ops_exclusive_order</code>) for this task instance.
Notes	Lists all notes associated with this record.

Monitoring Task Execution

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

File Monitor Task

- Overview
- Processing Flow for File Monitors
 - 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 File Monitor Task
 - File Monitor Task Details
 - File Monitor Task Details Field Descriptions
- Viewing a File Monitor Task Instance
 - File Monitor Task Instance Details
 - File Monitor Task Instance Details 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 Opwise Universal Agent for Windows, Linux/Unix, or z/OS running on the machine where you are monitoring for the file.

Processing Flow for File Monitors

File Monitor tasks are meant to be launched either by using a [File Monitor trigger](#) or by being included 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.



Note

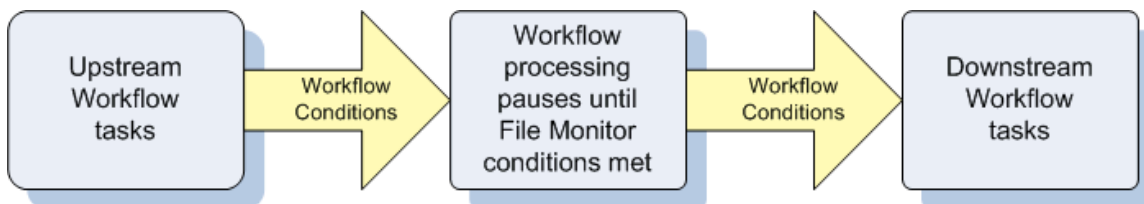
Any changes made to a File Monitor task are not recognized by its respective Triggers until those Triggers are disabled and re-enabled.

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 a

condition(s) is satisfied.

In this scenario, the File Monitor task launches when its associated File Monitor trigger is enabled.

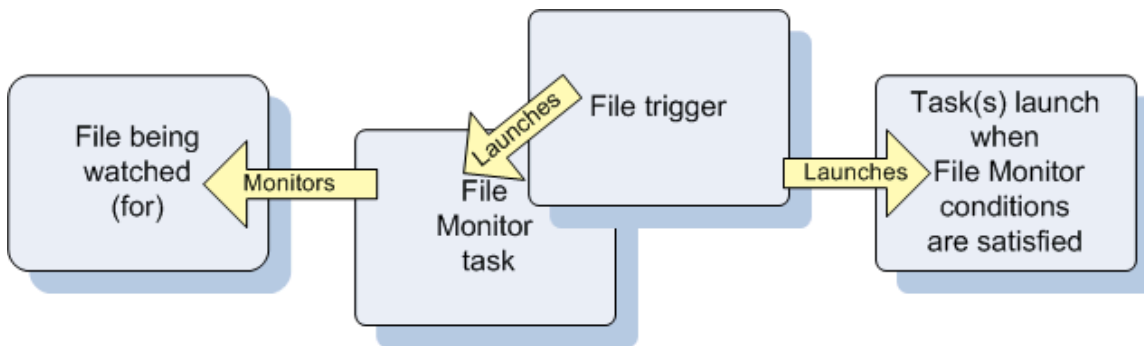
This method is best geared toward watching for the creation, deletion, or change in files. When the conditions in the File Monitor task are satisfied, the File Monitor task goes to SUCCESS and the tasks listed in the associated trigger are launched. The File Monitor task continues running until its conditions 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 Monitor 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 following [built-in variables](#) can be used in a File Monitor task to pass data where appropriate:

- Task and Task Instance variables
- File Monitor variables

Creating a File Monitor Task

Step 1 From the [Automation Center](#) navigation pane, select **Tasks > File Monitor Tasks**. The File Monitor Tasks list displays a list of all currently defined File Monitor tasks.

Below the list, File Monitor Task Details for a new File Monitor task displays. (You also can click the **New** button to display File Monitor Task Details for a new File Monitor task.)

Task Name	Task Description	Monitor Type	Updated By	Updated
stonebranch-filemonitor-01		Create	ops.admin	2014-06-13 14:19:55 -0400
stonebranch-filemonitor-02		Create	ops.admin	2014-06-13 14:20:04 -0400
stonebranch-filemonitor-03		Create	ops.admin	2014-06-13 14:20:09 -0400
stonebranch-filemonitor-04		Create	ops.admin	2014-06-13 14:20:13 -0400
stonebranch-filemonitor-05		Create	ops.admin	2014-06-13 14:20:18 -0400

File Monitor Details

File Monitor | Variables | Actions | Virtual Resources | Mutually Exclusive | Instances | File Monitor Triggers | Notes | Versions

General

Task Name: Version:

Task Description:

Member of Business Services:

Hold on Start:

Virtual Resource Priority: Hold Resources on Failure:

File Monitor Details

Agent: Agent Cluster:

Agent Variable: Agent Cluster Variable:

Credentials: Cluster Broadcast:

Credentials Variable:

Step 2 Enter/select Details for a new File Monitor task, using the [field descriptions](#) below as a guide.

- Required fields display in **boldface**.
- Default values for fields, if available, display automatically.

To display more of the Details fields on the screen, you can temporarily [hide the list](#).



Note

If you view [File Monitor Task Details](#) for an existing File Monitor task by clicking a task in the list, and then want to create a new File Monitor task, you must click the **New** button that displays above and below the Details.

Step 3 Click the **Save** button. The task is added to the database, and all buttons and tabs in the Task Details are enabled.

File Monitor Task Details



The following File Monitor Task Details is for an existing File Monitor task.


Depending on the values that you enter / select for these fields, and whether or not the File Monitor task has ever been launched, more (or less) fields may display. See the [field descriptions](#), below, for a description of all fields that may display in the File Monitor Task Details.

File Monitor Task Details Field Descriptions

The following table describes the fields, buttons, and tabs that display in the File Monitor Task Details.


Field Name	Description
General	This section contains general information about the task.
Task Name	User-defined name of this task, containing a maximum 255 alphanumeric characters; variables supported. It is the responsibility of the user to develop a workable naming scheme for tasks.
Version	System-supplied; version number of the current record, which is incremented by the Controller every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
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.

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	<p>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.</p>
File Monitor Details	<p>This section contains assorted detailed information about the task.</p>
Agent	<p>Name of the Agent resource that identifies the machine where the operation will run. If you do not specify an Agent, you must specify an Agent Cluster.</p>
Agent Variable	<p>If enabled, 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. Use the format: \${variable name}. The variable must be a supported type as described in Variables and Functions.</p> <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note If you select Agent Variable when updating multiple tasks, the Agent field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Agent Unresolved - into which you enter a variable, variable and text, or Agent name. The Agent field is not used.</p> </div>
Agent Cluster	<p>Group of Agents, one of which the Controller will choose to run this task. You can specify an agent cluster in addition to or in place of a specific Agent. If you specify an Agent and an agent cluster, the Controller first tries to run the task on the specific agent. If the Agent is not available, the Controller reverts to the agent cluster. See Agent Clusters for more information.</p>
Agent Cluster Variable	<p>If enabled, 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. Use the format: \${variable name}. The variable must be a supported type as described in Variables and Functions.</p> <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note If you select Agent Cluster Variable when updating multiple tasks, the Agent Cluster field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Agent Cluster Unresolved - into which you enter a variable, variable and text, or Agent Cluster name. The Agent Cluster field is not used.</p> </div>
Cluster Broadcast	<p>Group of Agents, all of which will run this task. You can specify a Cluster Broadcast in place of a specific Agent and/or agent cluster. Each instance of the task running on its own Agent becomes a separate task instance record in the database and displays separately on the Activity screen. See Agent Clusters for more information about defining agent clusters.</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>

Credentials Variable	<p>If enabled, the Credentials field 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> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Credentials Variable when updating multiple tasks, the Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Credentials Unresolved - into which you enter a variable, variable and text, or Credentials name. The Credentials field is not used.</p> </div>
Monitor Type	<p>Type of file event being monitored for.</p> <p>Options:</p> <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.
Trigger on Existence	If Monitor Type = Create; task is triggered if the file being monitored for creation already exists.
Monitor File(s)	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.
Maximum Files	If Monitor Type = Create, Delete, or Change; For searches that use wildcards, limits the number of files to be searched.
File Owner	If Monitor Type = Create, Delete, Change, or Exists; User name / group name of the owner of the file on the operating system; that is, the user name / group name returned by the operating system in the file ownership information. LDAP groups are supported. Specifying a file owner limits the search to files with that owner.
Stable (seconds)	If Monitor Type = Change or Create: Period of time, in seconds, during which the file has not changed.
By Percentage (+/-)	If Monitor 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 Monitor 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 Monitor 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 10MB, enter 10 in the By Size field and select MB in this field.
To Size	If Monitor 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 5KB, enter 5 in this field and select KB in the To scale field.

To Scale	If Monitor 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 5KB, enter 5 in the To Size field and select KB in this field.
Scan Text	If Monitor Type = Create, Change, or Exists; 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. The Controller will process this field as a regular expression.
Scan Forward	If Monitor 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.
Time Options	This section contains time-related statistics for task instances of the task.
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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.
Late Start Type	Required if Late Start is enabled. Options: <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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.
Late Start Duration	If Late Start Type = 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.
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: <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 = Time; Time after which the task finish time is considered late. Use HH:MM, 24-hour time.

Late Finish Duration	If Late Finish Type = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
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: <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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
User Estimated Duration	Estimated amount of time it should normally take to run this task. The Controller uses this information to calculate the User Estimated End Time on a task instance record.
Statistics	This section contains time-related statistics for task instances of the task.
First Time Ran	System-supplied; date and time this task first ran.
Last Time Ran	System-supplied; date and time the task last ran.
Last Instance Duration	System-supplied; amount of time the task took to run the last time it ran.
Lowest Instance Time	System-supplied; shortest amount of time this task has taken to run.
Average Instance Time	System-supplied; average amount of time this task takes to run.
Highest Instance Time	System-supplied; longest amount of time this task has taken to run.
Number of Instances	System-supplied; number of instances in the database for this task.
Buttons	This section identifies the buttons displayed above and below the Task Details that let you perform various actions.
Save	Saves a new task record in the Controller database.
Update	Saves updates to the record.

New	Displays empty (except for default values) Details for creating a new task.										
Launch Task	Manually launches the task.										
View Parents	Displays a list of any parent Workflow tasks for this task.										
Copy	Creates a copy of this task, which you are prompted to rename.										
Delete	<p>Deletes the current record.</p> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note You cannot delete a task if it is either:</p> <ul style="list-style-type: none"> Specified in an enabled Trigger. The only task specified in a disabled Trigger. </div>										
Refresh	Refreshes any dynamic data displayed in the Details.										
Close	For pop-up view only; closes the pop-up view of this task.										
Tabs	This section identifies the tabs across the top of the Task Details that provide access to additional information about the task.										
Variables	Lists all variables associated with this record.										
Actions	<p>Allows you to specify actions that the Controller will take automatically based on events that occur during the execution of this task.</p> <p>Events are:</p> <ul style="list-style-type: none"> Task instance status Exit codes Late start Late finish Early finish <p>Actions are:</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="background-color: #e6e6fa;">Abort Action</td> <td>Abort the task if certain events occur. For details, see Abort Actions.</td> </tr> <tr> <td style="background-color: #e6e6fa;">Email Notification</td> <td>Send an email if certain events occur. For details, see Email Notification Actions.</td> </tr> <tr> <td style="background-color: #e6e6fa;">Set Variable</td> <td>Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow.</td> </tr> <tr> <td style="background-color: #e6e6fa;">SNMP Notification</td> <td>Send an email if certain events occur. For details, see SNMP Notification Actions.</td> </tr> <tr> <td style="background-color: #e6e6fa;">System Operation</td> <td>Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions.</td> </tr> </table>	Abort Action	Abort the task if certain events occur. For details, see Abort Actions .	Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .	Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .	SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .	System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .
Abort Action	Abort the task if certain events occur. For details, see Abort Actions .										
Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .										
Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .										
SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .										
System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .										
Virtual Resources	Lists all Virtual Resources to which this task is assigned.										
Mutually Exclusive	Lists all tasks that have been set to be mutually exclusive of this task.										
Instances	Lists all instances of this Workflow.										

File Monitor Triggers	Lists 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, the Controller 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 Triggers Overview .
Notes	Lists all notes associated with this record.
Versions	Stores copies of all previous versions of the current record. See Record Versioning .

Viewing a File Monitor Task Instance

When a File Monitor task is launched, the Controller creates a task instance record of that task.

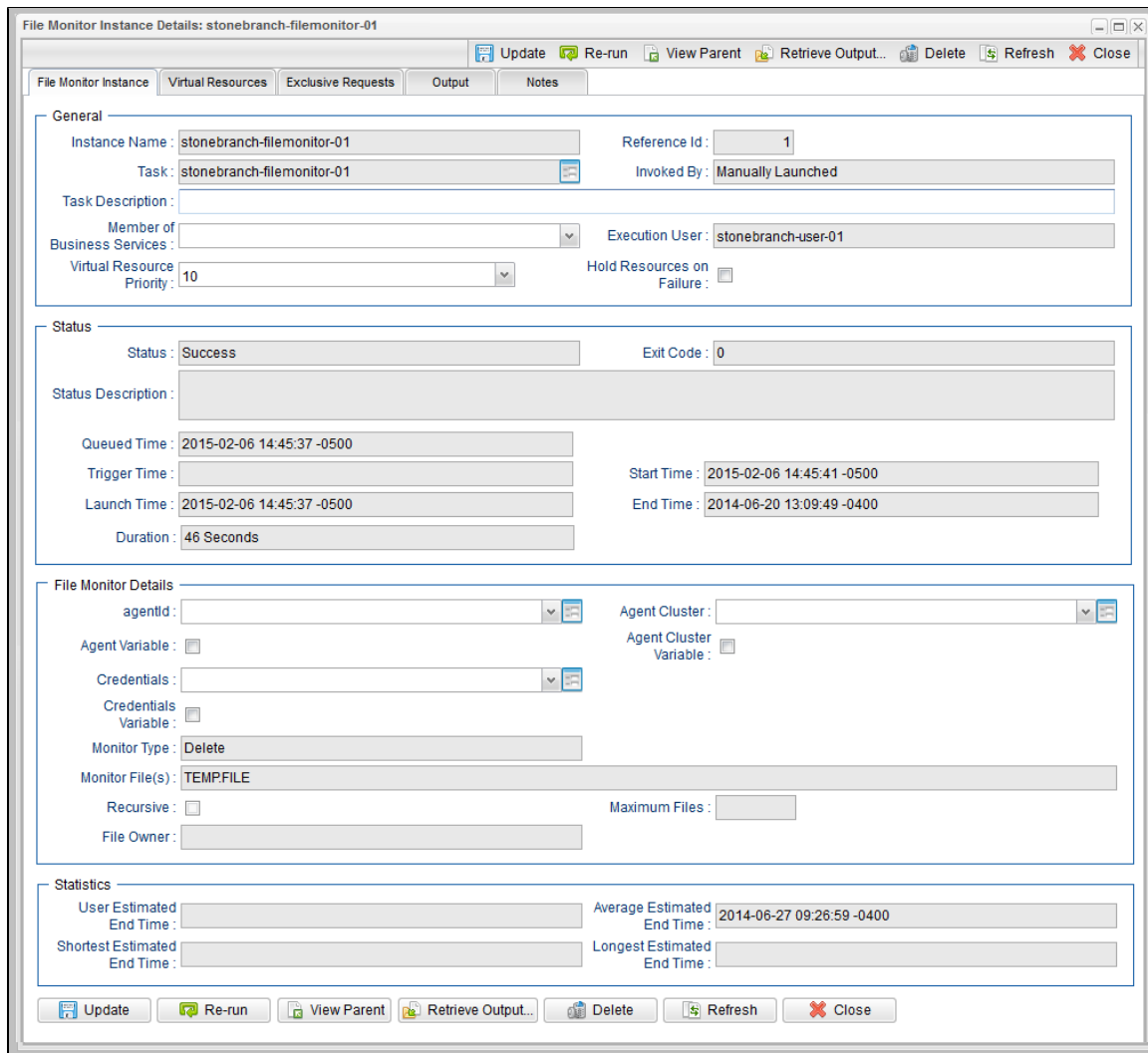
A task instance contains detailed information about a single execution of that task.

You can access a task instance from:

- **Instances tab** on the [File Monitor Task Details](#) for that task
- [Activity Monitor](#)
- [Task Instances list](#)

File Monitor Task Instance Details

The following File Monitor Task Instance Details contains information on the execution of the task shown in the [File Monitor Task Details](#).






File Monitor Task Instance Details Field Descriptions

The following table describes the fields, buttons, and tabs that display in File Monitor Task Instance Details.

Field Name	Description
General	This section contains general information about the task instance.
Instance Name	Name of this task instance.
Reference Id	System-supplied; the Controller increments this number each time the task is run.
Task	Name of the task that was run to create this task instance. Click the icon to display Task Details for the task.
Invoked by	System-supplied; how the task instance was launched. Options: <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 identify the user, check the Execution User column for that task instance on the Task Instances screen or, on most task instance screens, the Execution User field.

Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
Execution User	System-supplied; if the task was launched manually, the ID of the user who launched it.
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.
Status	This section contains information about the current status of the task instance.
Status	System-supplied; see Task Instance Statuses .
Exit Code	System-supplied; the exit code captured by the Agent when executing the task (for example, a command or script).
Status Description	System-supplied; additional information, if any, about the status of the task instance.
Queued Time	System-supplied; Date and time the task was queued for processing.
Trigger Time	System-supplied; Date and time the task instance was triggered.
Launch Time	System-supplied; Date and time the task instance was launched.
Start Time	System-supplied; Date and time the task instance started.
End Time	System-supplied; Date and time the task instance completed.
Duration	System-supplied; amount of time the task instance took to run.
File Monitor Details	This section contains assorted detailed information about the task instance.
Agent	Name of the Agent resource that identifies the machine where the operation will run. If you do not specify an Agent, you must specify an Agent Cluster .

Agent Variable	<p>If enabled, 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. Use the format: <code>\${variable name}</code>. The variable must be a supported type as described in Variables and Functions.</p> <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note If you select Agent Variable when updating multiple tasks, the Agent field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Agent Unresolved - into which you enter a variable, variable and text, or Agent name. The Agent field is not used.</p> </div>
Agent Cluster	<p>Group of Agents, one of which the Controller will choose to run this task. You can specify an agent cluster in addition to or in place of a specific Agent. If you specify an Agent and an agent cluster, the Controller first tries to run the task on the specific agent. If the Agent is not available, the Controller reverts to the agent cluster. See Agent Clusters for more information.</p>
Agent Cluster Variable	<p>If enabled, 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. Use the format: <code>\${variable name}</code>. The variable must be a supported type as described in Variables and Functions.</p> <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note If you select Agent Cluster Variable when updating multiple tasks, the Agent Cluster field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Agent Cluster Unresolved - into which you enter a variable, variable and text, or Agent Cluster name. The Agent Cluster field is not used.</p> </div>
Cluster Broadcast	<p>Group of Agents, all of which will run this task. You can specify a Cluster Broadcast in place of a specific Agent and/or agent cluster. Each instance of the task running on its own Agent becomes a separate task instance record in the database and displays separately on the Activity screen. See Agent Clusters for more information about defining agent clusters.</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>
Credentials Variable	<p>If enabled, the Credentials field 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> <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note If you select Credentials Variable when updating multiple tasks, the Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Credentials Unresolved - into which you enter a variable, variable and text, or Credentials name. The Credentials field is not used.</p> </div>
Monitor Type	<p>Type of file event being monitored for.</p> <p>Options:</p> <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.
Trigger on Existence	<p>If Monitor Type = Create; task is triggered if the file being monitored for creation already exists.</p>

Monitor File(s)	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.
Maximum Files	If Monitor Type = Create, Delete, or Change; For searches that use wildcards, limits the number of files to be searched.
File Owner	If Monitor Type = Create, Delete, Change, or Exists; User name / group name of the owner of the file on the operating system; that is, the user name / group name returned by the operating system in the file ownership information. LDAP groups are supported. Specifying a file owner limits the search to files with that owner.
Stable (seconds)	If Monitor Type = Change or Create: Period of time, in seconds, during which the file has not changed.
By Percentage (+/-)	If Monitor 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 Monitor 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 Monitor 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 10MB, enter 10 in the By Size field and select MB in this field.
To Size	If Monitor 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 5KB, enter 5 in this field and select KB in the To scale field.
To scale	If Monitor 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 5KB, enter 5 in the To Size field and select KB in this field.
Scan Text	If Monitor Type = Create, Change, or Exists; 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. The Controller will process this field as a regular expression.
Scan Forward	If Monitor 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.
Time Options	This section contains time-related specifications for the task instance.
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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.
Started Late	System-supplied; this field is flagged if the task started later than the time specified in the Late Start fields.

Late Start Type	<p>Required if Late Start is enabled.</p> <p>Options:</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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.
Late Start Duration	<p>If Late Start Type = 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.</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>
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.
Finished Late	System-supplied; this field is flagged if the task finished later than the time or duration specified in the Late Finish fields.
Late Finish Type	<p>Required if Late Finish is enabled.</p> <p>Options:</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 = Time; Time after which the task finish time is considered late. Use HH:MM, 24-hour time.
Late Finish Duration	If Late Finish Type = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
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.
Finished Early	System-supplied; this field is flagged if the task finished earlier than the time specified in the Early Finish fields.
Early Finish Type	<p>Required if Early Finish is enabled.</p> <p>Options:</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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Statistics	This section contains time-related statistics for the task instance.
User Estimated End Time	System-supplied; if the user entered information into the User Estimated Duration field in the task Details, the Controller uses this information to calculate an end time for the task instance, based on the date/time the task instance started.
Average Estimated End Time	System-supplied; average estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Shortest Estimated End Time	System-supplied; shortest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Longest Estimated End Time	System-supplied; longest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Buttons	This section identifies the buttons displayed above and below the Task Instance Details that let you perform various actions.
Update	Saves updates to the record.
Force Finish	See Force Finishing a Task .
Hold	Places the task instance on Hold (see Putting a Task on Hold).
Skip	For tasks loaded into the schedule that have not yet run; allows you to tell the Controller to skip this task. See Skipping a Task .
Re-run	See Re-running a Task .
View Parent	Displays the task instance Details for the parent Workflow of this task instance.
Retrieve Output	See Retrieving Output .
Delete	Deletes the current record.
Refresh	Refreshes any dynamic data displayed in the Details.
Close	For pop-up view only; closes the pop-up view of this task instance.
Tabs	This section identifies the tabs across the top of the Task Instance Details that provide access to additional information about the task instance.
Virtual Resources	Lists all Virtual Resources to which this task is assigned.

Exclusive Requests	Lists all records in the Exclusive Requests table (<code>ops_exclusive_order</code>) for this task instance.
Output	Displays output generated from the process, if any, based on specifications provided by the user in the Monitor Type fields in the task Details. Output also can be obtained by clicking the Retrieve Output button.
Notes	Lists all notes associated with this record.

Monitoring Task Execution

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

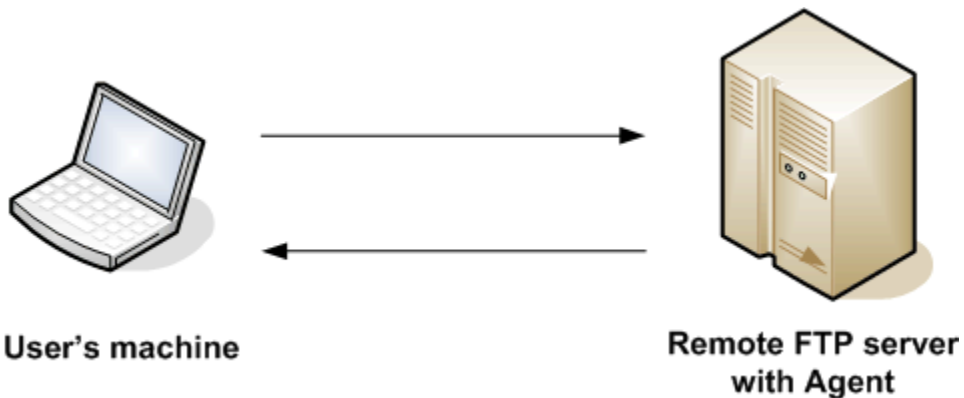
FTP File Monitor Task

- [Overview](#)
- [Creating an FTP File Monitor Task](#)
 - [FTP File Monitor Task Details](#)
 - [FTP File Monitor Task Details Field Descriptions](#)
- [Viewing an FTP File Monitor Task Instance](#)
 - [FTP File Monitor Task Instance Details](#)
 - [FTP File Monitor Task Instance Details Field Descriptions](#)
- [Running an FTP File Monitor Task](#)
- [Monitoring Task Execution](#)
- [Built-In Variables](#)

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 Opwise Universal 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.



Creating an FTP File Monitor Task

Step 1 From the [Automation Center](#) navigation pane, select **Tasks > FTP File Monitor Tasks**. The FTP File Monitor Tasks list displays a list of all currently defined FTP File Monitor tasks.

Below the list, FTP File Monitor Task Details for a new FTP File Monitor task displays. (You also can click the **New** button to display FTP File Monitor Task Details for a new FTP File Monitor task.)

Task Name	Task Description	Monitor Type	Updated By	Updated
stonebranch-ftpfilemonitortask-01		Exists	ops.admin	2014-06-13 14:21:21 -0400
stonebranch-ftpfilemonitortask-02		Exists	ops.admin	2014-06-13 14:21:29 -0400
stonebranch-ftpfilemonitortask-03		Exists	ops.admin	2014-06-13 14:21:33 -0400
stonebranch-ftpfilemonitortask-04		Exists	ops.admin	2014-06-13 14:21:38 -0400
stonebranch-ftpfilemonitortask-05		Exists	ops.admin	2014-06-13 14:21:42 -0400

FTP File Monitor Details

General

Task Name: Version:

Task Description:

Member of Business Services:

Hold on Start:

Virtual Resource Priority: Hold Resources on Failure:

FTP File Monitor Details

Agent: Agent Cluster:

Agent Variable: Agent Cluster Variable:

Credentials: Cluster Broadcast:

Credentials Variable:

Step 2 Enter/select Details for a new FTP File Monitor task, using the [field descriptions](#) below as a guide.

- Required fields display in **boldface**.
- Default values for fields, if available, display automatically.

To display more of the Details fields on the screen, you can temporarily [hide the list](#).



Note

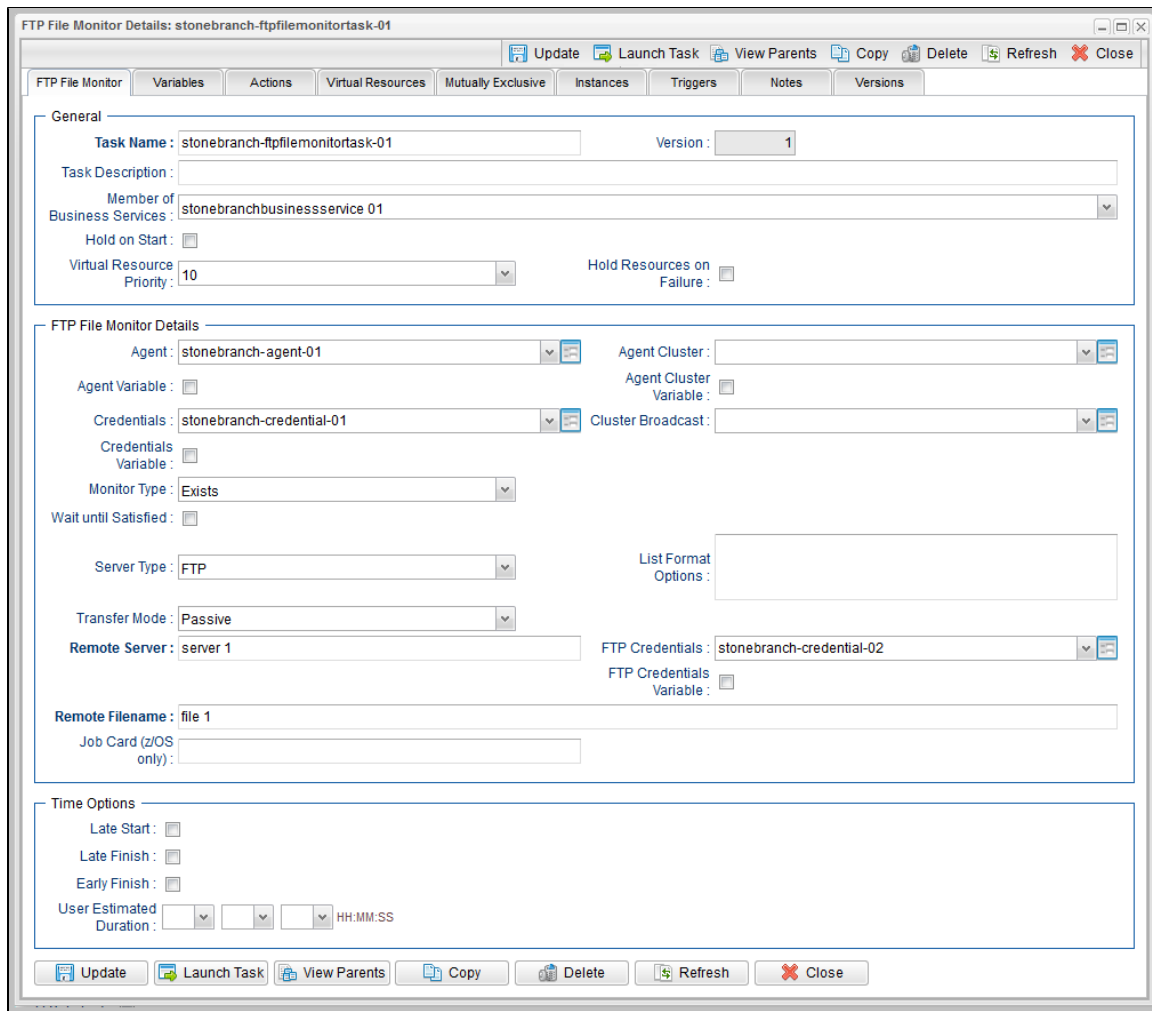
If you view [FTP File Monitor Task Details](#) for an existing File Monitor task by clicking a task in the list, and then want to create a new FTP File Monitor task, you must click the **New** button that displays above and below the Details.

Step 3 Click the **Save** button. The task is added to the database, and all buttons and tabs in the Task Details are enabled.

FTP File Monitor Task Details

The following FTP File Monitor Task Details is for an existing File Monitor task.



Depending on the values that you enter / select for these fields, and whether or not the FTP File Monitor task has ever been launched, more (or less) fields may display. See the [field descriptions](#), below, for a description of all fields that may display in the FTP File Monitor Task Details.




FTP File Monitor Task Details Field Descriptions


The following table describes the fields, buttons, and tabs that display in the FTP File Monitor Task Details.

Field Name	Description
General	This section contains general information about the task.
Task Name	User-defined name of this task, containing a maximum 255 alphanumeric characters; variables supported. It is the responsibility of the user to develop a workable naming scheme for tasks.
Version	System-supplied; version number of the current record, which is incremented by the Controller every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
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.
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.
File Monitor Details	This section contains assorted detailed information about the task.
Agent	Name of the Agent resource that identifies the machine where the operation will run. If you do not specify an Agent, you must specify an Agent Cluster .
Agent Variable	If enabled, 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. Use the format: <code>\${variable name}</code> . The variable must be a supported type as described in Variables and Functions . <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Agent Variable when updating multiple tasks, the Agent field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Agent Unresolved - into which you enter a variable, variable and text, or Agent name. The Agent field is not used.</p> </div>
Agent Cluster	Group of Agents, one of which the Controller will choose to run this task. You can specify an agent cluster in addition to or in place of a specific Agent. If you specify an Agent and an agent cluster, the Controller first tries to run the task on the specific agent. If the Agent is not available, the Controller reverts to the agent cluster. See Agent Clusters for more information.
Agent Cluster Variable	If enabled, 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. Use the format: <code>\${variable name}</code> . The variable must be a supported type as described in Variables and Functions . <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Agent Cluster Variable when updating multiple tasks, the Agent Cluster field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Agent Cluster Unresolved - into which you enter a variable, variable and text, or Agent Cluster name. The Agent Cluster field is not used.</p> </div>
Cluster Broadcast	Group of Agents, all of which will run this task. You can specify a Cluster Broadcast in place of a specific Agent and/or agent cluster. Each instance of the task running on its own Agent becomes a separate task instance record in the database and displays separately on the Activity screen. See Agent Clusters for more information about defining agent clusters.
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.

<p>Credentials Variable</p>	<p>If enabled, the Credentials field 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> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Credentials Variable when updating multiple tasks, the Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Credentials Unresolved - into which you enter a variable, variable and text, or Credentials name. The Credentials field is not used.</p> </div>
<p>Monitor Type</p>	<p>Type of file event being monitored for.</p> <p>Options:</p> <ul style="list-style-type: none"> • Exists - Checks to see if the file exists. • Missing - Checks to see if the file does not exist.
<p>Wait until Satisfied</p>	<p>If enabled, the task instance starts and continues to run until one of the following events occurs:</p> <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.
<p>Poll Interval (Seconds)</p>	<p>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.</p>
<p>Maximum Polls</p>	<p>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.</p>
<p>Stable (Seconds)</p>	<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> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <pre>-rwxr-xr-x 1 owner group 12345 Jan 1 2012 somefile.txt</pre> </div> <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 is provided to insert those statements into the FTP script that the file monitor task executes.</p>

Server Type	<p>Type of FTP server.</p> <p>Options:</p> <ul 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: 5px; 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: 5px; 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 in this field:</p> <pre style="border: 1px solid #ccc; padding: 5px; 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 this field:</p> <pre style="border: 1px solid #ccc; padding: 5px; 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>

Transfer Mode	<p>Transfer mode.</p> <p>Options:</p> <ul style="list-style-type: none"> • Active • Passive • Extended Passive
Remote Server	<p>Name or IP address of the remote server. This machine may or may not be the same as the Opswise Universal 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 that the Agent will use to access the FTP or SFTP server machine. If the File Transfer server and Agent are running on the same machine, enter the same credentials as those you entered in the Credentials field.</p>
FTP Credentials Variable	<p>If enabled, the FTP Credentials field 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> <div data-bbox="360 856 1446 1039" style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select FTP Credentials Variable when updating multiple tasks, the FTP Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - FTP Credentials Unresolved - into which you enter a variable, variable and text, or FTP Credentials name. The FTP Credentials field is not used.</p> </div>
Remote Filename	<p>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> <div data-bbox="360 1224 1446 1333" style="border: 1px solid #ccc; padding: 5px; background-color: #f0f0f0;"> <pre>//File TransferJOB01 JOB (File Transfer,001),FANNY,MSGCLASS=X,MSGLEVEL=(1,1),NOTIFY=&SYSUID,CLASS=A</pre> </div>
Time Options	<p>This section contains time-related specifications for the task.</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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.</p>
Late Start Type	<p>Required if Late Start is enabled.</p> <p>Options:</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>If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.</p>

Late Start Duration	<p>If Late Start Type = 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.</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>
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.</p> <p>Options:</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 = Time; 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 = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.</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.</p> <p>Options:</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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.</p>
User Estimated Duration	<p>Estimated amount of time it should normally take to run this task. The Controller uses this information to calculate the User Estimated End Time on a task instance record.</p>
Statistics	<p>This section contains time-related statistics for task instances of the task.</p>
First Time Ran	<p>System-supplied; date and time this task first ran.</p>

Last Time Ran	System-supplied; date and time the task last ran.
Last Instance Duration	System-supplied; amount of time the task took to run the last time it ran.
Lowest Instance Time	System-supplied; shortest amount of time this task has taken to run.
Average Instance Time	System-supplied; average amount of time this task takes to run.
Highest Instance Time	System-supplied; longest amount of time this task has taken to run.
Number of Instances	System-supplied; number of instances in the database for this task.
Buttons	This section identifies the buttons displayed above and below the Task Details that let you perform various actions.
Save	Saves a new task record in the Controller database.
Update	Saves updates to the record.
New	Displays empty (except for default values) Details for creating a new task.
Launch Task	Manually launches the task.
View Parents	Displays a list of any parent Workflow tasks for this task.
Copy	Creates a copy of this task, which you are prompted to rename.
Delete	<p>Deletes the current record.</p> <div style="background-color: #ffffcc; padding: 10px; margin: 10px 0;"> <p> Note You cannot delete a task if it is either:</p> <ul style="list-style-type: none"> Specified in an enabled Trigger. The only task specified in a disabled Trigger. </div>
Refresh	Refreshes any dynamic data displayed in the Details.
Close	For pop-up view only; closes the pop-up view of this task.
Tabs	This section identifies the tabs across the top of the Task Details that provide access to additional information about the task.
Variables	Lists all variables associated with this record.

Actions	<p>Allows you to specify actions that the Controller will take automatically based on events that occur during the execution of this task.</p> <p>Events are:</p> <ul style="list-style-type: none"> • Task instance status • Exit codes • Late start • Late finish • Early finish <p>Actions are:</p> <table border="1" data-bbox="321 478 1484 840"> <tr> <td data-bbox="321 478 461 548">Abort Action</td> <td data-bbox="461 478 1484 548">Abort the task if certain events occur. For details, see Abort Actions.</td> </tr> <tr> <td data-bbox="321 548 461 617">Email Notification</td> <td data-bbox="461 548 1484 617">Send an email if certain events occur. For details, see Email Notification Actions.</td> </tr> <tr> <td data-bbox="321 617 461 686">Set Variable</td> <td data-bbox="461 617 1484 686">Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow.</td> </tr> <tr> <td data-bbox="321 686 461 756">SNMP Notification</td> <td data-bbox="461 686 1484 756">Send an email if certain events occur. For details, see SNMP Notification Actions.</td> </tr> <tr> <td data-bbox="321 756 461 825">System Operation</td> <td data-bbox="461 756 1484 825">Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions.</td> </tr> </table>	Abort Action	Abort the task if certain events occur. For details, see Abort Actions .	Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .	Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .	SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .	System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .
Abort Action	Abort the task if certain events occur. For details, see Abort Actions .										
Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .										
Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .										
SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .										
System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .										
Virtual Resources	Lists all Virtual Resources to which this task is assigned.										
Mutually Exclusive	Lists all tasks that have been set to be mutually exclusive of this task.										
Instances	Lists all instances of this Workflow.										
Triggers	Lists 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, the Controller 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 Triggers Overview .										
Notes	Lists all notes associated with this record.										
Versions	Stores copies of all previous versions of the current record. See Record Versioning .										

Viewing an FTP File Monitor Task Instance

When an FTP File Monitor task is launched, the Controller creates a task instance record of that task.

A task instance contains detailed information about a single execution of that task.

You can access a task instance from:

- **Instances tab** on the [FTP File Monitor Task Details](#) for that task
- [Activity Monitor](#)
- [Task Instances list](#)

FTP File Monitor Task Instance Details




The following FTP File Monitor Task Instance Details contains information on the execution of the task shown in the [FTP File Monitor Task Details](#)

FTP File Monitor Task Instance Details Field Descriptions

The following table describes the fields, buttons, and tabs that display in FTP File Monitor Task Instance Details.

Field Name	Description
General	This section contains general information about the task instance.
Instance Name	Name of this task instance.
Reference Id	System-supplied; the Controller increments this number each time the task is run.
Task	Name of the task that was run to create this task instance. Click the icon to display Task Details for the task.

Invoked by	<p>System-supplied; how the task instance was launched.</p> <p>Options:</p> <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 identify the user, check the Execution User column for that task instance on the Task Instances screen or, on most task instance screens, the Execution User field.
Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
Execution User	System-supplied; if the task was launched manually, the ID of the user who launched it.
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.
Status	This section contains information about the current status of the task instance.
Status	System-supplied; see Task Instance Statuses .
Exit Code	System-supplied; the exit code captured by the Agent when executing the task (for example, a command or script).
Status Description	System-supplied; additional information, if any, about the status of the task instance.
Queued Time	System-supplied; Date and time the task was queued for processing.
Trigger Time	System-supplied; Date and time the task instance was triggered.
Launch Time	System-supplied; Date and time the task instance was launched.
Start Time	System-supplied; Date and time the task instance started.
End Time	System-supplied; Date and time the task instance completed.
Duration	System-supplied; amount of time the task instance took to run.
File Monitor Details	This section contains assorted detailed information about the task instance.

Agent	Name of the Agent resource that identifies the machine where the operation will run. If you do not specify an Agent, you must specify an Agent Cluster .
Agent Variable	<p>If enabled, 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. Use the format: <code>\${variable name}</code>. The variable must be a supported type as described in Variables and Functions.</p> <div data-bbox="360 401 1446 583" style="background-color: #ffffcc; padding: 10px;"> <p> Note If you select Agent Variable when updating multiple tasks, the Agent field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Agent Unresolved - into which you enter a variable, variable and text, or Agent name. The Agent field is not used.</p> </div>
Agent Cluster	Group of Agents, one of which the Controller will choose to run this task. You can specify an agent cluster in addition to or in place of a specific Agent. If you specify an Agent and an agent cluster, the Controller first tries to run the task on the specific agent. If the Agent is not available, the Controller reverts to the agent cluster. See Agent Clusters for more information.
Agent Cluster Variable	<p>If enabled, 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. Use the format: <code>\${variable name}</code>. The variable must be a supported type as described in Variables and Functions.</p> <div data-bbox="360 936 1446 1119" style="background-color: #ffffcc; padding: 10px;"> <p> Note If you select Agent Cluster Variable when updating multiple tasks, the Agent Cluster field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Agent Cluster Unresolved - into which you enter a variable, variable and text, or Agent Cluster name. The Agent Cluster field is not used.</p> </div>
Cluster Broadcast	Group of Agents, all of which will run this task. You can specify a Cluster Broadcast in place of a specific Agent and/or agent cluster. Each instance of the task running on its own Agent becomes a separate task instance record in the database and displays separately on the Activity screen. See Agent Clusters for more information about defining agent clusters.
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	<p>If enabled, the Credentials field 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> <div data-bbox="360 1566 1446 1749" style="background-color: #ffffcc; padding: 10px;"> <p> Note If you select Credentials Variable when updating multiple tasks, the Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Credentials Unresolved - into which you enter a variable, variable and text, or Credentials name. The Credentials field is not used.</p> </div>

Monitor Type	<p>Type of file event being monitored for.</p> <p>Options:</p> <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	<p>If enabled, the task instance starts and continues to run until one of the following events occurs:</p> <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)	<p>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.</p>
Maximum Polls	<p>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.</p>
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 style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;">-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 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.</p> <p>Options:</p> <ul style="list-style-type: none"> • FTP • SFTP

List Format Options

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.

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:

```
site listfmt 1
site namefmt 1
```

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.

```
site dirstyle
```

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.

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 `/uagtests/data/somefile*.txt`, enter the following in this field:

```
dir /uagtests/data/somefile*.txt
```

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 this field:

```
site dirstyle
dir /uagtests/data/somefile*.txt
```


Invalid statements or valid statements that do not control the file list format are ignored.

Transfer Mode

Transfer mode.

Options:

- Active
- Passive
- Extended Passive

Remote Server	<p>Name or IP address of the File Transfer server. This machine may or may not be the same as the Opwise Universal 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 that the Agent will use to access the FTP or SFTP server machine. If the File Transfer server and Agent are running on the same machine, enter the same credentials as those you entered in the Credentials field.</p>
FTP Credentials Variable	<p>If enabled, the FTP Credentials field 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> <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note If you select FTP Credentials Variable when updating multiple tasks, the FTP Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - FTP Credentials Unresolved - into which you enter a variable, variable and text, or FTP Credentials name. The FTP Credentials field is not used.</p> </div>
Remote Filename	<p>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> <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>
z/OS ID (z/OS only)	<p>z/OS execID, used internally by the z/OS Agent to identify each z/OS task.</p>
Time Options	<p>This section contains time-related specifications for the task instance.</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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.</p>
Started Late	<p>System-supplied; this field is flagged if the task started later than the time specified in the Late Start fields.</p>
Late Start Type	<p>Required if Late Start is enabled.</p> <p>Options:</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>If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.</p>

Late Start Duration	<p>If Late Start Type = 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.</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>
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>
Finished Late	<p>System-supplied; this field is flagged if the task finished later than the time or duration specified in the Late Finish fields.</p>
Late Finish Type	<p>Required if Late Finish is enabled.</p> <p>Options:</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 = Time; 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 = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.</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>
Finished Early	<p>System-supplied; this field is flagged if the task finished earlier than the time specified in the Early Finish fields.</p>
Early Finish Type	<p>Required if Early Finish is enabled.</p> <p>Options:</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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.</p>
Statistics	<p>This section contains time-related statistics for the task instance.</p>

User Estimated End Time	System-supplied; if the user entered information into the User Estimated Duration field in the task Details, the Controller uses this information to calculate an end time for the task instance, based on the date/time the task instance started.
Average Estimated End Time	System-supplied; average estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Shortest Estimated End Time	System-supplied; shortest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Longest Estimated End Time	System-supplied; longest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Buttons	This section identifies the buttons displayed above and below the Task Instance Details that let you perform various actions.
Update	Saves updates to the record.
Force Finish	See Force Finishing a Task .
Hold	Places the task instance on Hold (see Putting a Task on Hold).
Skip	For tasks loaded into the schedule that have not yet run; allows you to tell the Controller to skip this task. See Skipping a Task .
Re-run	See Re-running a Task .
View Parent	Displays the task instance Details for the parent Workflow of this task instance.
Retrieve Output	See Retrieving Output .
Delete	Deletes the current record.
Refresh	Refreshes any dynamic data displayed in the Details.
Close	For pop-up view only; closes the pop-up view of this task instance.
Tabs	This section identifies the tabs across the top of the Task Instance Details that provide access to additional information about the task instance.
Virtual Resources	Lists all Virtual Resources to which this task is assigned.
Exclusive Requests	Lists all records in the Exclusive Requests table (<code>ops_exclusive_order</code>) for this task instance.
Output	Displays output generated from the process, if any, based on specifications provided by the user in the Monitor Type fields in the task Details. Output also can be obtained by clicking the Retrieve Output button.
Notes	Lists all notes associated with this record.

Running an FTP File Monitor Task

You can run an FTP File Monitor task:

- Manually, by clicking the [Launch Task](#) or [Launch Task with Variables](#) button in the FTP File Monitor Tasks list or FTP File Monitor Task Details [Action menu](#).
- As part of a [Workflow](#).
- [Specify triggers](#) that run the task automatically based on times or events.

Monitoring Task Execution

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

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](#).

System Monitor Task

- Overview
- Creating a System Monitor Task
 - System Monitor Task Details
 - System Monitor Task Details Field Descriptions
- Viewing a System Monitor Task Instance
 - System Monitor Task Instance Details
 - System Monitor Task Instance Details Field Descriptions
- Running a System Monitor Task
- 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 Opwise Universal Agent running on it.

Creating a System Monitor Task

Step 1 From the [Automation Center](#) navigation pane, select **Tasks > System Monitor Tasks**. The System Monitor Tasks list displays a list of all currently defined System Monitor tasks.

Below the list, System Monitor Task Details for a new System Monitor task displays. (You also can click the **New** button to display System Monitor Task Details for a new System Monitor task.)

The screenshot shows the 'System Monitors' section of the Opwise Controller. At the top, there is a list of 5 System Monitors with columns for Task Name, Task Description, Monitor Type, Updated By, and Updated. Below the list, the 'System Monitor Details' form is displayed, which includes tabs for System Monitor, Variables, Actions, Virtual Resources, Mutually Exclusive, Instances, Triggers, Notes, and Versions. The 'System Monitor' tab is active, showing fields for Task Name, Version (1), Task Description, Member of Business Services, Hold on Start, Virtual Resource Priority (10), Hold Resources on Failure, Agent, Agent Cluster, Agent Variable, Agent Cluster Variable, Credentials, and Cluster Broadcast.

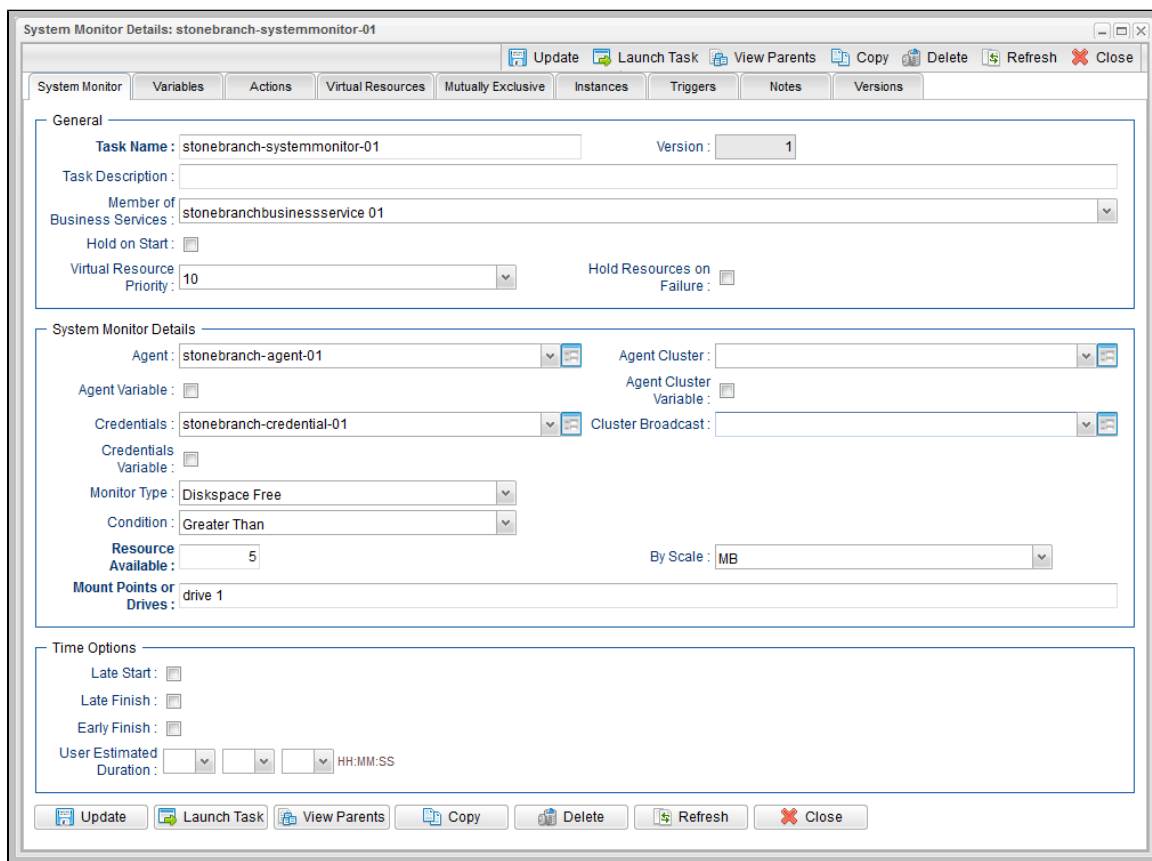
Task Name	Task Description	Monitor Type	Updated By	Updated
stonebranch-systemmonitor-01		Diskspace Free	ops.admin	2014-06-13 14:22:34 -0400
stonebranch-systemmonitor-02		Diskspace Free	ops.admin	2014-06-13 14:22:43 -0400
stonebranch-systemmonitor-03		Diskspace Free	ops.admin	2014-06-13 14:22:49 -0400
stonebranch-systemmonitor-04		Diskspace Free	ops.admin	2014-06-13 14:22:53 -0400
stonebranch-systemmonitor-05		Diskspace Free	ops.admin	2014-06-13 14:22:57 -0400

Step 2	<p>Enter/select Details for a new System Monitor task, using the field descriptions below as a guide.</p> <ul style="list-style-type: none"> • Required fields display in boldface. • Default values for fields, if available, display automatically. <p>To display more of the Details fields on the screen, you can temporarily hide the list.</p> <div style="background-color: #ffffcc; padding: 10px; margin: 10px 0;"> <p>Note If you view System Monitor Task Details for an existing System Monitor task by clicking a task in the list, and then want to create a new System Monitor task, you must click the New button that displays above and below the Details.</p> </div>
Step 3	<p>Click the Save button. The task is added to the database, and all buttons and tabs in the Task Details are enabled.</p>

System Monitor Task Details

The following System Monitor Task Details is for an existing System Monitor task.


Depending on the values that you enter / select for these fields, and whether or not the System Monitor task has ever been launched, more (or less) fields may display. See the [field descriptions](#), below, for a description of all fields that may display in the System Monitor Task Details.





System Monitor Task Details Field Descriptions

The following table describes the fields, buttons, and tabs that display in the System Monitor Task Details.


Field Name	Description
General	This section contains general information about the task.

Task Name	User-defined name of this task, containing a maximum 255 alphanumeric characters; variables supported. It is the responsibility of the user to develop a workable naming scheme for tasks.
Version	System-supplied; version number of the current record, which is incremented by the Controller every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
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.
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.
System Monitor Details	This section contains assorted detailed information about the task.
Agent	Name of the Agent resource that identifies the machine where the operation will run. If you do not specify an Agent, you must specify an Agent Cluster .
Agent Variable	If enabled, 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. Use the format: <code>\${variable name}</code> . The variable must be a supported type as described in Variables and Functions . <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Agent Variable when updating multiple tasks, the Agent field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Agent Unresolved - into which you enter a variable, variable and text, or Agent name. The Agent field is not used.</p> </div>
Agent Cluster	Group of Agents, one of which the Controller will choose to run this task. You can specify an agent cluster in addition to or in place of a specific Agent. If you specify an Agent and an agent cluster, the Controller first tries to run the task on the specific agent. If the Agent is not available, the Controller reverts to the agent cluster. See Agent Clusters for more information.

Agent Cluster Variable	<p>If enabled, 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. Use the format: <code>\${variable name}</code>. The variable must be a supported type as described in Variables and Functions.</p> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Agent Cluster Variable when updating multiple tasks, the Agent Cluster field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Agent Cluster Unresolved - into which you enter a variable, variable and text, or Agent Cluster name. The Agent Cluster field is not used.</p> </div>
Cluster Broadcast	<p>Group of Agents, all of which will run this task. You can specify a Cluster Broadcast in place of a specific Agent and/or agent cluster. Each instance of the task running on its own Agent becomes a separate task instance record in the database and displays separately on the Activity screen. See Agent Clusters for more information about defining agent clusters.</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>
Credentials Variable	<p>If enabled, the Credentials field 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> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Credentials Variable when updating multiple tasks, the Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Credentials Unresolved - into which you enter a variable, variable and text, or Credentials name. The Credentials field is not used.</p> </div>
Monitor Type	<p>Type of system status to monitor for.</p> <p>Options:</p> <ul style="list-style-type: none"> • Diskspace Free
Condition	<p>Specifies whether you want to check for free disk space greater than or less than the amount specified in the Resource Available field.</p>
Resource Available	<p>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 1GB of free disk space, select Greater Than in the Condition field, enter 1 in the Resource Available field, and select GB in the By Scale field.</p>
By Scale	<p>Scale of the number you entered in the Resource Available field. Options: KB (kilobyte), MB (megabyte), GB (gigabyte).</p>
Mount Points or Drives	<p>Use this field to limit the check to a specific mount point or drive, such as drive C: for Windows.</p>
Time Options	<p>This section contains time-related specifications for the task.</p>

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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.
Late Start Type	Required if Late Start is enabled. Options: <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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.
Late Start Duration	If Late Start Type = 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.
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: <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 = Time; Time after which the task finish time is considered late. Use HH:MM, 24-hour time.
Late Finish Duration	If Late Finish Type = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
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: <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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
User Estimated Duration	Estimated amount of time it should normally take to run this task. The Controller uses this information to calculate the User Estimated End Time on a task instance record.
Statistics	This section contains time-related statistics for task instances of the task.
First Time Ran	System-supplied; date and time this task first ran.
Last Time Ran	System-supplied; date and time the task last ran.
Last Instance Duration	System-supplied; amount of time the task took to run the last time it ran.
Lowest Instance Time	System-supplied; shortest amount of time this task has taken to run.
Average Instance Time	System-supplied; average amount of time this task takes to run.
Highest Instance Time	System-supplied; longest amount of time this task has taken to run.
Number of Instances	System-supplied; number of instances in the database for this task.
Buttons	This section identifies the buttons displayed above and below the Task Details that let you perform various actions.
Save	Saves a new task record in the Controller database.
Update	Saves updates to the record.
New	Displays empty (except for default values) Details for creating a new task.
Launch Task	Manually launches the task.
View Parents	Displays a list of any parent Workflow tasks for this task.
Copy	Creates a copy of this task, which you are prompted to rename.

Delete	<p>Deletes the current record.</p> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note You cannot delete a task if it is either:</p> <ul style="list-style-type: none"> Specified in an enabled Trigger. The only task specified in a disabled Trigger. </div>										
Refresh	Refreshes any dynamic data displayed in the Details.										
Close	For pop-up view only; closes the pop-up view of this task.										
Tabs	This section identifies the tabs across the top of the Task Details that provide access to additional information about the task.										
Variables	Lists all variables associated with this record.										
Actions	<p>Allows you to specify actions that the Controller will take automatically based on events that occur during the execution of this task.</p> <p>Events are:</p> <ul style="list-style-type: none"> Task instance status Exit codes Late start Late finish Early finish <p>Actions are:</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="background-color: #e0e0e0;">Abort Action</td> <td>Abort the task if certain events occur. For details, see Abort Actions.</td> </tr> <tr> <td style="background-color: #e0e0e0;">Email Notification</td> <td>Send an email if certain events occur. For details, see Email Notification Actions.</td> </tr> <tr> <td style="background-color: #e0e0e0;">Set Variable</td> <td>Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow.</td> </tr> <tr> <td style="background-color: #e0e0e0;">SNMP Notification</td> <td>Send an email if certain events occur. For details, see SNMP Notification Actions.</td> </tr> <tr> <td style="background-color: #e0e0e0;">System Operation</td> <td>Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions.</td> </tr> </table>	Abort Action	Abort the task if certain events occur. For details, see Abort Actions .	Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .	Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .	SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .	System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .
Abort Action	Abort the task if certain events occur. For details, see Abort Actions .										
Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .										
Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .										
SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .										
System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .										
Virtual Resources	Lists all Virtual Resources to which this task is assigned.										
Mutually Exclusive	Lists all tasks that have been set to be mutually exclusive of this task.										
Instances	Lists all instances of this Workflow.										
File Monitor Triggers	Lists 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, the Controller 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 Triggers Overview .										
Notes	Lists all notes associated with this record.										

Versions	Stores copies of all previous versions of the current record. See Record Versioning .
-----------------	---

Viewing a System Monitor Task Instance

When a System Monitor task is launched, the Controller creates a task instance record of that task.

A task instance contains detailed information about a single execution of that task.

You can access a task instance from:

- **Instances tab** on the [System Monitor Task Details](#) for that task
- [Activity Monitor](#)
- [Task Instances list](#)

System Monitor Task Instance Details

The following System Monitor Task Instance Details contains information on the execution of the task shown in the [System Monitor Task Details](#).

The screenshot displays the 'System Monitor Task Instance Details' window for 'stonebranch-systemmonitor-01'. The window is divided into several sections:



- General:** Instance Name: stonebranch-systemmonitor-01, Reference Id: 6, Task: stonebranch-systemmonitor-01, Invoked By: Manually Launched, Task Description: (empty), Member of Business Services: stonebranchbusinessservice 01, Execution User: stonebranch-user-01, Virtual Resource Priority: 10, Hold Resources on Failure: (checkbox).
- Status:** Status: Success, Exit Code: 0, Status Description: (empty), Queued Time: 2015-02-06 14:45:37 -0500, Trigger Time: (empty), Start Time: 2015-02-06 14:45:41 -0500, Launch Time: 2015-02-06 14:45:37 -0500, End Time: 2014-06-20 13:09:49 -0400, Actual Available: 3.6, By Scale: GB.
- System Monitor Details:** agentid: (empty), Agent Cluster: (empty), Agent Variable: (checkbox), Agent Cluster Variable: (checkbox), Credentials: stonebranch-credential-01, Credentials Variable: (checkbox), Monitor Type: Diskspace Free, Condition: Greater Than, Resource Available: 1, By Scale: MB, Mount Points or Drives: C:.
- Statistics:** User Estimated End Time: (empty), Average Estimated End Time: 2014-06-09 15:11:42 -0400, Shortest Estimated End Time: 2014-06-09 15:11:42 -0400, Longest Estimated End Time: 2014-06-09 15:11:42 -0400.


Buttons for Update, Re-run, View Parent, Delete, Refresh, and Close are located at the top and bottom of the window.

System Monitor Task Instance Details Field Descriptions

The following table describes the fields, buttons, and tabs that display in System Monitor Task Instance Details.

Field Name	Description
General	This section contains general information about the task instance.
Instance Name	Name of this task instance.
Reference Id	System-supplied; the Controller increments this number each time the task is run.
Task	Name of the task that was run to create this task instance. Click the icon to display Task Details for the task.
Invoked by	System-supplied; how the task instance was launched. Options: <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 identify the user, check the Execution User column for that task instance on the Task Instances screen or, on most task instance screens, the Execution User field.
Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
Execution User	System-supplied; if the task was launched manually, the ID of the user who launched it.
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.
Status	This section contains information about the current status of the task instance.
Status	System-supplied; see Task Instance Statuses .
Exit Code	System-supplied; the exit code captured by the Agent when executing the task (for example, a command or script).
Status Description	System-supplied; additional information, if any, about the status of the task instance.
Queued Time	System-supplied; Date and time the task was queued for processing.
Trigger Time	System-supplied; Date and time the task instance was triggered.

Launch Time	System-supplied; Date and time the task instance was launched.
Start Time	System-supplied; Date and time the task instance started.
End Time	System-supplied; Date and time the task instance completed.
Duration	System-supplied; amount of time the task instance took to run.
Actual Available	Amount of free disk space on the specified system.
By Scale	Scale of the number show in the Actual Available field. Options: <ul style="list-style-type: none"> • KB (kilobyte) • MB (megabyte) • GB (gigabyte)
System Monitor Details	This section contains assorted detailed information about the task instance.
Agent	Name of the Agent resource that identifies the machine where the operation will run. If you do not specify an Agent, you must specify an Agent Cluster .
Agent Variable	If enabled, 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. Use the format: \${variable name}. The variable must be a supported type as described in Variables and Functions . <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Agent Variable when updating multiple tasks, the Agent field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Agent Unresolved - into which you enter a variable, variable and text, or Agent name. The Agent field is not used.</p> </div>
Agent Cluster	Group of Agents, one of which the Controller will choose to run this task. You can specify an agent cluster in addition to or in place of a specific Agent. If you specify an Agent and an agent cluster, the Controller first tries to run the task on the specific agent. If the Agent is not available, the Controller reverts to the agent cluster. See Agent Clusters for more information.
Agent Cluster Variable	If enabled, 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. Use the format: \${variable name}. The variable must be a supported type as described in Variables and Functions . <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Agent Cluster Variable when updating multiple tasks, the Agent Cluster field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Agent Cluster Unresolved - into which you enter a variable, variable and text, or Agent Cluster name. The Agent Cluster field is not used.</p> </div>

Cluster Broadcast	Group of Agents, all of which will run this task. You can specify a Cluster Broadcast in place of a specific Agent and/or agent cluster. Each instance of the task running on its own Agent becomes a separate task instance record in the database and displays separately on the Activity screen. See Agent Clusters for more information about defining agent clusters.
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	<p>If enabled, the Credentials field 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> <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note If you select Credentials Variable when updating multiple tasks, the Credentials field does not convert from a reference field into a text field. Instead, the Update / Update Filtered dialog contains an additional field - Credentials Unresolved - into which you enter a variable, variable and text, or Credentials name. The Credentials field is not used.</p> </div>
Monitor Type	<p>Type of file event being monitored for.</p> <p>Options:</p> <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.
Condition	Specifies whether you want to check for free disk space greater than or less than the amount specified in the Resource Available field.
Resource Available	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 1GB of free disk space, select Greater Than in the Condition field, enter 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	Use this field to limit the check to a specific mount point or drive, such as drive C: for Windows.
Time Options	This section contains time-related specifications for the task instance.
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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.
Started Late	System-supplied; this field is flagged if the task started later than the time specified in the Late Start fields.

Late Start Type	<p>Required if Late Start is enabled.</p> <p>Options:</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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.
Late Start Duration	<p>If Late Start Type = 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.</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>
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.
Finished Late	System-supplied; this field is flagged if the task finished later than the time or duration specified in the Late Finish fields.
Late Finish Type	<p>Required if Late Finish is enabled.</p> <p>Options:</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 = Time; Time after which the task finish time is considered late. Use HH:MM, 24-hour time.
Late Finish Duration	If Late Finish Type = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
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.
Finished Early	System-supplied; this field is flagged if the task finished earlier than the time specified in the Early Finish fields.
Early Finish Type	<p>Required if Early Finish is enabled.</p> <p>Options:</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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Statistics	This section contains time-related statistics for the the task instance.
User Estimated End Time	System-supplied; if the user entered information into the User Estimated Duration field in the task Details, the Controller 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	System-supplied; shortest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Average Estimated End Time	System-supplied; average estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Longest Estimated End Time	System-supplied; longest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Buttons	This section identifies the buttons displayed above and below the Task Instance Details that let you perform various actions.
Update	Saves updates to the record.
Force Finish	See Force Finishing a Task .
Hold	Places the task instance on Hold (see Putting a Task on Hold).
Skip	For tasks loaded into the schedule that have not yet run; allows you to tell the Controller to skip this task. See Skipping a Task .
Re-run	See Re-running a Task .
View Parent	Displays the task instance Details for the parent Workflow of this task instance.
Retrieve Output	See Retrieving Output .
Delete	Deletes the current record.
Refresh	Refreshes any dynamic data displayed in the Details.
Close	For pop-up view only; closes the pop-up view of this task instance.
Tabs	This section identifies the tabs across the top of the Task Instance Details that provide access to additional information about the task instance.
Virtual Resources	Lists all Virtual Resources to which this task is assigned.

Exclusive Requests	Lists all records in the Exclusive Requests table (<code>ops_exclusive_order</code>) for this task instance.
Output	Displays output generated from the process, if any, based on specifications provided by the user in the Monitor Type fields in the task Details. Output also can be obtained by clicking the Retrieve Output button.
Notes	Lists all notes associated with this record.

Running a System Monitor Task

You can run a System Monitor task:

- Manually, by clicking the [Launch Task](#) or [Launch Task with Variables](#) button in the System Monitor Tasks list or System Monitor Task Details [Action](#) menu.
- As part of a [workflow](#).
- [Specify triggers](#) that run the task automatically based on times or events.

Monitoring Task Execution

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

Creating Task Actions

Opswise Controller lets you create the following actions for tasks and workflows:

Action Type	Description
Abort	Allows you to abort a waiting or running task instance
Email Notification	Allows you to generate email notifications based on various events and statuses.
Set Variable	Allows you to set a variable to a specific value for a task or workflow.
SNMP Notification	Allows you to generate SNMP notifications to be sent to an SNMP Manager.
System Operation	Allows you to run an Opswise Controller system operation based on specified conditions.

Abort Actions

- [Overview](#)
- [Creating an Abort Action](#)
- [Abort Action Details Field Descriptions](#)

Overview

The Abort Action allows you to abort a waiting or running task instance either by:

- Issuing a Skip command on a waiting task instance.
- 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 create one or more Abort Actions for any Opswise Controller task. For Workflow tasks, you can also specify whether you want the Abort Action instructions to apply to the workflow itself, the workflow an/or its tasks, or to the tasks only.

Creating an Abort Action

Step 1 Display the Task Details of the task for which you are creating the Abort Action.

Step 2 Click the **Actions** tab. A list of any defined Actions for that task displays.

Status	Description	Type Details	Exit Codes	On Late Start	On Late Finish	On Early Finish	Cancel Process if Active	Override Exit Code	Updated By	Updated
FAILED			20	No	No	No	No		ops.admin	2014-06-30 10:02
CANCELLED			12	No	No	No	No		ops.admin	2014-06-30 10:02

Step 3 Click the **New** button that displays on the Abort Actions row. The Abort Action Details pop-up displays.

Step 4 Using the field descriptions, below, as a guide, complete the fields as needed.


Step 5 Click the **Save** button, or right-click in the Details and then click **Save**, to save the record and return to the Actions List.

Step 6 If appropriate, repeat these steps for any additional Abort Actions you want to create.

Abort Action Details Field Descriptions

The following table describes the fields and buttons that display in the Abort Actions Details.

Field Name	Description
Action Criteria	This section contains criteria for performing the action.
Action Inheritance	For Workflow tasks only; the records this action applies to. Options: <ul style="list-style-type: none"> • SELF - This action applies only to the workflow; it is not inherited by its children tasks. • SELF/CHILDREN - This action applies to the workflow and its contained tasks (children). • CHILDREN - This action applies only to the tasks within the workflow (children).
Status	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.
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	Generates the action or notification if the task started late, based on the Late Start Time specified in the task.
On Late Finish	Generates the action or notification if the task finishes late, based on the Late Finish time specified in the task.
{anchor:On Early Finish	Generates the action or notification if the task finishes early, based on the Early Finish Time specified in the task.
Description	Description of this action.

Action Details	This section contains additional details about the action.
Cancel Process if Active	If enabled, instructs the Controller 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>
Buttons	This section identifies the buttons displayed above and below the Action Details that let you perform various actions.
Save	Saves a new Action record in the Controller database.
Update	Saves updates to the record.
Delete	Deletes the current record.
Refresh	Refreshes any dynamic data displayed in the Details.
Close	Closes the Details pop-up of this action.

Email Notification Actions

- [Overview](#)
- [Creating an Email Notification](#)
- [Email Notification Details Field Descriptions](#)

Overview

You can create one or more Email Notifications for any Opwise Controller 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 for which you create 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 also can generate notifications based on the status of [Agents and Agent clusters](#), [Cluster nodes](#), and [OMS Servers](#).

Creating an Email Notification

Step 1 Display the Task Details of the task for which you are creating the Email Notification.

Step 2 Click the **Actions** tab. A list of any defined Actions for that task displays.

Status	Description	Type Details	Exit Codes	On Late Start	On Late Finish	On Early Finish	Cancel Process if Active	Override Exit Code	Updated By	Updated
FAILED			20	No	No	No	No		ops.admin	2014-06-30 10:02
CANCELLED			12	No	No	No	No		ops.admin	2014-06-30 10:02

Step 3 Click the **New** button that displays on the Email Notifications row. The Email Notification Details pop-up displays.

Step 4 Using the field descriptions, below, as a guide, complete the fields as needed.

Step 5 Click the **Save** button, or right-click in the Details and then click **Save**, to save the record and return to the Actions List.

Step 6 If appropriate, repeat these steps for any additional Email Notifications you want to create.

Email Notification Details Field Descriptions

The table below describes the fields and buttons that display in the Email Notification Details.

Field Name	Description
Action Criteria	This section contains criteria for performing the action.
Action Inheritance	For Workflow tasks only; the records this action applies to. Options: <ul style="list-style-type: none"> • SELF - This action applies only to the workflow; it is not inherited by its children tasks. • SELF/CHILDREN - This action applies to the workflow and its contained tasks (children). • CHILDREN - This action applies only to the tasks within the workflow (children).
Status	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.

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	Generates the action or notification if the task started late, based on the Late Start Time specified in the task.
On Late Finish	Generates the action or notification if the task finishes late, based on the Late Finish time specified in the task.
On Early Finish	Generates the action or notification if the task finishes early, based on the Early Finish Time specified in the task.
Description	Description of this action.
Action Details Details	This section contains additional details about the action.
Email Template	<p>Name of an Email template defined in an Email Template Details. An Email template allows you to specify standard recipients and text for outgoing emails. Enter the name of an existing Email template, select an Email template from the drop-down list, or click the Details icon to create a new Email template.</p> <p>Every Email template specifies an Email connection. If you do not specify an Email template in this field, you must specify an Email connection in the Email Connection field.</p> <p>If you specify both an Email template (in this field) and an Email Connection, the Email server specified in the Email Connection field overrides the Email server specified in this field.</p> <p>(Any information specified in an Email task overrides what is specified in an Email template.)</p>
Email Connection	<p>Required if an Email template is not specified in the Email Template field; name of an Email connection defined via the Email Connection definition screen. An email connection specifies information about an email server. Enter the name of an existing Email connection, or click the magnifying glass to browse for an existing Email connection or create a new Email connection.</p> <p>If you specify both an Email Template and an Email connection (in this field), the Email connection specified in this field overrides the Email connection specified in the Email Template field.</p>
Reply-To	Email address of the sender. Use commas to separate multiple recipients. Variables and functions supported.
To	Email address of the recipient. Use commas to separate multiple recipients. Variables and functions supported.
CC	Email address of the party being sent a copy of the email, if any. Use commas to separate multiple recipients. Variables and functions supported.
BCC	Email address of the party being sent a blind (hidden) copy of the email, if any. Use commas to separate multiple recipients. Variables and functions supported.
Subject	Subject line of the email. Variables and functions supported.
Body	Text of the email message. Variables and functions supported. If both the email template and the email task contain text in the body, the text is appended.
Attach Standard Output	For Agent-based tasks only; attach any standard output generated by the associated task.

Attach Standard Error	For Agent-based tasks only; attach standard error data generated by the associated task.
Attach File	<p>For Agent-based tasks only (except z/OS); attach any single text file that is accessible by the Agent. Full path name is required. Wildcards are NOT supported.</p> <p>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.</p>
Attach Job Log	For z/OS tasks only; attach any job logs generated by the associated task.
Start Line	<p>Attach data beginning at the line indicated.</p> <ul style="list-style-type: none"> • If a Start Line value is not specified, the default is 1. • If the Start Line value is -1, data will be retrieved starting at the end of the file.
Number of Lines	If Automatic Output Retrieval = Standard Output, Standard Error, File, or Standard Output/Error; 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 Controller system property.
Scan Text:	Regex pattern that the Controller will search for a match for in STDOUT/STDERR or a specified file. The Controller will include the Number of Lines above and below the first line matched.
File Name	For Attach File only; path and file name of the file you want to attach to the email notification.
Buttons	This section identifies the buttons displayed above and below the Action Details that let you perform various actions.
Save	Saves a new Action record in the Controller database.
Update	Saves updates to the record.
Delete	Deletes the current record.
Refresh	Refreshes any dynamic data displayed in the Details.
Close	Closes the Details pop-up of this action.

Set Variable Actions

For information on how to create Set Variable actions for use within a task or workflow, see [Creating a Set Variable Action within a Task or Workflow](#) in *Variables and Functions*.

SNMP Notification Actions

- [Overview](#)
- [Creating an SNMP Notification](#)
- [SNMP Notification Details Field Descriptions](#)

Overview

You can create one or more SNMP notifications for any Opwise Controller 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 also can generate notifications based on the status of [Agents and Agent clusters](#), [Cluster nodes](#), and [OMS Servers](#).

Creating an SNMP Notification

Step 1 Display the Task Details of the task for which you are creating the SNMP Notification.

Step 2 Click the **Actions** tab. A list of any defined Actions for that task displays.

Linux/Unix Task Details: stonebranch-linuxunixtask-01

Linux/Unix Task | Variables | **Actions** | Virtual Resources | Mutually Exclusive | Instances | Triggers | Notes | Versions

2 Abort Actions

Status	Description	Type Details	Exit Codes	On Late Start	On Late Finish	On Early Finish	Cancel Process if Active	Override Exit Code	Updated By	Updated
FAILED			20	No	No	No	No		ops.admin	2014-06-30 10:02
CANCELLED			12	No	No	No	No		ops.admin	2014-06-30 10:02

0 Email Notifications | New

0 Set Variables | New

0 SNMP Notifications | New

0 System Operations | New

Step 3 Click the **New** button that displays on the SNMP Notifications row. The SNMP Notification Details pop-up displays.

Step 4 Using the field descriptions, below, as a guide, complete the fields as needed.

Step 5 Click the **Save** button, or right-click in the Details and then click **Save**, to save the record and return to the Actions List.

Step 6 If appropriate, repeat these steps for any additional Email Notifications you want to create.

SNMP Notification Details Field Descriptions

The table below describes the fields and buttons that display in the SNMP Notification Details.

Field Name	Description
Action Criteria	This section contains criteria for performing the action.
Action Inheritance	For Workflow tasks only; the records this action applies to. Options: <ul style="list-style-type: none"> • SELF - This action applies only to the workflow; it is not inherited by its children tasks. • SELF/CHILDREN - This action applies to the workflow and its contained tasks (children). • CHILDREN - This action applies only to the tasks within the workflow (children).
Status	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.
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	Generates the action or notification if the task started late, based on the Late Start Time specified in the task.
On Late Finish	Generates the action or notification if the task finishes late, based on the Late Finish time specified in the task.
On Early Finish	Generates the action or notification if the task finishes early, based on the Early Finish Time specified in the task.
Description	Description of this SNMP notification.

Action Details	This section contains additional details about the action.
SNMP Manager	The SNMP Manager that will receive the SNMP notification.
Notification Severity	Severity of this notification. Options: <ul style="list-style-type: none"> • Normal (1) • Warning (2) • Minor (3) • Major (4) • Critical (5)
Buttons	This section identifies the buttons displayed above and below the Action Details that let you perform various actions.
Save	Saves a new Action record in the Controller database.
Update	Saves updates to the record.
Delete	Deletes the current record.
Refresh	Refreshes any dynamic data displayed in the Details.
Close	Closes the Details pop-up of this action.

System Operation Actions

- [Overview](#)
- [Creating a System Operation](#)
- [System Operation Details Field Descriptions](#)

Overview

A System Operation allows you to run an Opwise Controller 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 create one or more System Operations for any Controller task. For Workflow tasks, you can also specify whether you want a System Operation action to apply to the workflow itself, the workflow and/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.

Creating a System Operation

Step 1 Display the Task Details of the task for which you are creating the System Operation.

Step 2 Click the **Actions** tab. A list of any defined Actions for that task displays. For example, the following Actions list shows that two Abort Actions have been created for this task.

Status	Description	Type Details	Exit Codes	On Late Start	On Late Finish	On Early Finish	Cancel Process if Active	Override Exit Code	Updated By	Updated
FAILED			20	No	No	No	No		ops.admin	2014-06-30 10:02
CANCELLED			12	No	No	No	No		ops.admin	2014-06-30 10:02

Step 3 Click the **New** button that displays on the System Operations row. The System Operation Details pop-up displays.

Step 4 Using the field descriptions, below, as a guide, complete the fields as needed.


Step 5 Click the **Save** button, or right-click in the Details and then click **Save**, to save the record and return to the Actions List.

Step 6 If appropriate, repeat these steps for any additional System Operations you want to create.

System Operation Details Field Descriptions

The table below describes the fields and buttons that display in the System Operation Details.

Field Name	Description
Action Criteria	This section contains criteria for performing the action.
Type Details	Displays - on the System Operations actions list - the type of System Operation defined in this action.
Action Inheritance	For Workflow tasks only; the records this action applies to. Options: <ul style="list-style-type: none"> • SELF - This action applies only to the workflow; it is not inherited by its children tasks. • SELF/CHILDREN - This action applies to the workflow and its contained tasks (children). • CHILDREN - This action applies only to the tasks within the workflow (children).
Status	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.
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	Generates the action or notification if the task started late, based on the Late Start Time specified in the task.
On Late Finish	Generates the action or notification if the task finishes late, based on the Late Finish time specified in the task.
On Early Finish	Generates the action or notification if the task finishes early, based on the Early Finish Time specified in the task.
Description	Description of this System Operation.

Action Details	This section contains additional details about the action.
System Operation	<p>Specific system operation to perform.</p> <p>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 • Run Task Instance Command
System Notification	<p>Status of the specified system operation (see above) that will trigger a system notification.</p> <p>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 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>

Command	<p>If System Operation is Run Task Instance Command: Type of task instance command to run.</p> <p>Options:</p> <ul style="list-style-type: none"> • Cancel • Force Finish • Force Finish (Halt) • Force Finish/Cancel • Force Finish/Cancel (Halt) • Skip • Unskip • Hold • Release • Release Recursive • Clear All Dependencies • Clear Exclusive • Clear Predecessors • Clear Resources
Instance Lookup Option	<p>If System Operation is Run Task Instance Command: Specification for how to search for the task instance to run a command against.</p> <p>Options:</p> <ul style="list-style-type: none"> • Instance Name • Instance Name/Task • Instance Id • Task
Instance Name	<p>If Instance Lookup Option is Instance Name or Instance Name/Task: Required - Name of the task instance to run the command against. Variables supported.</p>
Instance Criteria	<p>If Instance Lookup Option is Instance Name, Instance Name/Task, or Task: Additional criteria for selecting a specific task instance if multiple task instances have matching names.</p> <ul style="list-style-type: none"> • Oldest Active Instance (default) • Newest Active Instance • Newest Instance (Unskip command only) • Oldest Instance (Unskip command only) <p>(An Active task instance is an instance that is not in any of these statuses: Skipped, Finished, Success.)</p>
Workflow Instance Name	<p>If Instance Lookup Option is Instance Name, Instance Name/Task, or Task: Name of the workflow in which the specified task instance is contained. Variables supported.</p>
Task Reference	<p>If Instance Lookup Option is Instance Name/Task or Task: Required - Name of the task for which the task instance was run.</p>
Task Reference Variable	<p>If Instance Lookup Option is Instance Name/Task or Task: the Task field (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>
Instance Id	<p>If Instance Lookup Option is Instance Id: ID of task instance to run the command against. The instance Id (sysid) is a 32-digit hexadecimal number. You can use the <code>ops_task_id</code> variable or <code>\$_siblingid('mytask')</code> function to get the instance Id.</p>
Buttons	<p>This section identifies the buttons displayed above and below the Action Details that let you perform various actions.</p>
Save	<p>Saves a new Action record in the Controller database.</p>
Update	<p>Saves updates to the record.</p>
Delete	<p>Deletes the current record.</p>
Refresh	<p>Refreshes any dynamic data displayed in the Details.</p>
Close	<p>Closes the Details pop-up of this action.</p>

Copying Tasks

- Overview
- Copying One or More Tasks from a Tasks List
- Copying a Task from the Task Details

Overview

You can make copies of all Opwise Controller records, including tasks, using the standard method for [Copying a Record](#): selecting **Insert** on the [Action menu](#).

However, this does not make copies of other records that may be attached to the task, such as Notifications, Actions, and Variables.

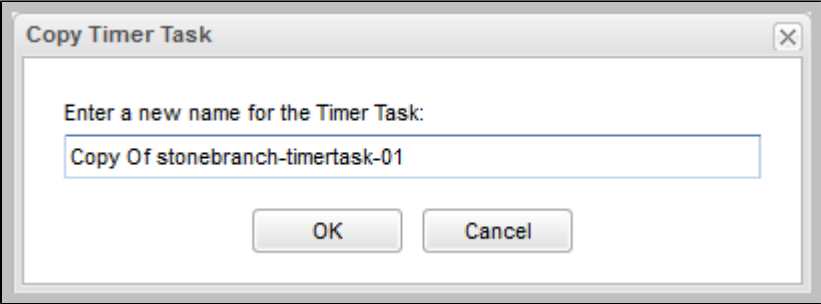
The Copy option allows you to make a complete copy of a task, including all of its associated records. It does not copy referenced records, such as virtual resources, but retains the relationship to these records for the copied task.

Copying One or More Tasks from a Tasks List

Step 1	From the Automation Center navigation pane, select Tasks > <task type> . The Tasks list for that task type displays.
Step 2	Locate the task(s) you want to copy (see Filtering).
Step 3	<p>Copy the task(s):</p> <p>Copy One Task</p> <ol style="list-style-type: none"> 1. Right-click the Task Name. 2. On the Action menu, select Copy. A Copy Task pop-up dialog displays. <div data-bbox="300 1014 1112 1316" data-label="Image"> </div> <ol style="list-style-type: none"> 3. Enter a new name for the task and click OK. <p>Copy Multiple Tasks</p> <ol style="list-style-type: none"> 1. Ctrl-Click the tasks you want to copy. 2. Right-click any of the selected tasks. 3. On the Action menu, select Copy. 4. On the Confirmation pop-up that displays, click OK. The copied tasks are added to the list, with Copy of added as a prefix to the Task Name for each task. If a task with that Copy of name already exists, a numerical suffix is added to the task name.

Copying a Task from the Task Details

Step 1 Select a task from a Tasks list. The Task Details for that task displays.

Step 2	<p>Either:</p> <ul style="list-style-type: none">• Click the Copy button.• Right-click the Details to display the Action menu, and then select Copy. <p>A Copy Task pop-up dialog displays.</p> <div data-bbox="224 310 1040 611" data-label="Image"></div>
Step 3	<p>Enter a new name for the task and click OK.</p>

Setting Mutually Exclusive Tasks

Setting Mutually Exclusive Tasks

You can set a task to be mutually exclusive with one or more other tasks. Opwise Controller 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 and, in the Task Details for that task, click the **Mutually Exclusive Tasks** tab. The Mutually Exclusive Tasks list displays a list of any tasks that have been set to be mutually exclusive with this task:

Type	Exclusive Task	Updated By	Updated
Direct	stonebranch-taskmonitor-01	stonebranch-user-02	2014-06-30 10:54:41 -0400
Direct	stonebranch-taskmonitor-02	stonebranch-user-01	2014-06-30 10:55:13 -0400

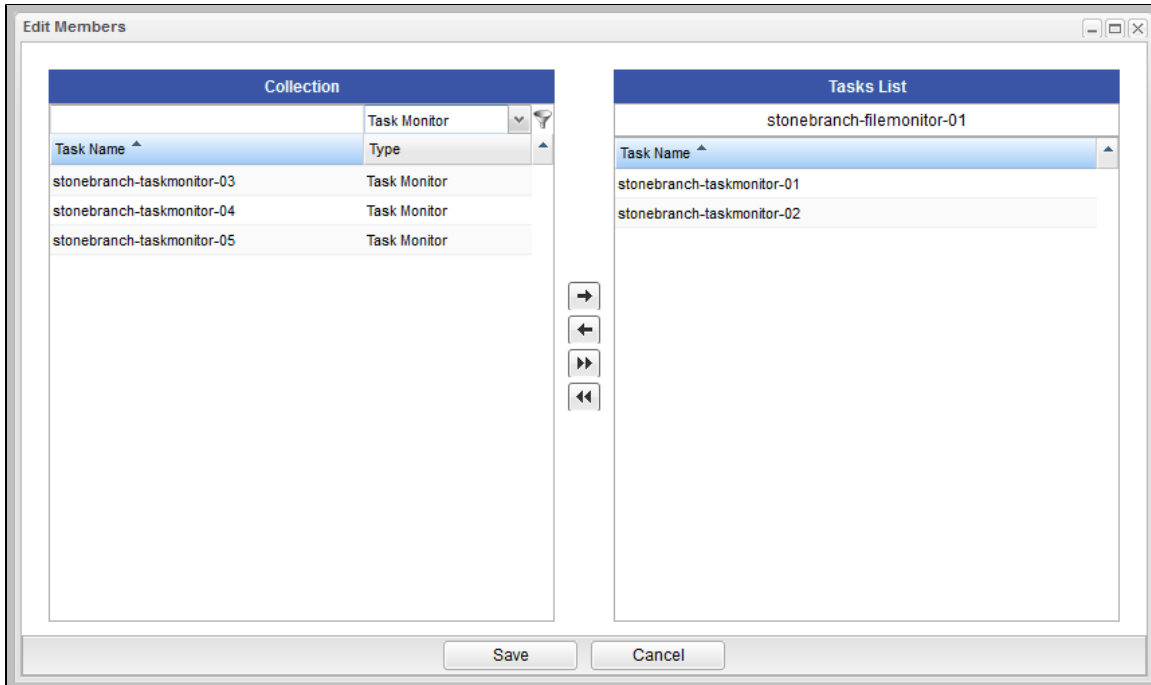
For any tasks on the list, the **Type** field indicates which task the mutually exclusive dependency was added to:

- **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 the **Edit** button to display the Edit Members dialog.



- The **Collection** window displays all Controller tasks. To display only a specific type of task, select that task type from the drop-down field at the top of the **Collection** window.
- The **Tasks List** window displays all tasks that are to be run mutually exclusive with this task.

Step 3 If you want to filter the tasks in the Collection window, click the filter icon (see [Filtering](#) for information about how to construct a filter.)

Step 4 Move tasks from the **Collection** window to the **Tasks List** window:

1. To move a single task, double-click it or click it once and then click the > arrow.
2. To move multiple tasks, Ctrl-click them and then click the > arrow.
3. To move all tasks, click the >> arrow.

To move tasks from the **Tasks List** window to the **Collection** window:

1. To move a single task, double-click it or click it once and then click the < arrow.
2. To move multiple tasks, Ctrl-click them and then click the < arrow.
3. To move all tasks, click the << arrow.

Step 5 Click **Save**. All of the tasks in the **Tasks List** window will list this task (as Type **Indirect**) on their Mutually Exclusive Tasks list.

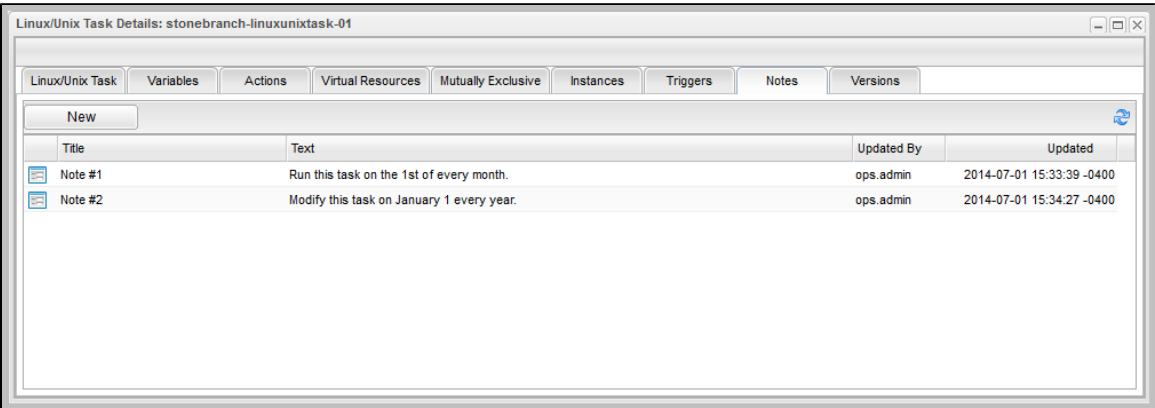
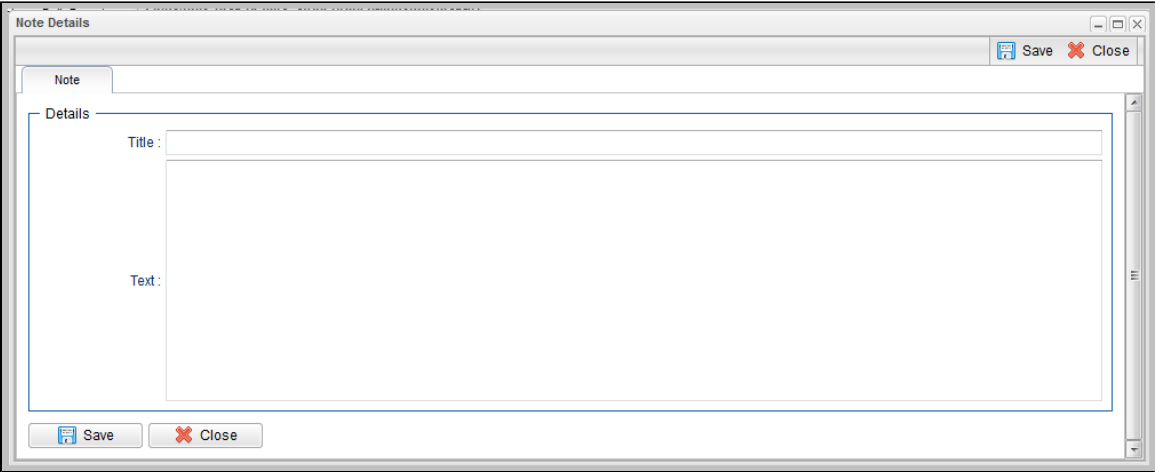
Creating Notes

- Introduction
- Adding a Note
- Note Details Field Descriptions
- Deleting a Note

Introduction

You can create a note for any Opwise Controller [task](#) or [script](#). The note can consist of information needed by operations personnel or other instructions or tips.

Adding a Note

Step 1	Open the task or script to which you want to attach a note.
Step 2	Click the Notes tab. The Notes list displays a list of notes (if any) that have been created for this task or script.
	
Step 3	Click the New button. The Note Details displays.
	
Step 4	Enter a Title and Text for the note.
Step 5	Click the Save button.

Note Details Field Descriptions

Field Name	Description
Details	This section contains detailed information about the note.
Title	Title of this note. Displays in the Title column on the Notes list.
Text	Text of the note.
Buttons	This section identifies the buttons displayed above and below the Note Details that let you perform various actions.
Save	Saves a new task record in the Controller database.
Update	Saves updates to the record.
Refresh	Refreshes any dynamic data displayed in the Details.
Delete	Deletes the current record.
Close	For pop-up view only; closes the pop-up view of this task.

Deleting a Note

To delete a note, either:

- Right-click the note in the Notes list and then click **Delete**.
- [Open](#) the note you want to delete and click the **Delete** button.

Manually Running and Controlling Tasks

- Overview
- Issuing Commands Against Task Instances
 - Issuing Commands from the Activity Monitor
 - Issuing Commands from the Task Instances List
 - Issuing Commands from the Workflow Monitor
- Commands Supported for Task Instance Statuses
 - Agent-Based Task Types
- Manually Launching a Task
 - Launch One or More Tasks from a Tasks List
 - Launch a Task from Task Details
 - 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 Monitor or Task Instances List
 - Set Priority on a Task Instance from the Workflow Monitor
- Re-running a Task Instance
 - Re-run a Task Instance from the Activity Monitor or Task Instances List
 - Re-run a Task Instance from the Workflow Monitor
- Cancelling a Task Instance
 - Cancel a Task Instance from the Activity Monitor or Task Instances List
 - Cancel a Task Instance from the Workflow Monitor
- Force Finishing a Task Instance
 - Force Finish a Task Instance from the Activity Monitor or Task Instances List
 - Force Finish a Task Instance from the Workflow Monitor
- Force Finishing (Halt) a Task Instance
 - Force Finish (Halt) a Task Instance from the Activity Monitor or Task Instances List
 - Force Finish (Halt) a Task Instance from the Workflow Monitor
- Force Finish/Cancelling a Task Instance
 - Force Finish/Cancel a Task Instance from the Activity Monitor
 - Force Finish/Cancel a Task Instance from the Workflow Monitor
- Force Finish/Cancelling (Halt) a Task Instance
 - Force Finish/Cancel (Halt) a Task Instance from the Activity Monitor
 - Force Finish/Cancel (Halt) a Task Instance from the Workflow Monitor
- Putting a Task Instance on Hold
 - Hold a Task Instance from the Activity Monitor or Task Instances List
 - Hold a Task Instance from the Workflow Monitor
- Releasing a Task Instance from Hold
 - Release a Held Task Instance from the Activity Monitor or Task Instances List
 - Release a Held Task Instance from the Workflow Monitor
- Skipping a Task Instance
 - Skip a Task Instance from the Activity Monitor or Task Instances List
 - Skip a Task Instance from the Workflow Monitor
- Showing or Hiding Skipped Task Instances
 - Setting Show / Hide Skipped Tasks from the Workflow Task Details
 - Setting Show / Hide Skipped Tasks from the Workflow Task Instance Details
 - Setting Show / Hide Skipped Tasks from the Workflow Monitor
- Unskipping a Task Instance
 - Unskip a Task Instance from the Activity Monitor or Task Instances List
 - Unskip a Task Instance from the Workflow Monitor
- Marking a Dependency as Satisfied
 - Marking a Dependency as Satisfied from the Workflow Monitor
- Clearing Predecessor Dependencies of a Task Instance
 - Clearing Predecessor Dependencies of a Task Instance from the Workflow Monitor
- Clearing Resource Dependencies of a Task Instance
 - Clear Resource Dependencies of a Task Instance from the Activity Monitor or Task Instances List
 - Clear Resource Dependencies of a Task Instance from the Workflow Monitor
- Clearing Mutually Exclusive Dependencies of a Task Instance
 - Clear Mutually Exclusive Dependencies of a Task Instance from the Activity Monitor or Task Instances List
 - Clear Mutually Exclusive Dependencies of a Task Instance from the Workflow Monitor
- Clearing All Dependencies of a Task Instance
 - Clear All Dependencies of a Task Instance from the Activity Monitor or Task Instances List
 - Clear All Dependencies of a Task Instance from the Workflow Monitor

Overview

A number of commands are available on the [Activity Monitor](#) and the [Task Instances list](#) 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](#).

Issuing Commands Against Task Instances

You can issue commands against task instances from:

- [Activity Monitor](#)
- [Task Instances list](#) (and the [Task Instances list for a specific task](#))
- [Workflow Monitor](#).

See [Commands Supported for Task Instance Statuses](#), below, for a list of task instances (and their statuses) for which these commands can be issued.

See [Task Instance Status Types](#) for a description of each type of task instance status.

Command	Description
Cancel	Cancels a running task instance (see Cancelling a Task Instance).
Clear All Dependencies	Workflow tasks only: Clears all dependencies (predecessors, resources, and exclusive) of a task instance (see Clearing All Dependencies of a Task Instance).
Clear Exclusive	Clears mutually exclusive dependencies of a task instance (see Clearing Mutually Exclusive Dependencies of a Task Instance).
Clear Predecessors	Workflow tasks only: Clears predecessor dependencies of a task instance (see Clearing Predecessor Dependencies of a Task Instance).
Clear Resources	Clears resource dependencies of a task instance (see Clearing Resource Dependencies of a Task Instance).
Force Finish	Places a task instance into the Finished status (see Force Finishing a Task Instance).
Force Finish (Halt)	Places a task instance into the Finished status (see Force Finishing (Halt) a Task Instance).
Force Finish/Cancel	Cancels a task and places it into the Finished status (see Force Finish/Cancelling a Task Instance).
Force Finish/Cancel (Halt)	Cancels a task and places it into the Finished status (see Force Finish/Cancelling (Halt) 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 instances 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 Priority	Sets the priority of a task instance in Started, Running, or Queued status to High, Medium, or Low (see Changing the Priority of a Task Instance).
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 of a task instance (see Unskipping a Task Instance).

Issuing Commands from the Activity Monitor

646 Task Instances							
Instance Name	Type	Status	Invoked By	Start Time	End Time	Updated	Filter...
lecu-wkfl-sleep	Workflow	Success	Manually Launched	2014-09-02 12:57:55 -0400	2014-09-02 12:58:36 -0400	2014-09-02 12:58:36 -0400	
sleep 10	Timer	Success	Workflow: lecu-wkfl-sleep	2014-09-02 12:58:26 -0400	2014-09-02 12:58:36 -0400	2014-09-02 12:58:36 -0400	
Sleep 30	Timer	Success	Workflow: lecu-wkfl-sleep	2014-09-02 12:57:56 -0400	2014-09-02 12:58:26 -0400	2014-09-02 12:58:26 -0400	
Sleep 0	Timer	Success	Workflow: lecu-wkfl-sleep	2014-09-02 12:57:56 -0400	2014-09-02 12:57:56 -0400	2014-09-02 12:57:56 -0400	
Sleep 60	Timer	Skipped	Workflow: lecu-wkfl-sleep		2014-09-02 12:57:55 -0400	2014-09-02 12:57:55 -0400	
Sleep 60	Timer	Skipped	Workflow: lecu-wkfl-sleep		2014-09-02 12:57:55 -0400	2014-09-02 12:57:55 -0400	
Sleep 60	Timer	Skipped	Workflow: lecu-wkfl-sleep		2014-09-02 12:57:55 -0400	2014-09-02 12:57:55 -0400	
Sleep 30	Timer	Skipped	Workflow: lecu-wkfl-sleep		2014-09-02 12:57:55 -0400	2014-09-02 12:57:55 -0400	
Sleep 30	Timer	Skipped	Workflow: lecu-wkfl-sleep		2014-09-02 12:57:55 -0400	2014-09-02 12:57:55 -0400	
win-exit-code	Windows	Failed	Manually Launched	2014-09-02 12:44:03 -0400	2014-09-02 12:44:03 -0400	2014-09-02 12:44:03 -0400	
zos-workflow-regression-test	Workflow	Success	Manually Launched	2014-09-02 11:52:47 -0400	2014-09-02 12:10:30 -0400	2014-09-02 12:10:30 -0400	
zos-workflow-simple-load-test-01	Workflow	Success	Workflow: zos-workflow-regression-test	2014-09-02 11:56:39 -0400	2014-09-02 12:10:30 -0400	2014-09-02 12:10:30 -0400	
zos-task-load-simple-01	z/OS	Success	Workflow: zos-workflow-simple-load-test-01	2014-09-02 12:10:29 -0400	2014-09-02 12:10:29 -0400	2014-09-02 12:10:30 -0400	
zos-task-load-simple-01	z/OS	Success	Workflow: zos-workflow-simple-load-test-01	2014-09-02 12:10:25 -0400	2014-09-02 12:10:26 -0400	2014-09-02 12:10:27 -0400	
zos-task-load-simple-01	z/OS	Success	Workflow: zos-workflow-simple-load-test-01	2014-09-02 12:10:23 -0400	2014-09-02 12:10:23 -0400	2014-09-02 12:10:24 -0400	
zos-task-load-simple-01	z/OS	Success	Workflow: zos-workflow-simple-load-test-01	2014-09-02 12:10:20 -0400	2014-09-02 12:10:20 -0400	2014-09-02 12:10:21 -0400	
zos-workflow-simple-load-test-02	Workflow	Success	Workflow: zos-workflow-regression-test	2014-09-02 11:56:39 -0400	2014-09-02 12:10:19 -0400	2014-09-02 12:10:19 -0400	
zos-task-load-simple-02	z/OS	Success	Workflow: zos-workflow-simple-load-test-02	2014-09-02 12:10:17 -0400	2014-09-02 12:10:17 -0400	2014-09-02 12:10:19 -0400	
zos-task-load-simple-01	z/OS	Success	Workflow: zos-workflow-simple-load-test-01	2014-09-02 12:10:15 -0400	2014-09-02 12:10:15 -0400	2014-09-02 12:10:19 -0400	
zos-task-load-simple-02	z/OS	Success	Workflow: zos-workflow-simple-load-test-02	2014-09-02 12:10:12 -0400	2014-09-02 12:10:12 -0400	2014-09-02 12:10:15 -0400	
zos-task-load-simple-01	z/OS	Success	Workflow: zos-workflow-simple-load-test-01	2014-09-02 12:10:11 -0400	2014-09-02 12:10:11 -0400	2014-09-02 12:10:14 -0400	
zos-workflow-simple-load-test-03	Workflow	Success	Workflow: zos-workflow-regression-test	2014-09-02 11:56:39 -0400	2014-09-02 12:10:13 -0400	2014-09-02 12:10:13 -0400	
zos-task-load-simple-03	z/OS	Success	Workflow: zos-workflow-simple-load-test-03	2014-09-02 12:10:09 -0400	2014-09-02 12:10:09 -0400	2014-09-02 12:10:13 -0400	
zos-task-load-simple-02	z/OS	Success	Workflow: zos-workflow-simple-load-test-02	2014-09-02 12:10:07 -0400	2014-09-02 12:10:08 -0400	2014-09-02 12:10:11 -0400	
zos-task-load-simple-01	z/OS	Success	Workflow: zos-workflow-simple-load-test-01	2014-09-02 12:10:06 -0400	2014-09-02 12:10:06 -0400	2014-09-02 12:10:10 -0400	

Issue a Command Against a Single Task Instance

Either:

- Right-click a task instance on the list to display an [Action menu](#) of available commands for that task instance.
- Click the Details icon of a task instance to display the Task Instance Details, and then right-click in the Details to display an [Action menu](#) of available commands for that task instance.

Issue a Command Against Multiple Task Instances

Press **Ctrl** and right-click each task instance that you want to issue a command against to display an [Action menu](#) of available commands that can be issued against all of the selected task instances.

(You also can press **Ctrl** and right-click a single task instance and then press **Shift** and right-click another task instance to select the group of task instances between the first and second task instance, inclusive.)

Issuing Commands from the Task Instances List

Instance Name	Type	Status	Invoked By	Start Time	End Time	Updated
ecu-uag-003237rc	z/OS	Success	Manually Launched	2014-08-21 21:28:09 -0400	2014-08-21 21:28:10 -0400	2014-08-21 21:28:11 -0400
zos-task-run-simple	z/OS	Success	Workflow: zos-workflow-regression-test	2014-08-21 21:29:39 -0400	2014-08-21 21:29:39 -0400	2014-08-21 21:29:41 -0400
zos-task-security-auth-01	z/OS	Success	Workflow: zos-workflow-user-authentication	2014-08-21 21:29:42 -0400	2014-08-21 21:29:42 -0400	2014-08-21 21:29:44 -0400
zos-task-security-auth-02	z/OS	Success	Workflow: zos-workflow-user-authentication	2014-08-21 21:29:42 -0400	2014-08-21 21:29:43 -0400	2014-08-21 21:29:45 -0400
zos-task-security-auth-03	z/OS	Success	Workflow: zos-workflow-user-authentication	2014-08-21 21:29:43 -0400	2014-08-21 21:29:44 -0400	2014-08-21 21:29:47 -0400
Sleep 0	Timer	Success	Workflow: zos-workflow-user-authentication	2014-08-21 21:29:47 -0400	2014-08-21 21:29:47 -0400	2014-08-21 21:29:47 -0400
zos-task-security-auth-13	z/OS	Finished	Workflow: zos-workflow-user-authentication		2014-08-21 21:29:48 -0400	2014-08-21 21:29:48 -0400
zos-task-security-auth-10	z/OS	Finished	Workflow: zos-workflow-user-authentication		2014-08-21 21:29:48 -0400	2014-08-21 21:29:48 -0400
zos-task-security-auth-12	z/OS	Finished	Workflow: zos-workflow-user-authentication		2014-08-21 21:29:48 -0400	2014-08-21 21:29:48 -0400
zos-task-security-auth-11	z/OS	Finished	Workflow: zos-workflow-user-authentication		2014-08-21 21:29:48 -0400	2014-08-21 21:29:49 -0400
zos-task-failure	z/OS	Skipped	Workflow: zos-workflow-user-authentication		2014-08-21 21:29:48 -0400	2014-08-21 21:29:49 -0400
Sleep 0	Timer	Success	Workflow: zos-workflow-user-authentication	2014-08-21 21:29:49 -0400	2014-08-21 21:29:49 -0400	2014-08-21 21:29:49 -0400
zos-workflow-user-authenti...	Work...	Success	Workflow: zos-workflow-regression-test	2014-08-21 21:29:41 -0400	2014-08-21 21:29:49 -0400	2014-08-21 21:29:49 -0400
zos-step-action-setup	Timer	Success	Workflow: zos-workflow-step-actions	2014-08-21 21:29:49 -0400	2014-08-21 21:29:50 -0400	2014-08-21 21:29:50 -0400

Issue a Command Against a Single Task Instance

Either:

- Right-click a task instance to display an [Action menu](#) of available commands for that task instance.
- Click a task instance to display the Task Instance Details below the list, or click the Details icon to display a Details pop-up for the task instance, and then right-click in the Details to display an [Action menu](#) of available commands for that task instance.

Issue a Command Against Multiple Task Instances

Press **Ctrl** and right-click each task instance that you want to issue a command against to display an [Action menu](#) of available commands that can be issued against all of the selected task instances.

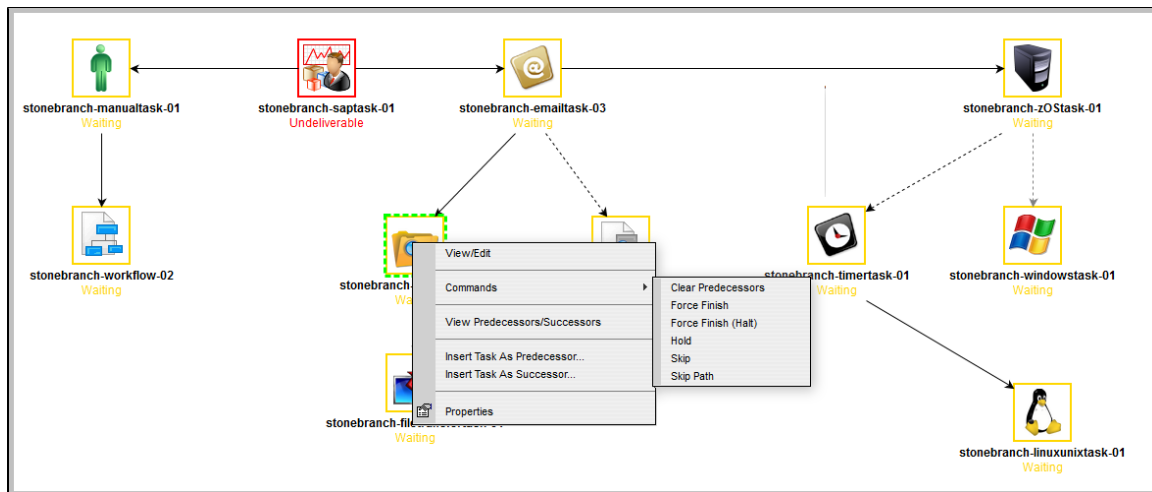
(You also can press **Ctrl** and right-click a single task instance and then press **Shift** and right-click another task instance to select the group of task instances between the first and second task instance, inclusive.)

Issuing Commands from the Workflow Monitor

From the Workflow Monitor, you can issue a command against a single task instance.

Step 1	Right-click the task instance to display a pop-up menu of tasks appropriate for the selected task instance.
Step 2	Click Commands and then click the command that you want to issue against the selected task instance.

For example:



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 (60)	Manual	<ul style="list-style-type: none"> Cancel Force Finish Force Finish (Halt) Force Finish/Cancel Force Finish/Cancel (Halt) Set Started Set Completed
Cancel Pending (99)	Agent-based	<ul style="list-style-type: none"> Force Finish Force Finish (Halt)
Cancelled (130)	All	<ul style="list-style-type: none"> Force Finish Force Finish (Halt) Re-run - Not applicable for Workflow tasks.
Confirmation Required (125)	z/OS	<ul style="list-style-type: none"> Force Finish Force Finish (Halt) Re-run - Not applicable for Workflow tasks.
Defined (0)	All	<ul style="list-style-type: none"> Clear All Dependencies Clear Predecessors Force Finish Force Finish (Halt) Hold Skip Skip Path Release Recursive - Workflow tasks only.

Exclusive Requested (22)	All	<ul style="list-style-type: none"> • Clear All Dependencies • Force Finish • Force Finish (Halt) • Hold • Skip • Skip Path
Exclusive Wait (23)	All	<ul style="list-style-type: none"> • Clear All Dependencies • Clear Exclusive • Force Finish • Force Finish (Halt) • Hold • Skip • Skip Path • Release Recursive - Workflow tasks only.
Execution Wait (33)	Agent-based	<ul style="list-style-type: none"> • Force Finish • Force Finish (Halt) • Hold • Skip • Skip Path
Failed (140)	All*	<ul style="list-style-type: none"> • Force Finish • Force Finish (Halt) • Re-run - Not applicable for Workflow tasks.
Finished (190)	All	<ul style="list-style-type: none"> • Re-run - Not applicable for Workflow tasks.
Held (20)	All	<ul style="list-style-type: none"> • Clear All Dependencies • Clear Predecessors • Force Finish • Force Finish (Halt) • Release • Skip • Skip Path • Release Recursive - Workflow tasks only.
In Doubt (110)	Agent-based	<ul style="list-style-type: none"> • Force Finish • Force Finish (Halt)
Queued (40)	Agent-based	<ul style="list-style-type: none"> • Cancel • Force Finish • Force Finish (Halt) • Hold • Set Priority
Resource Requested (25)	All tasks using Virtual Resources	<ul style="list-style-type: none"> • Clear All Dependencies • Force Finish • Force Finish (Halt) • Hold • Skip • Skip Path

Resource Wait (30)	All tasks using Virtual Resources	<ul style="list-style-type: none"> • Clear All Dependencies • Force Finish • Force Finish (Halt) • Hold • Skip • Skip Path • Release Recursive - Workflow tasks only.
Running (80)	All	<ul style="list-style-type: none"> • Cancel • Force Finish • Force Finish (Halt) • Force Finish/Cancel • Force Finish/Cancel (Halt) • Release Recursive - Workflow tasks only. • Set Priority
Running Problems (81)	Workflow	<ul style="list-style-type: none"> • Cancel • Force Finish • Force Finish (Halt) • Force Finish/Cancel • Force Finish/Cancel (Halt) • Hold • Release Recursive - Workflow tasks only.
Skipped (180)	All	<ul style="list-style-type: none"> • Unskip
Start Failure (120)	All	<ul style="list-style-type: none"> • Force Finish • Force Finish (Halt) • Re-run - Not applicable for Workflow tasks.
Started (70)	Agent-based and Manual	<ul style="list-style-type: none"> • Cancel • Force Finish • Force Finish (Halt) • Force Finish/Cancel • Force Finish/Cancel (Halt) • Set Completed - Manual tasks only. • Set Priority
Submitted (43)	z/OS	<ul style="list-style-type: none"> • Force Finish • Force Finish (Halt)
Success (200)	All	<ul style="list-style-type: none"> • Re-run - Not applicable for Workflow tasks.
Undeliverable (35)	Agent-based	<ul style="list-style-type: none"> • Force Finish • Force Finish (Halt) • Hold • Skip • Skip Path

Waiting (10)	All	<ul style="list-style-type: none"> • Clear All Dependencies • Clear Predecessors • Force Finish • Force Finish (Halt) • Hold • Skip • Skip Path • Release Recursive - Workflow tasks only
--------------	-----	---

* Workflows cannot go to Failed status.

Agent-Based Task Types

The following task types are Agent-based task types:

- Linux/Unix
- Windows
- z/OS
- Indesca
- SAP
- File Transfer
- File Monitor
- FTP File Monitor
- System Monitor

Manually Launching a Task

Two methods are available for manually launching a task:

- From a tasks list
- From task Details

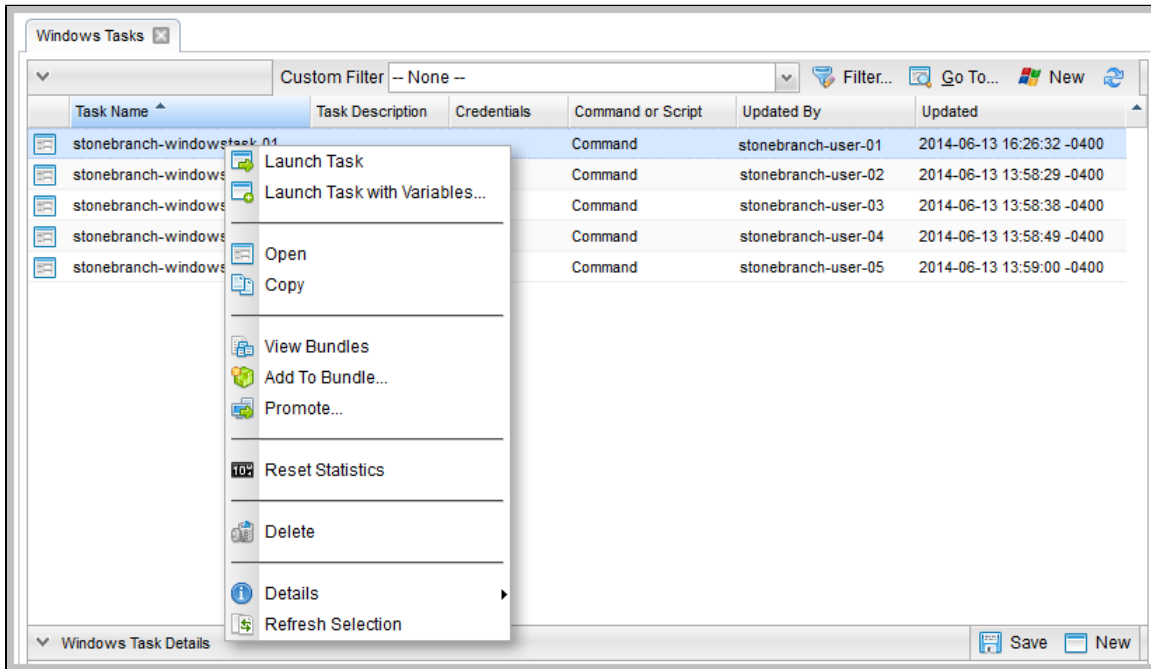
Launch One or More Tasks from a Tasks List

Step 1	From the Automation Center navigation pane, select Tasks > <type of task> . The Tasks list for that task type displays.
---------------	--

Step 2 Either:

- Right-click a single task.
- Ctrl-click multiple tasks, release the Ctrl key, and right-click any of the selected tasks.

An **Action** menu displays.

**Step 3** Select either:

- **Launch Task**
- **Launch Task with Variables**

**Note**

If you selected multiple tasks, **Launch with Variables** is disabled.

Opwise Controller creates an instance of each selected task and runs it.

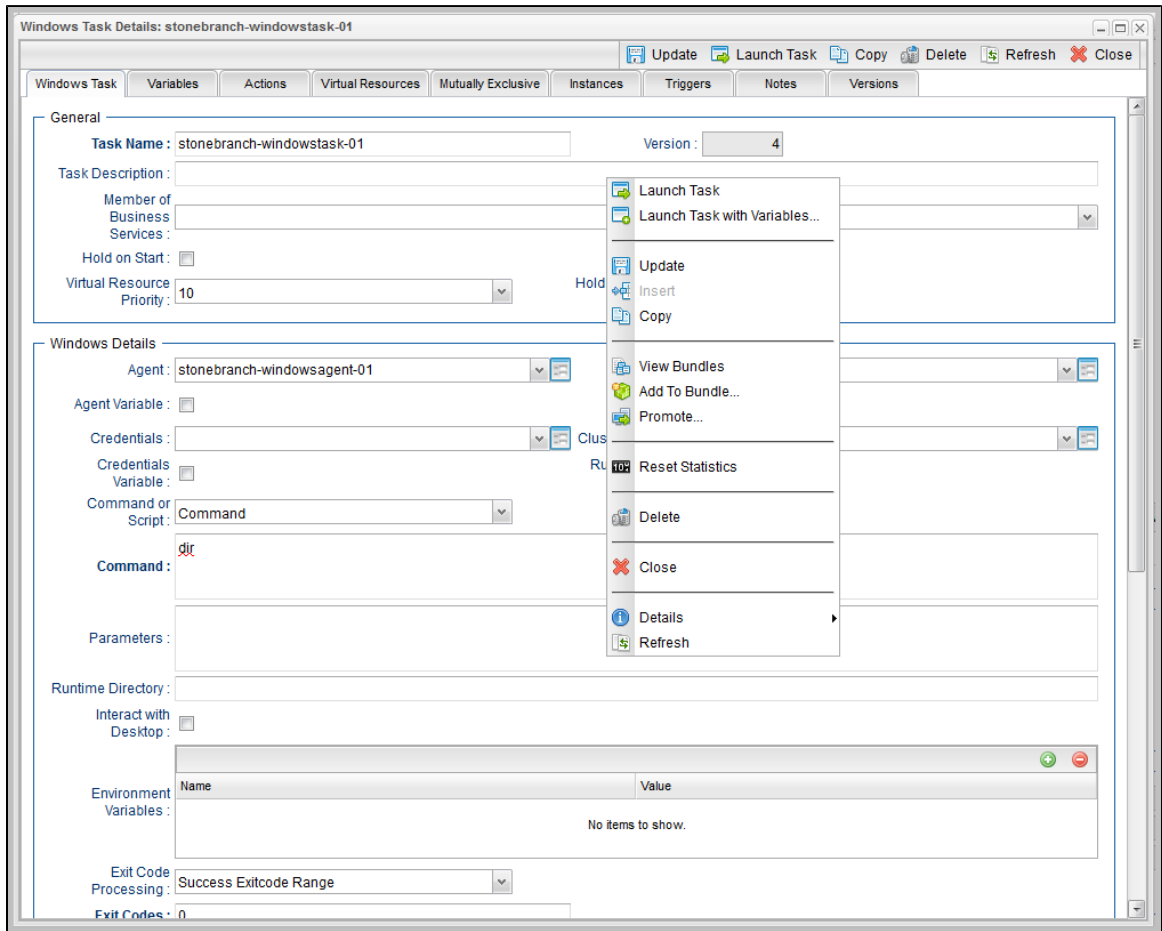
Step 4 To view Details about running task instances, select **Task Instances > Activity** from the **Automation Center** navigation pane and click on the task instance.

Launch a Task from Task Details

Step 1 Select the task you want to launch.

Step 2 Either:

- Click the **Launch Task** button.
- Right-click anywhere in the Details to display an **Action menu** and click either:
 - **Launch Task**
 - **Launch Task with Variables**



Provide Temporary Variable Values and Launch a Task Manually

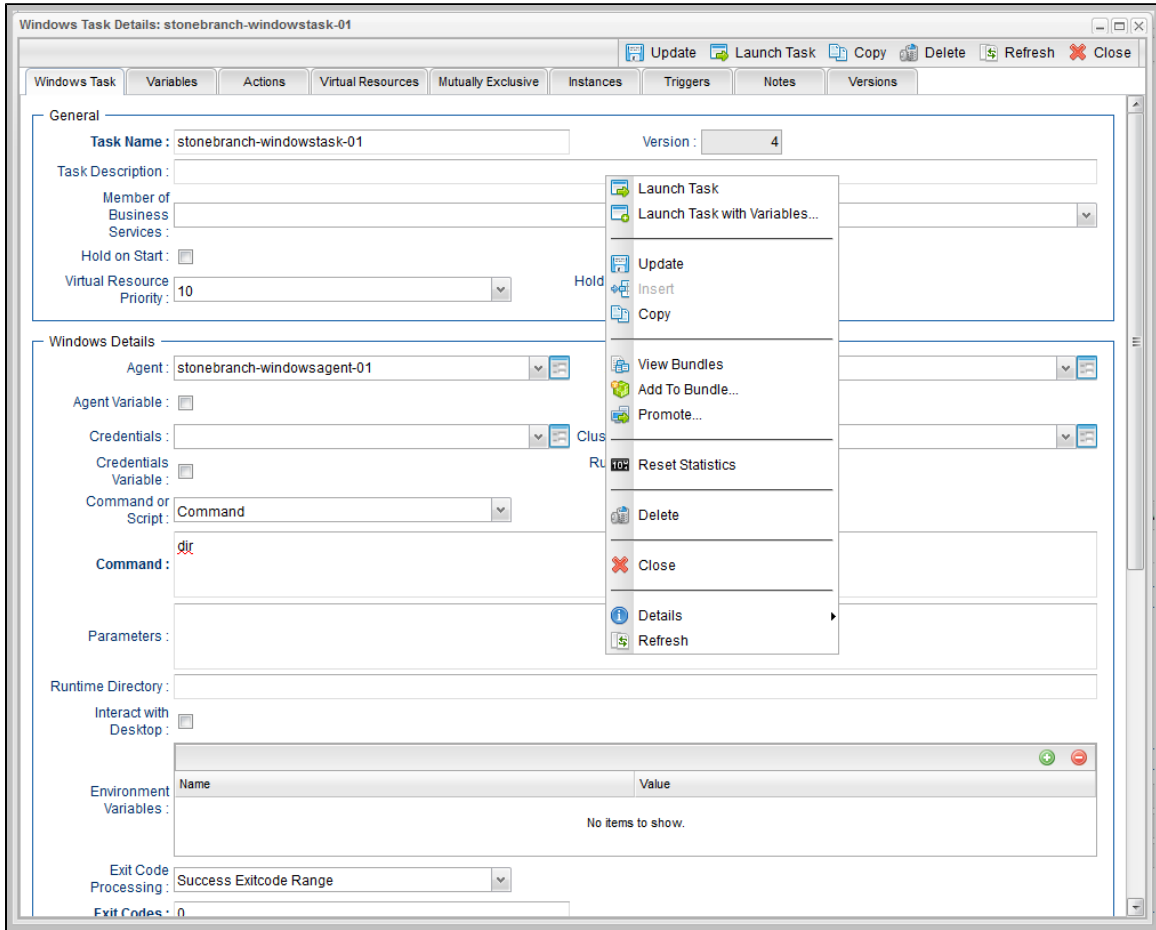
The Controller supports a Launch with Variables feature that allows you to quickly provide values for the variables specified in the task and launch it. All task types support the Launch with Variables feature.

(The following procedure assumes the task is already set up with variables where required.)

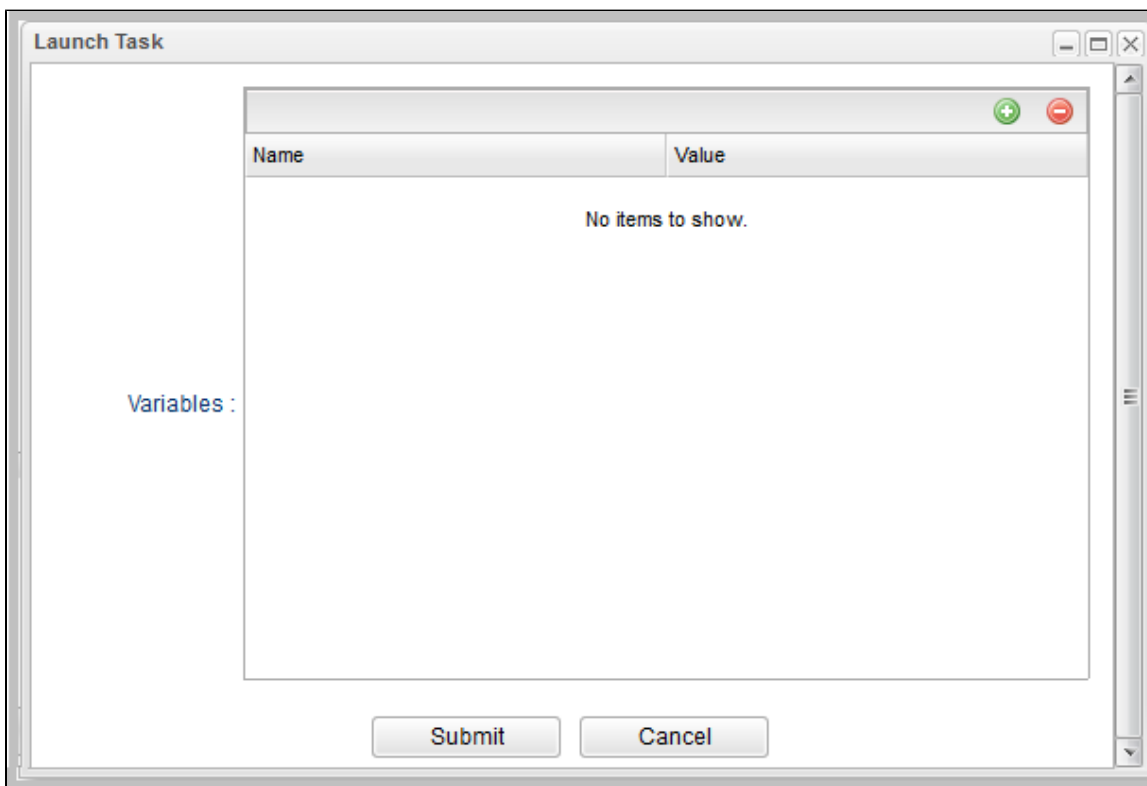
To launch a task using Launch with Variables:

Step 1 Display the task you want to launch.

Step 2 Access the Action menu.



Step 3 Select **Launch Task 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**. The Controller populates the variables with the values you supplied and launches the task.

Changing the Priority of a Task Instance

You can change the priority of a task instance so that it 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 Controller instance.

You can change the priority of a Windows, Linux/Unix, or z/OS task instance while it is in any of the following statuses: Started, Running, Queued.

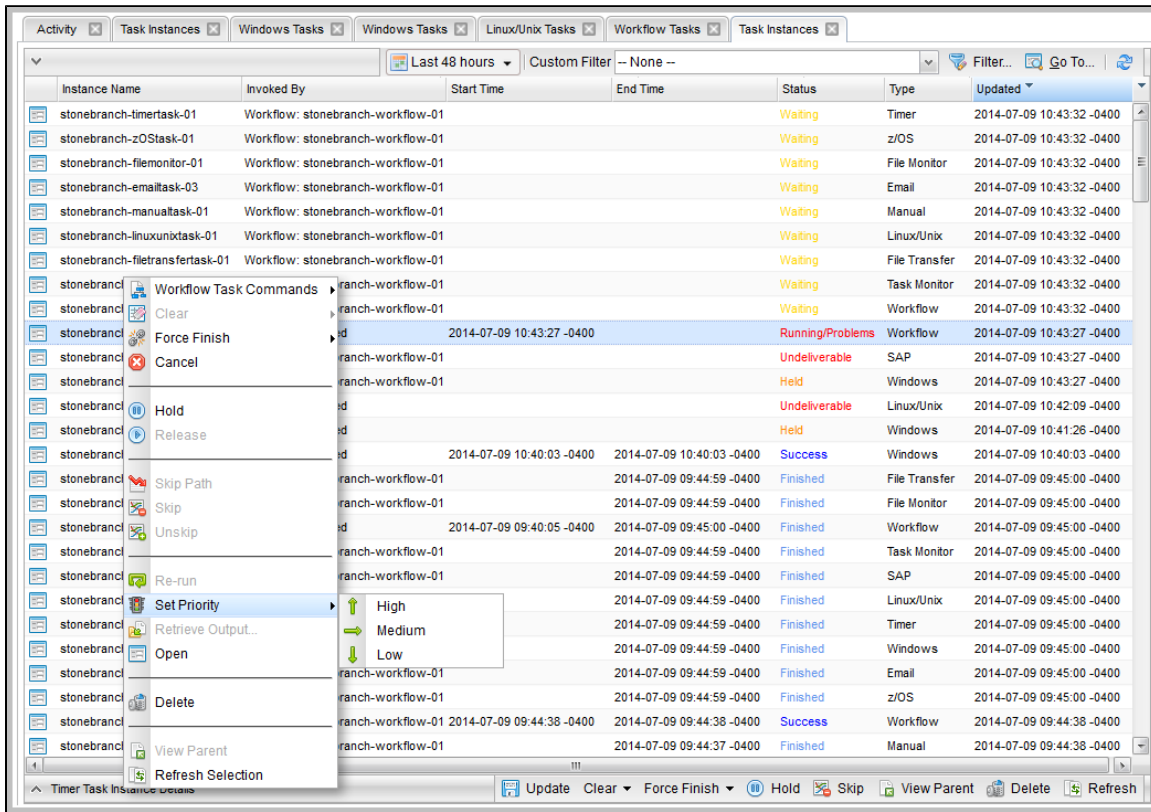
Two methods are available for changing the priority of a task instance:

- From the Activity Monitor or Task Instances list
- From the Workflow Monitor

Set Priority on a Task Instance from the Activity Monitor or Task Instances List

Step 1 Select the task instance for which you want to set the priority.

Step 2 Right-click the task instance and, on the Action menu, select the priority level.



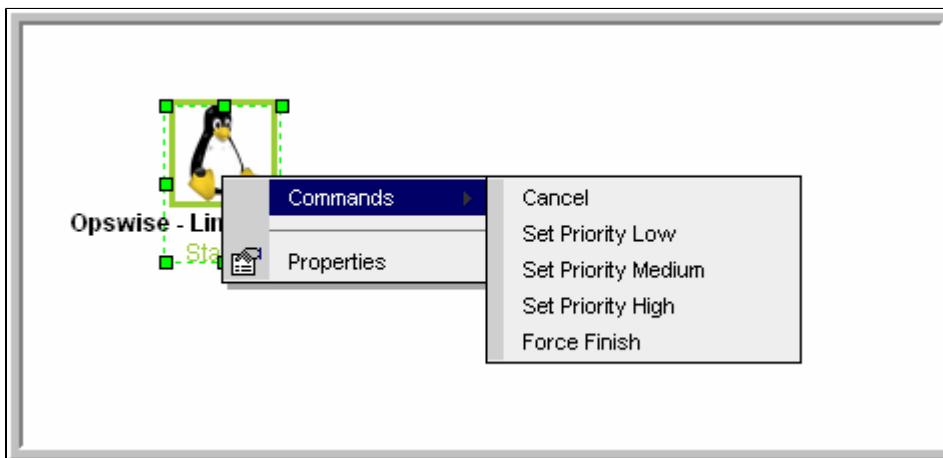
Step 3 When the Set Priority command has been executed, the Controller displays the following message at the bottom of the list:
Command Set Priority <High/Medium/Low> successfully against task instance <name>.

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 the priority.

Step 2 Select the task instance for which you want to set priority.

Step 3 Select **Commands**.



Step 4 Select a priority for the task instance.

Re-running a Task Instance

If a task instance is part of a Workflow, you can re-run it as long as the Workflow has not completed.

If a 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 Monitor.

You can re-run a task instance while it is in any of the following statuses: Success, Start Failure, Failed, Cancelled, Finished.



Note

You cannot re-run a Workflow task instance.

When you re-run a task instance, the Controller 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 [History list](#) (one for each time it ran).

Two methods are available for re-running a task instance:

- From the Activity Monitor or Task Instances list
- From the Workflow Monitor (if the task instance is running as part of a Workflow)

Re-run a Task Instance from the Activity Monitor or Task Instances List

Step 1	Select the task instance 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 Workflow Monitor

Step 1	View the Workflow that contains the task instance you want to re-run.
Step 2	Select the task instance 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.

You can cancel a task instance while it is in any of the following statuses: Queued, Action Required, Started, Running.

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.

Two methods are available for cancelling a task instance:

- From the Activity Monitor or Task Instances list
- From the Workflow Monitor (if the task instance is running as part of a Workflow)

Cancel a Task Instance from the Activity Monitor or Task Instances List

Step 1	Select the task instance 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.
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.

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.

One purpose of Force Finish is to allow successor 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, the Controller 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 successor 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.

Two methods are available for Force Finishing a task instance:

- From the Activity Monitor or Task Instances list
- From the Workflow Monitor (if the task instance is running as part of a Workflow)

Force Finish a Task Instance from the Activity Monitor or Task Instances List

Step 1	Select the task instance 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.
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 Finishing (Halt) a Task Instance

Just as with the [Force Finish](#) command, the Force Finish (Halt) command puts a task instance into the Finished status, regardless of what the task instance is doing.

However, Force Finish (Halt) prevents successor task instances in a Workflow from being run. Those tasks will not run until you re-run the task against which you had executed Force Finish (Halt).

You can Force Finish (Halt) 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.

If a task instance is running when the user issues a Force Finish (Halt), the Controller 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 (Halt) command. All successor task instances waiting for successful completion of this task instance will remain in Waiting status.

Similarly, task monitors are not released if a Force Finish (Halt) is executed against a task being monitored.

**Note**

There are two areas in the user interface that you can check to determine if a task instance was forced finished with halt:

- [Additional Information](#) field in the [Audit Details](#) for that force finished task instance.
- [Status Description](#) field in the [Task Instance Details](#) for that force finished task instance.

Two methods are available for Force Finishing (Halt) a task instance:

- From the Activity Monitor
- From the Workflow Monitor (if the task instance is running as part of a Workflow)

Force Finish (Halt) a Task Instance from the Activity Monitor or Task Instances List

Step 1	Select the task instance you want to Force Finish (Halt).
Step 2	Click Force Finish (Halt) . The task status changes to Finished.

Force Finish (Halt) a Task Instance from the Workflow Monitor

Step 1	View the workflow that contains the task instance you want to Force Finish (Halt).
Step 2	Select the task instance.
Step 3	Select Commands .
Step 4	Select Force Finish (Halt) . The task status changes to Finished and the Workflow Console opens to display information about the Force Finish (Halt).

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.

You can Force Finish/Cancel a task instance while it is in any of the following statuses: Queued, Action Required, Started, Running.

One purpose of Force Finish/Cancel is to cancel a task instance and allow successor 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 Timer 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 the Controller marks the task instance as Finished; processing does not continue. Assuming they have no other dependencies, all successor 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.

Two methods are available for Force Finish/Canceling a task instance:

- From the Activity Monitor or Task Instances list
- From the Workflow Monitor (if the task instance is running as part of a Workflow)

Force Finish/Cancel a Task Instance from the Activity Monitor

Step 1	Select the task instance 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.
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.

Force Finish/Cancelling (Halt) a Task Instance

Just as with the [Force Finish/Cancel](#) command, the Force Finish/Cancel (Halt) command cancels a task instance and puts it into Finished status, regardless of what the task instance is doing.

However, Force Finish/Cancel (Halt) prevents successor task instances in a Workflow from being run. Those tasks will not run until you re-run the task against which you had executed Force Finish/Cancel (Halt).

Task monitors are not released if a Force Finish/Cancel (Halt) is executed against a task being monitored.

You can Force Finish/Cancel (Halt) a task instance while it is in any of the following statuses: Queued, Action Required, Started, Running.



Note

The Force Finish/Cancel (Halt) command is not implemented for Timer 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 (Halt) 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 (Halt) command, the Agent cancels the task instance, if possible, and then the Controller marks the task instance as Finished; processing does not continue. All successor task instances waiting for successful completion of this task instance remain in Waiting status.
- 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.



Note

There are two areas in the user interface that you can check to determine if a task instance was forced finish / cancelled with halt:

- [Additional Information](#) field in the [Audit Details](#) for that force finish / cancelled task instance.
- [Status Description](#) field in the [Task Instance Details](#) for that force finish / cancelled task instance.

Two methods are available for Force Finish/Cancelling (Halt) a task instance:

- From the Activity Monitor
- From the Workflow Monitor (if the task instance is running as part of a Workflow)

Force Finish/Cancel (Halt) a Task Instance from the Activity Monitor

Step 1	Select the task instance you want to Force Finish/Cancel (Halt).
Step 2	Click Force Finish/Cancel (Halt) . The task status changes to Finished.

Force Finish/Cancel (Halt) a Task Instance from the Workflow Monitor

Step 1	View the workflow that contains the task instance you want to Force Finish/Cancel (Halt).
Step 2	Select the task instance.
Step 3	Select Commands .
Step 4	Select Force Finish/Cancel (Halt) . The task status changes to Finished and the Console opens to display information about the Force Finish/Cancel (Halt).

Putting a Task Instance on Hold

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.

You can put a task instance on hold while it is in any of the following statuses: Defined, Waiting, Resource Wait, Queued.

Two methods are available for putting a task instance on hold:

- From the Activity Monitor or Task Instances list
- From the Workflow Monitor (if the task instance is running as part of a Workflow)

Hold a Task Instance from the Activity Monitor or Task Instances List

Step 1	Select the task instance 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.
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

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.

You can release a non-Workflow task instance from hold from the Activity Monitor or Task Instances list while it is in the following status: Held.

To release the Workflow, use one of the following commands:

- To release the entire held Workflow and its task instances, use **Release Recursive**.
- To release a Workflow that is not in Held status but has task instances that are in Held status, use **Release Recursive**. In this case, you can issue **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.

Two methods are available for releasing a task instance from hold:

- From the Activity Monitor or Task Instances list
- From the Workflow Monitor (if the task instance is running as part of a Workflow)

Release a Held Task Instance from the Activity Monitor or Task Instances List

Step 1	Select the task instance 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.
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 also can skip a task instance or a task instance path so that the task instance and all of its dependent task instances automatically are skipped as well.

You can skip any task instance as long as it has not yet started running; that is, while it is in any of the following statuses: Defined, Waiting, Held, Resource Requested, Resource Wait.

Two methods are available for skipping a task instance:

- From the Activity Monitor or Task Instances list
- From the Workflow Monitor (if the task instance is running as part of a Workflow)



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 Details (using the trigger's Skip Count field) so that the Controller skips the next N number of trigger occurrences for launching the task.
2. Modifying a Workflow Details 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 Details 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 Monitor or Task Instances List

Step 1	Select the task instance 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.
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 on the Workflow Monitor either:

- Before the Workflow is running
- While the Workflow is running

Three methods are available for selecting whether or not to show or hide skipped task instances:

- From the Workflow Task Details
- From the Workflow Task Instance Details
- From the Workflow Monitor

Setting Show / Hide Skipped Tasks from the Workflow Task Details

Step 1	Display the Workflow Task Details for the Workflow that you want to show/hide Skipped task instances.
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 Show / Hide Skipped Tasks from the Workflow Task Instance Details

Step 1	Display the Workflow Task Instance Details for the Workflow task instance that you want to show/hide Skipped task instances.
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.

Setting Show / Hide Skipped Tasks from the Workflow Monitor

Open the Workflow task instance in the Workflow Monitor. By default, the Workflow Monitor will show or hide skipped task instances based on the Workflow task 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 Monitor or Task Instances list
2. From the Workflow Monitor

Unskip a Task Instance from the Activity Monitor or Task Instances List

Step 1	Select the task instance 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 . (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.

Marking a Dependency as Satisfied

For task instances running inside of a Workflow, you can mark a single predecessor dependency as satisfied to allow the task instance to run.

Marking a dependency as satisfied has the same result as [clearing a dependency](#).

You can mark a dependency as satisfied on task instances in the following status: Defined, Waiting, Held.

One method is available for marking a dependency as satisfied:

- From the Workflow Monitor

Marking a Dependency as Satisfied from the Workflow Monitor

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 Predecessor Dependencies of a Task Instance

For a task instance running inside of a Workflow, you can clear all predecessor dependencies to allow that task instance to run. Clearing a predecessor dependency has the same result as [satisfying a predecessor dependency](#).

You can clear predecessor dependencies of a task instance while it is in any of the following statuses: 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. To clear all dependencies, see [Clearing All Dependencies from Tasks](#), below.

One method is available for clearing predecessor dependencies of a task instance:

- From the Workflow Monitor

Clearing Predecessor Dependencies of a Task Instance from the Workflow Monitor

Step 1	View the Workflow that contains the task instance whose predecessor dependencies you want to satisfy.
Step 2	Select the task instance for which you want to clear predecessor dependencies.
Step 3	Select Commands .
Step 4	Select Clear Predecessors . The task instance is launched normally.

Clearing Resource Dependencies of a Task Instance

For task instances for which resources have been defined, you can clear those resource dependencies.

You can clear resource dependencies of task instances while it is in the following status: Resource Wait.

Two methods are available for clearing resource dependencies from task instances:

1. From the Activity Monitor or Task Instances list
2. From the Workflow Monitor

Clear Resource Dependencies of a Task Instance from the Activity Monitor or Task Instances List

Step 1	Select the task instance 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 . (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 of a Task Instance

For task instances that are mutually exclusive with other task instances, you can clear those mutually exclusive dependencies.

Any task instances that were mutually exclusive with this task instance will no longer be mutually exclusive.

You can clear mutually exclusive dependencies of a task instance while it is in the following status: Exclusive Wait.

Two methods are available for clearing mutually exclusive dependencies from task instances:

1. From the Activity Monitor or Task Instances list
2. From the Workflow Monitor

Clear Mutually Exclusive Dependencies of a Task Instance from the Activity Monitor or Task Instances List

Step 1	Select the task instance whose mutually exclusive dependencies you want to clear.
Step 2	Click Clear Exclusive . Mutually exclusive dependencies of the task instance are cleared, and the task instance is launched normally.

Clear Mutually Exclusive Dependencies of a Task Instance from the Workflow Monitor

Step 1	View the Workflow that contains the task instance you want to clear of mutually exclusive dependencies.
Step 2	Select a task instance . (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.

Clearing All Dependencies of a Task Instance

You can clear all dependencies (predecessors, resources, and exclusive) to allow a task instance to run.

You can clear all dependencies of a task instance while it is in any the following status: Defined, Waiting, Held, Exclusive Requested, Exclusive Wait, Resource Requested, Resource Wait.

Two methods are available for clearing mutually exclusive dependencies from task instances:

1. From the Activity Monitor or Task Instances list
2. From the Workflow Monitor

Clear All Dependencies of a Task Instance from the Activity Monitor or Task Instances List

Step 1	Select the task instance whose dependencies you want to satisfy.
Step 2	Click Clear All Dependencies . The task instance is launched normally.

Clear All Dependencies of 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 for which you want to clear predecessor dependencies.
Step 3	Select Commands .
Step 4	Select Clear All Dependencies . The task instance is launched normally.

Creating and Maintaining Workflows

- Overview
- Creating a Workflow Task
 - Workflow Task Details
 - Workflow Task Details Field Descriptions
- Viewing a Workflow Task Instance
 - Workflow Task Instance Details
 - Workflow Task Instance Details Field Descriptions
- Designing a Workflow
 - Designing, Modifying, or Viewing Multiple Workflows
 - Workflow Editor Modes
 - Workflow Editor Toolbar
 - Workflow Editor Tools
 - Searching For and Adding Tasks
 - Task Actions
 - Connector Actions
 - Specifying Connections
 - Specifying Conditions on 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
- Viewing a Task Instance in a Workflow from the Activity Monitor or Task Instances List
- Inserting a Task in an Active Workflow
 - Insert Task as Predecessor
 - Insert Task as Successor
 - Insert Task with Multiple Predecessors/Successors
- Viewing Task Predecessors / Successors
 - Viewing Task Instance Predecessors / Successors from the Activity Monitor or Task Instances List
 - Viewing Task or Task Instance Predecessors / Successors from the Workflow Editor or Workflow Monitor
 - Viewing Detailed Status of Task Instance Predecessors / Successors from Workflow Monitor
- Forecasting a Workflow for a Specific Date/Time
- Modifying Tasks in a Workflow

Overview

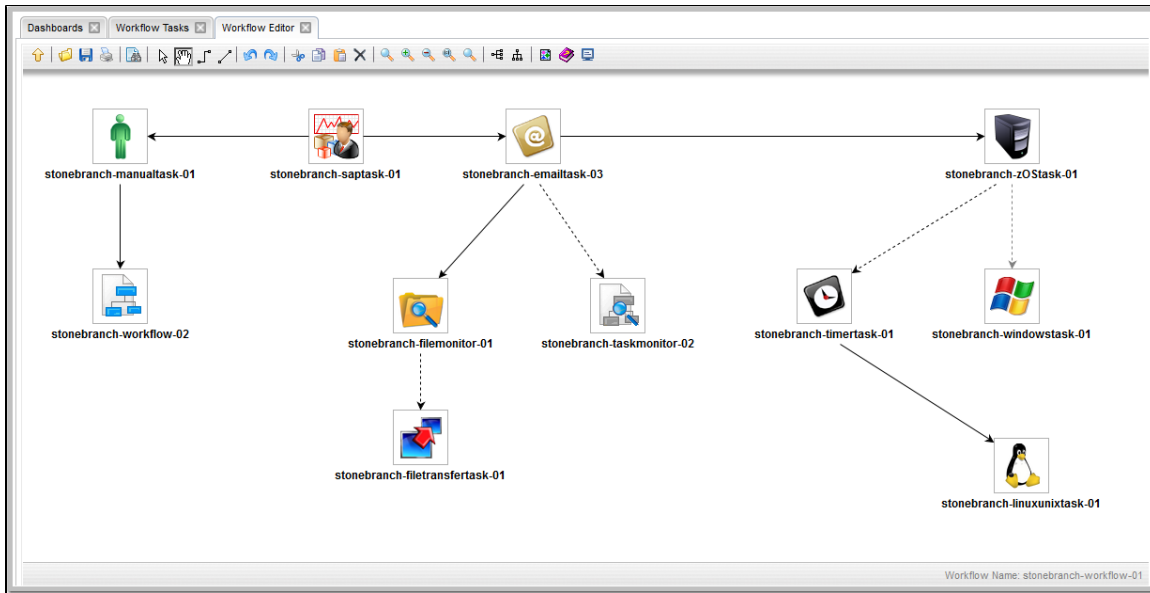
A Workflow is a task comprised of one or more tasks of one or more task types, including other Workflows.

To create a Workflow, you enter [task Details](#) just as you would for any task, and then design the Workflow on the [Workflow Editor](#), where you add, connect, and organize the tasks that will comprise the Workflow.

During and after a Workflow run, you can view detailed results of the run in the [Workflow Task Instance Details](#), and monitor the status of the Workflow on the [Workflow Monitor](#).

Although a typical Workflow is a single group of tasks, a Workflow can contain multiple groups of tasks and/or individual tasks. Each task in a group of tasks is connected to one or more other tasks in that group. Each connection is made from one task (the predecessor) to another task (the successor). There is no limit to the number of connections that can be made to or from a task.

The following sample Workflow contains a single group of tasks.



When you launch a Workflow, the first predecessor task in all task groups, and all individual tasks, begin to run. In this example, task **stonebranch-saptask-01** would begin to run.

Creating a Workflow Task

Step 1 From the [Automation Center](#) navigation pane, select **Tasks > Workflow Tasks**. The Workflow Tasks list displays a list of all currently defined Workflow tasks.

Below the list, Workflow Task Details for a new Workflow task displays. (You also can click the **New** button to display Workflow Task Details for a new Workflow task.)

The screenshot shows the 'Workflow Tasks' interface. At the top, there is a list of 5 workflow tasks:

Task Name	Task Description	Default Calendar	Updated By	Updated
stonebranch-workflow-01			ops.admin	2014-06-24 15:07:33 -0400
stonebranch-workflow-02			ops.admin	2014-06-23 16:10:24 -0400
stonebranch-workflow-03			ops.admin	2014-06-23 16:10:29 -0400
stonebranch-workflow-04			ops.admin	2014-06-23 16:10:34 -0400
stonebranch-workflow-05			ops.admin	2014-06-23 16:10:39 -0400

Below the list is the 'Workflow Task Details' form. The 'General' tab is active, showing the following fields:

- Task Name**: [Text Field]
- Version**: [Number Field] (value: 1)
- Task Description**: [Text Field]
- Member of Business Services**: [Dropdown Menu]
- Hold on Start**:
- Virtual Resource Priority**: [Number Field] (value: 10)
- Hold Resources on Failure**:

The 'Workflow Details' section includes:

- Show/Hide Skipped Tasks**: [Dropdown Menu] (value: Show Skipped)
- Default Calendar**: [Dropdown Menu]

The 'Time Options' section includes:

- Late Start**:

Step 2 Enter/select Details for a new Workflow task, using the [field descriptions](#) below as a guide.

- Required fields display in **boldface**.
- Default values for fields, if available, display automatically.

To display more of the Details fields on the screen, you can temporarily [hide the list](#).



Note

If you view [Workflow Task Details](#) for an existing Workflow task by clicking a Workflow in the list, and then want to create a new Workflow task, you must click the **New** button that displays above and below the Details.

Step 3 Click the **Save** button. The Workflow task is added to the database, and all buttons and tabs in Workflow Task Details are enabled.

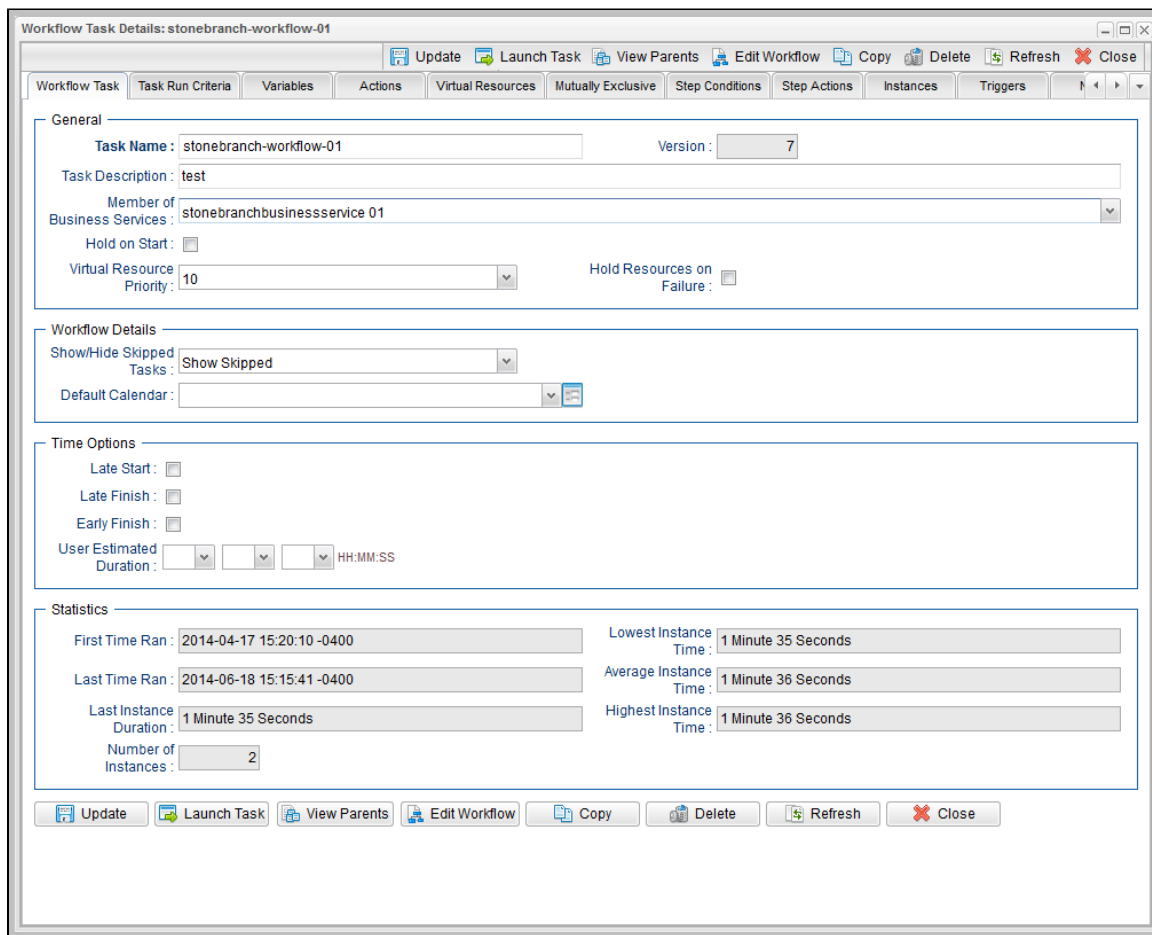
Step 4 Click the **Edit Workflow** button to display a blank Workflow Editor canvas.

Step 5 Follow the instructions provided below for [designing the Workflow](#) in the Workflow Editor.

Workflow Task Details

The following Workflow Task Details is for an existing Workflow task.


Depending on the values that you enter / select for these fields, and whether or not the Workflow task has ever been launched, more (or less) fields may display. See the [field descriptions](#), below, for a description of all fields that may display in the Workflow Task Details.



Workflow Task Details Field Descriptions

The following table describes the fields, buttons, and tabs that display in Workflow Task Details.

Field Name	Description
General	This section contains general information about the Workflow task.
Task Name	User-defined name of this task, containing a maximum 255 alphanumeric characters; variables supported. It is the responsibility of the user to develop a workable naming scheme for tasks.
Version	System-supplied; version number of the current record, which is incremented by the Controller every time a user updates a record. Click the Versions tab to view previous versions. For details, see Record Versioning .
Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
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.
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. <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p> Note Enabling this field will have no effect on Workflows; they cannot go to Failed status.</p> </div>
Workflow Details	This section contains assorted detailed information about the Workflow task.
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
Auto Layout	If Show/Hide Skipped Tasks = Hide Skipped ; specification for the layout of the Workflow Monitor. Options: <ul style="list-style-type: none"> • -- None -- • Horizontal Layout • Vertical Layout
Default Calendar	Default calendar used by the Workflow. If the Workflow is launched by a trigger, the trigger calendar overrides this default calendar.
Time Options	This section contains time-related specifications for the task.
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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.
Late Start Type	Required if Late Start is enabled. Options: <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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.

Late Start Duration	<p>If Late Start Type = 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.</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>
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.</p> <p>Options:</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 = Time; 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 = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.</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.</p> <p>Options:</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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.</p>
User Estimated Duration	<p>Estimated amount of time it should normally take to run this task. The Controller uses this information to calculate the User Estimated End Time on a task instance record.</p>
Statistics	<p>This section contains time-related statistics for task instances of the task.</p>
First Time Ran	<p>System-supplied; date and time this task first ran.</p>

Last Time Ran	System-supplied; date and time the task last ran.
Last Instance Duration	System-supplied; amount of time the task took to run the last time it ran.
Lowest Instance Time	System-supplied; shortest amount of time this task has taken to run.
Average Instance Time	System-supplied; average amount of time this task takes to run.
Highest Instance Time	System-supplied; longest amount of time this task has taken to run.
Number of Instances	System-supplied; number of instances in the database for this task.
Buttons	This section identifies the buttons displayed above and below the Workflow Task Details that let you perform various actions.
Save	Saves a new task record in the Controller database.
New	Displays empty (except for default values) Details for creating a new task.
Update	Saves updates to the record.
View Parents	Displays a list of any parent Workflow tasks for this task.
Launch	Manually launches this task.
Edit Workflow	Displays the Workflow Editor for this task.
Copy	Creates a copy of this Workflow task, which you are prompted to rename.
Delete	Deletes the current record.
Refresh	Refreshes any dynamic data displayed in the Details.
Close	In pop-up view only; closes the pop-up view of this task.
Tabs	This section identifies the tabs across the top of the Workflow Task Details that provide access to additional information about the task.
Workflow Task	Displays details about this Workflow task.
Task Run Criteria	Workflows only. Allows you to specify skip and run criteria for specific tasks in the workflow.
Variables	Lists all variables associated with this record.

Actions	<p>Allows you to specify actions that the Controller will take automatically based on events that occur during the execution of this task.</p> <p>Events are:</p> <ul style="list-style-type: none"> • Task instance status • Exit codes • Late start • Late finish • Early finish <p>Actions are:</p> <table border="1" data-bbox="269 478 1484 840"> <tr> <td data-bbox="269 478 412 548">Abort Action</td> <td data-bbox="417 478 1484 548">Abort the task if certain events occur. For details, see Abort Actions.</td> </tr> <tr> <td data-bbox="269 554 412 623">Email Notification</td> <td data-bbox="417 554 1484 623">Send an email if certain events occur. For details, see Email Notification Actions.</td> </tr> <tr> <td data-bbox="269 630 412 699">Set Variable</td> <td data-bbox="417 630 1484 699">Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow.</td> </tr> <tr> <td data-bbox="269 705 412 774">SNMP Notification</td> <td data-bbox="417 705 1484 774">Send an email if certain events occur. For details, see SNMP Notification Actions.</td> </tr> <tr> <td data-bbox="269 781 412 850">System Operation</td> <td data-bbox="417 781 1484 850">Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions.</td> </tr> </table>	Abort Action	Abort the task if certain events occur. For details, see Abort Actions .	Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .	Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .	SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .	System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .
Abort Action	Abort the task if certain events occur. For details, see Abort Actions .										
Email Notification	Send an email if certain events occur. For details, see Email Notification Actions .										
Set Variable	Used in tasks and workflows to set a variable based on the occurrence of certain events. For details, see Creating a Set Variable Action within a Task or Workflow .										
SNMP Notification	Send an email if certain events occur. For details, see SNMP Notification Actions .										
System Operation	Run an Opwise Controller system operation based on specified conditions. For details, see System Operation Actions .										
Virtual Resources	Lists all Virtual Resources to which this task is assigned.										
Mutually Exclusive	Lists all tasks that have been set to be mutually exclusive of this task.										
Step Conditions	Lists all step conditions defined for this task.										
Step Actions	Lists all step actions created for this task.										
Instances	Lists all instances of this Workflow.										
Triggers	Lists 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, the Controller 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 Triggers Overview .										
Notes	Lists all notes associated with this record.										
Versions	Stores copies of all previous versions of the current record. See Record Versioning .										

Viewing a Workflow Task Instance

When a Workflow task is launched, the Controller creates a task instance record of that Workflow (and task instance records of all tasks in that Workflow).

A task instance contains detailed information about a single execution of that task.

You can access a Workflow task instance from:

- **Instances tab** on the [Workflow Task Details](#) for that task
- [Activity Monitor](#)
- [Task Instances list](#)


Workflow Task Instance Details

The following Workflow Task Instance Details contains information on the execution of the Workflow shown in the [Workflow Task Details](#).

Workflow Task Instance Details Field Descriptions

The following table describes the fields, buttons, and tabs that display in Workflow Task Instance Details.

Field Name	Description
General	This section contains general information about the task instance.
Instance Name	Name of this task instance.
Reference Id	System-supplied; the Controller increments this number each time the task is run.
Task	Name of the task that was run to create this task instance. Click the icon to display Task Details for the task.
Invoked by	System-supplied; how the task instance was launched. Options: <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 identify the user, check the Execution User column for that task instance on the Task Instances screen or, on most task instance screens, the Execution User field.

Task Description	User-defined; description of this record. (Maximum = 200 characters.)
Member of Business Services	User-defined; allows you to select one or more Business Services that this record belongs to.
Execution User	System-supplied; if the task was launched manually, the ID of the user who launched it.
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. <div style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;">  Note Enabling this field will have no effect on Workflows; they cannot go to Failed status. </div>
Progress	Number of tasks in the Workflow that are in Success, Skipped, or Finished status / Total number of tasks in the Workflow.
Status	This section contains information about the current status of the task instance.
Status	System-supplied; see Task Instance Statuses .
Status Description	System-supplied; additional information, if any, about the status of the task instance.
Queued Time	System-supplied; Date and time the task was queued for processing.
Trigger Time	System-supplied; Date and time the task instance was triggered.
Launch Time	System-supplied; Date and time the task instance was launched.
Start Time	System-supplied; Date and time the task instance started.
End Time	System-supplied; Date and time the task instance completed.
Duration	System-supplied; amount of time the task instance took to run.
Workflow Details	This section contains assorted detailed information about the task instance.

Show/Hide Skipped Tasks	<p>Specification to either show or hide tasks that have been skipped in a Workflow (see Skipping a Task).</p> <p>Options:</p> <ul style="list-style-type: none"> • Show Skipped • Hide Skipped
Time Options	This section contains information on time-related specifications for the task instance.
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. The Started Late field displays in the task instance Details only if the user specified a Late Start in the task Details.</p>
Late Start Type	<p>Required if Late Start is enabled.</p> <p>Options:</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	If Late Start Type = Time; Time after which the task start time is considered late. Use HH:MM, 24-hour time.
Late Start Duration	<p>If Late Start Type = 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.</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	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.</p> <p>Options:</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 = Time; Time after which the task finish time is considered late. Use HH:MM, 24-hour time.
Late Finish Duration	If Late Finish Type = Duration; Longest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.

Finished Late	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: <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 = Time; 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 = Duration; Shortest amount of time this task instance should take to run. You can specify any combination of hours, minutes, and seconds.
Finished Early	System-supplied; this field is flagged if the task finished earlier than the time specified in the Early Finish fields.
Statistics	This section contains time-related statistics for the task instance.
User Estimated End Time	System-supplied; if the user entered information into the User Estimated Duration field in the task Details, the Controller uses this information to calculate an end time for the task instance, based on the date/time the task instance started.
Average Estimated End Time	System-supplied; average estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Shortest Estimated End Time	System-supplied; shortest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Longest Estimated End Time	System-supplied; longest estimated end time of the task instance, calculated by the Controller based on the date/time the task instance started.
Buttons	This section identifies the buttons displayed above and below the Workflow Task Instance Details that let you perform various actions.
Update	Saves updates to the record.
Force Finish	See Force Finishing a Task .
Hold	Places the task instance on Hold (see Putting a Task on Hold).
Skip	For tasks loaded into the schedule that have not yet run; allows you to tell the Controller to skip this task. See Skipping a Task .

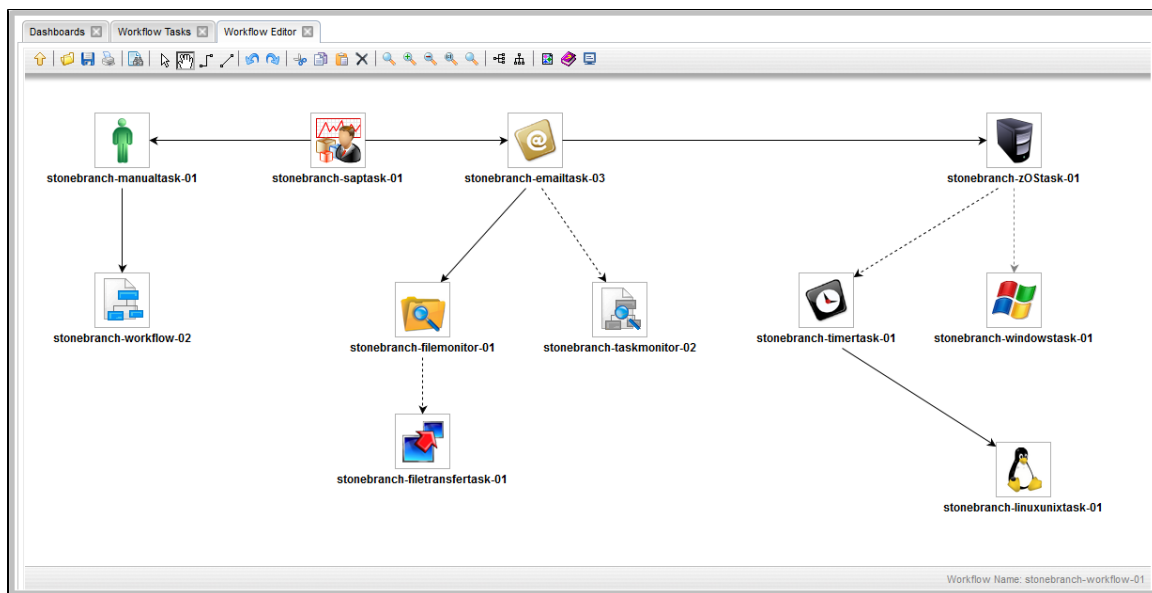
Release Recursive	Releases the entire held workflow and its tasks (see Releasing a Task From Hold).
View Parent	Displays the task instance Details for the parent Workflow of this task instance.
View Workflow	Displays the graphical workflow.
Delete	Deletes the current record.
Refresh	Refreshes any dynamic data displayed in the Details.
Close	For pop-up view only; closes the pop-up view of this task instance.
Tabs	This section identifies the tabs across the top of the Workflow Task Instance Details that provide access to additional information about the task instance.
Workflow Task Instance	Displays details about this Workflow task instance.
Virtual Resources	Lists all Virtual Resources to which this task is assigned.
Exclusive Requests	Lists all records in the Exclusive Requests table (<code>ops_exclusive_order</code>) for this task instance.
Step Conditions	Lists all step conditions defined for this task.
Notes	Lists all notes associated with this record.

Designing a Workflow

The Workflow Editor lets you select tasks for a Workflow, position them within the Workflow, and specify dependency relationships between the tasks.

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 is a sample Workflow Editor for a Workflow consisting 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; you can also put other Workflows into a Workflow.



See [Workflow Editor Icons](#) for a guide to the editing tools available for designing Workflows; for more detailed instructions, see the individual topics in this section, below.

Designing, Modifying, or Viewing Multiple Workflows

You can design, modify, or simply view multiple Workflows in the Workflow Editor at the same time.



Step 1	From the Workflow Tasks list, open the Workflow task for which you want to design, modify, or view the Workflow.
Step 2	Click the Edit Workflow button to display the Workflow Editor for the Workflow.
Step 3	Click the Workflow Tasks tab at the top of the Workflow Editor (next to the Workflow Editor tab) to redisplay the Workflow Tasks list.
Step 4	Open another Workflow task for which you want to design, modify, or view the Workflow.
Step 5	Click the Edit Workflow button to display the Workflow Editor for this second Workflow under a second tab.
Step 6	Open and display as many Workflows as desired, clicking the appropriate tab to create, modify, and view each one.

The actions you can perform within the Workflow Editor depend on the Workflow Editor mode.

Workflow Editor Modes

The Workflow Editor has three modes. Each mode, which you select by clicking the appropriate icon in the [Workflow Editor toolbar](#), allows you to use one or more specific Workflow Editor [tools](#).

Icon	Mode	Description
	Select	Clicking the Select icon on the Workflow Editor toolbar places the Workflow Editor in Select mode, which allows you to use all Workflow Editor tools except Connect and Pan . The Select mode is the default mode for the Workflow Editor.
	Connect	Clicking a Connect icon on the Workflow Editor toolbar places the Workflow Editor in Connect mode, which allows you to create connections between tasks. In Connect mode, you also can use all other Workflow Editor tools except Connect .

	<p>Pan</p>	<p>Clicking the Pan icon on the Workflow Editor toolbar places the Workflow Editor in Pan mode, which allows you to scroll around in the Workflow. (You also can use the Outline tool in any mode) to scroll around in the Workflow.</p> <p>In Pan mode, you also can use all other Workflow Editor tools except Connect.</p> <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note Pan mode does not allow you to use the drag and drop method for moving, deleting, and copying groups of Workflow elements.</p> </div>
---	-------------------	--

Workflow Editor Toolbar

The Workflow Editor toolbar, which displays above the Workflow Editor, contains icons for every tool that you need to design a Workflow.

A shadowed outline around the icon in the toolbar indicates the currently selected [Workflow Editor mode](#).



The following example shows that the Workflow Editor is in **Select** mode.























You also can perform all [task actions](#) and [connector actions](#) in any mode.

Workflow Editor Tools

The following table describes the tools used to define Workflows, including the tools associated with the three [Workflow Editor modes](#). The tools are listed by icon, as they appear - left to right - in the [Workflow Editor toolbar](#).

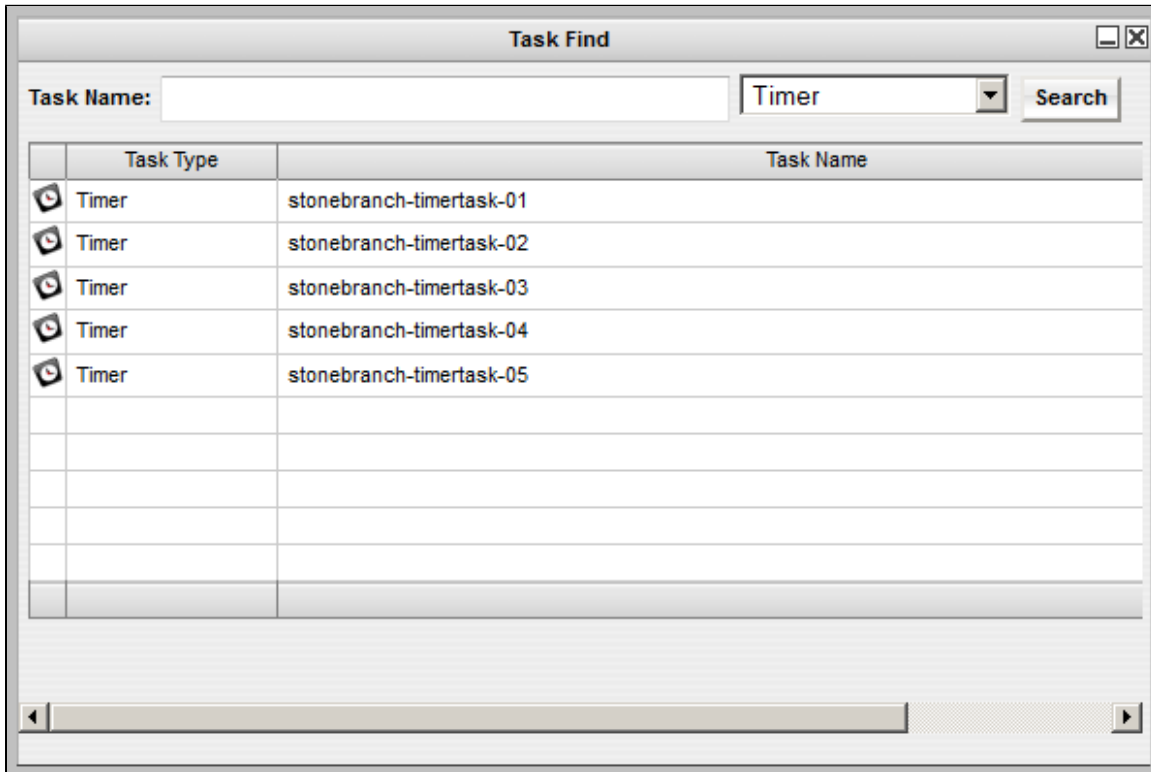
Icon	Tool	Description
	<p>Go to Parent</p>	<p>If this Workflow is embedded in another Workflow, navigates to the parent. If this Workflow has more than one parent, clicking this icon displays a list of all parent Workflows; in this case, to navigate to a parent Workflow, double-click the Workflow's icon. To return to your original Workflow, click its name. (The current Workflow name displays in the lower corner of the editor.)</p> <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note To open an embedded Workflow (sub-Workflow) from its Parent Workflow, double-click the embedded Workflow.</p> </div>

	Pan	<p>Places the Workflow Editor in Pan mode.</p> <p>To pan around the Workflow, click anywhere in the Workflow Editor canvas and drag your cursor to display any section of the Workflow.</p> <p>To return to your original location, click the Actual Size icon.</p>
	Connect (bent) Connect (straight)	<p>Places the Workflow Editor in Connect mode.</p> <p>To create a connection:</p> <ol style="list-style-type: none"> 1. Click either icon. 2. Click a predecessor task. If you are using Firefox, click the center of the task icon. If you are using Internet Explorer, click the task name. 3. Drag the cursor to the successor task. You will see a colored line as you drag. 4. When you reach the successor task, release the cursor. The connector 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	<p>Deletes the selected element(s) (tasks, connectors, or both) and keeps a copy in memory. Use Ctrl+Click to select an multiple elements.</p> <div style="background-color: #ffffcc; padding: 5px; margin-top: 10px;"> <p> Note Do not use cut and paste to move Workflow elements; use select and drag.</p> </div>
	Copy	Copies the selected element(s) (tasks, connectors, or both). Use Ctrl+Click to select and copy multiple elements.
	Paste	Pastes the copied or cut element(s) to the currently open Workflow.
	Delete	Permanently deletes the selected elements. Delete does not keep a copy of the deleted elements 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 clicking the Actual Size icon.
	Zoom In	Zooms in (enlarges) the Workflow. To return the Workflow to its default size, click the Actual Size icon.
	Zoom Out	Zooms out (diminishes) the Workflow. To return the Workflow to its default size, click the Actual Size icon.
	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, click the Actual Size icon.
	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 (see Monitoring Workflows). This icon only appears on the icon bar when you are monitoring a running Workflow.
	Outline	<p>For large Workflows, the outline provides a way of positioning a specific area of the Workflow in the display, without using Pan mode.</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 Console, see Monitoring Workflows .

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](#). 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**.

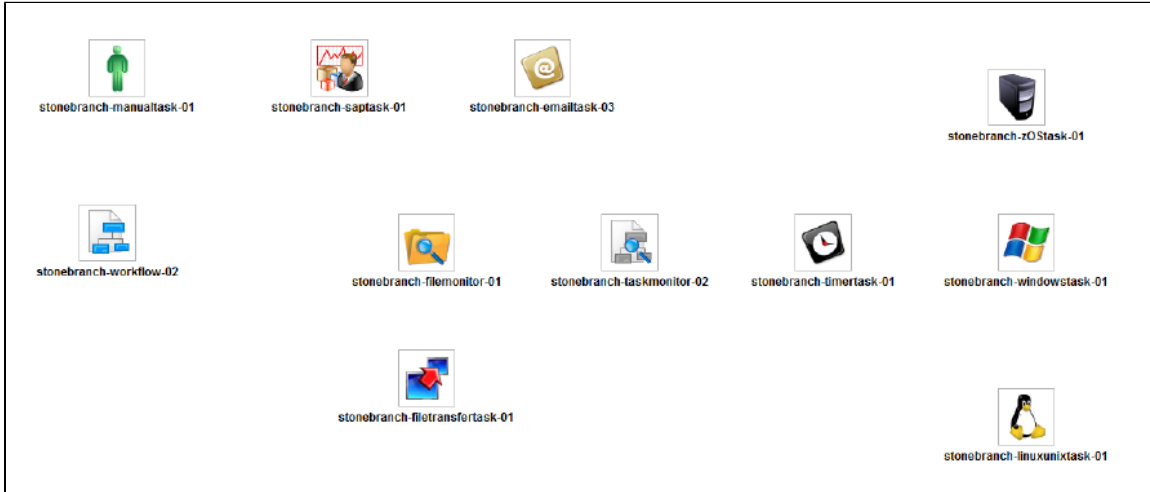
The Task Find dialog lists the task(s) that match your search criteria.

**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.

You can add any task to a Workflow multiple times. To differentiate between multiple additions of the same task in a Workflow, you can assign each of them a unique name used only within the Workflow; the name of the task in the Task Details will not change. To change the name of a task within a Workflow, display the [Task Properties](#) menu and select a new name in the **name** field.

Step 4 Repeat these steps until you have added all the tasks you need.

Step 5 Position the tasks on the canvas as desired. The connections that you will make between the tasks determine the order in which the tasks run, so position the tasks accordingly. For example:

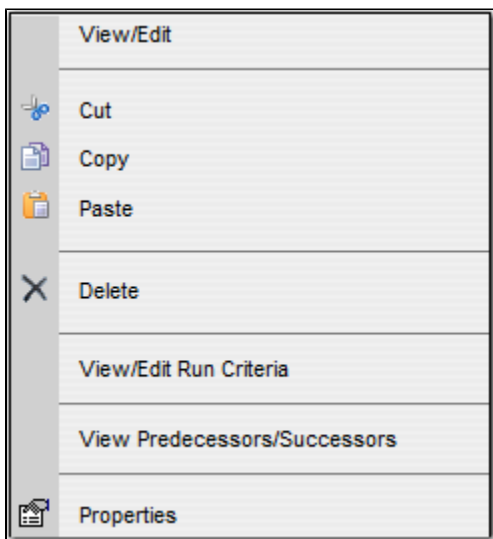


Step 6 To close the **Task Find** window, click the **X** in the upper right corner. Or, to keep the window open but minimized, click the minimize icon.

Task Actions

For each task in a Workflow, the Controller provides a menu of actions that you can take on that task. You can perform task actions in any [Workflow Editor mode](#).

To display the menu of task actions, right-click a task.



The following table describes each action in the menu:

Action	Description
View/Edit	Allows you to view and edit the Task Details for this task.
Cut	Cuts the task from its location in the Workflow. You can paste a cut task back into the Workflow.
Copy	Creates a copy of the task for inclusion in the Workflow.
Paste	Pastes a copy of the task into the Workflow.
Delete	Deletes the task from the Workflow. You cannot paste a deleted task back into the Workflow.

View/Edit Run Criteria	Allows you to view and edit any run criteria that has been created for this task.
View Predecessors/Successors	Allows you to view all predecessors and successors of this task in the Workflow.
Properties	Displays a menu of properties for the task.

Task Properties

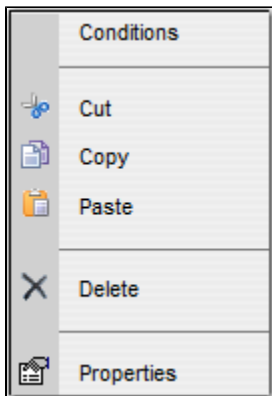
One of the [task actions](#) available for any task in a Workflow is to display a list of properties for that task.

You can change the name of a task within a Workflow by entering a new name in the **name** field. By Default, the **name** of a task is its name in the Task Details. Changing the **name** of a task within a Workflow does not change its name in the Task Details.

Connector Actions

For each connector in a Workflow, the Controller provides a menu of actions that you can take on that connection. You can perform connector actions in any [Workflow Editor mode](#).

To display the menu of connector actions, right-click a connection.



The following table describes each action in the menu:

Action	Description
--------	-------------

Conditions	Allows you to select a dependency condition for the connection.
Cut	Cuts the task from its location in the Workflow. You can paste a cut task back into the Workflow.
Copy	Creates a copy of the task for inclusion in the Workflow.
Paste	Pastes a copy of the connector into the Workflow.
Delete	Deletes the connector from the Workflow. You cannot paste a deleted connector back into the Workflow.
Properties	Displays a menu of properties for the connector.

Connector Properties

One of the [connector actions](#) available for any connector in a Workflow is to display a list of properties for that connector.

The screenshot shows a 'Properties' dialog box with the following fields and values:

- ID: 21
- Style: straightEdge
- label: (empty)
- description: (empty)
- condition: SUCCESS

Buttons: OK, Cancel

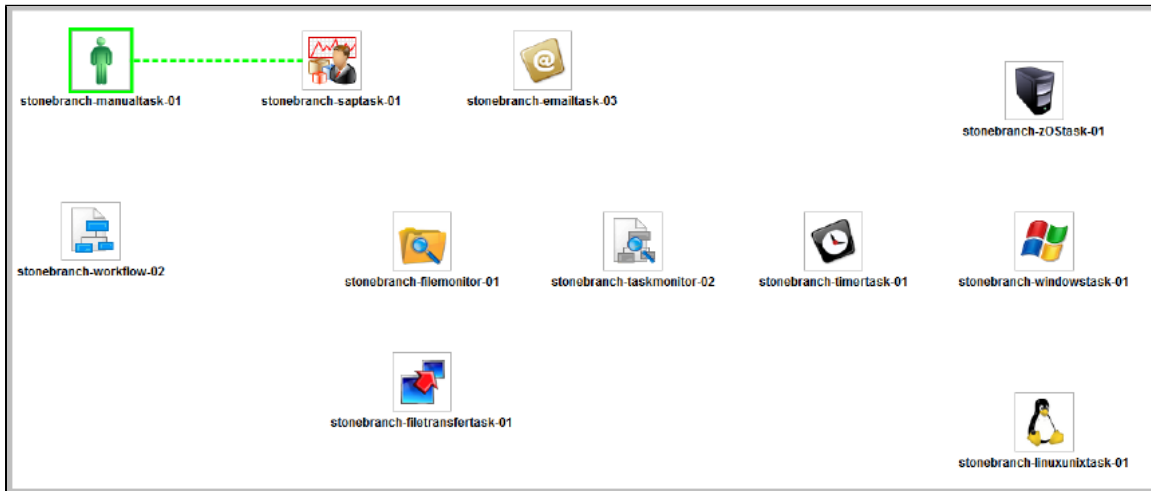
The information in the Connector properties is view-only.

Specifying Connections

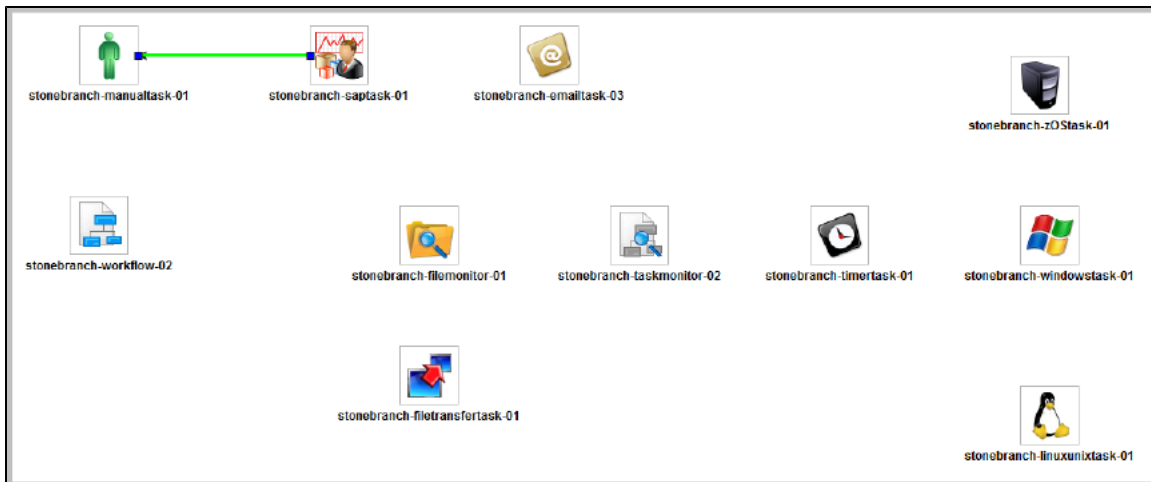
After you have added tasks and positioned them on the Workflow Editor canvas, you must connect the tasks.

Step 1	Click one of the Connect icons to place the Workflow Editor in Connect mode .
Step 2	Click the predecessor task; that is, the task whose status will trigger the next task. <ul style="list-style-type: none"> • 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 successor 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 successor, highlighted task.



Step 4 When the cursor is at the center of the successor 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 to be added.

Specifying Conditions on Connections

When you create a connection between the tasks in a Workflow, you can specify a dependency condition between the predecessor and successor task. The successor task will run if the condition that you specify matches the condition of the predecessor task.

By default, a Success condition is specified automatically for a connection when you create it; if the predecessor task runs to Success, the successor task will run.

The dependency condition that you select for a connection, as allowed for the task type of the predecessor task (see [Conditions per Task Type](#), below), determines the format of the connector in the Workflow Editor.

Condition	Connector Format	Description
Success	Solid black line	Run the successor task if the predecessor task goes to Success.
Failure	Dotted black line	Run the successor task if the predecessor task goes to Failure.
Success/Failure	Dotted black line	Run the successor task if the predecessor task goes to Success or Failure.

Exit Code(s)	Dotted black line	Run the successor task if the predecessor 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.
Step Conditions	Dotted gray line	For predecessor z/OS tasks only: Run the successor task if the running predecessor z/OS task has a step end matching the specified Step Condition(s).

Connector Graphics for All Condition Types, below, illustrates a composite Workflow containing connectors for all condition types.

Conditions per Task Type

The following table identifies the dependency conditions that are supported for each type of task:

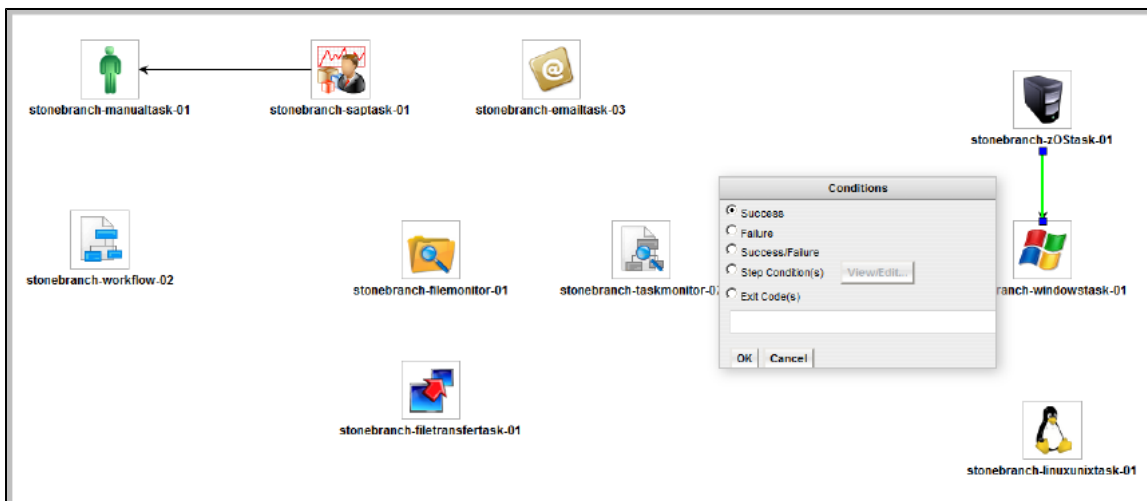
Task Type	Success	Failure	Success/Failure	Step Condition(s)	Exit Code(s)
Workflow	✓				✓
Linux/Unix	✓	✓	✓		✓
Windows	✓	✓	✓		✓
z/OS	✓	✓	✓	✓	✓
Indesca	✓	✓	✓		✓
SAP	✓	✓	✓		✓
File Transfer	✓	✓	✓		✓
Manual	✓				✓
Timer	✓				✓
SQL	✓	✓	✓		✓
Stored Procedures	✓	✓	✓		✓
Email	✓	✓	✓		✓
Task Monitor	✓	✓	✓		✓
File Monitor	✓	✓	✓		✓
FTP File Monitor	✓	✓	✓		✓
System Monitor	✓	✓	✓		✓
Application Control	✓	✓	✓		✓

Specifying a Condition

To specify a dependency condition for a connection between a predecessor and successor task:

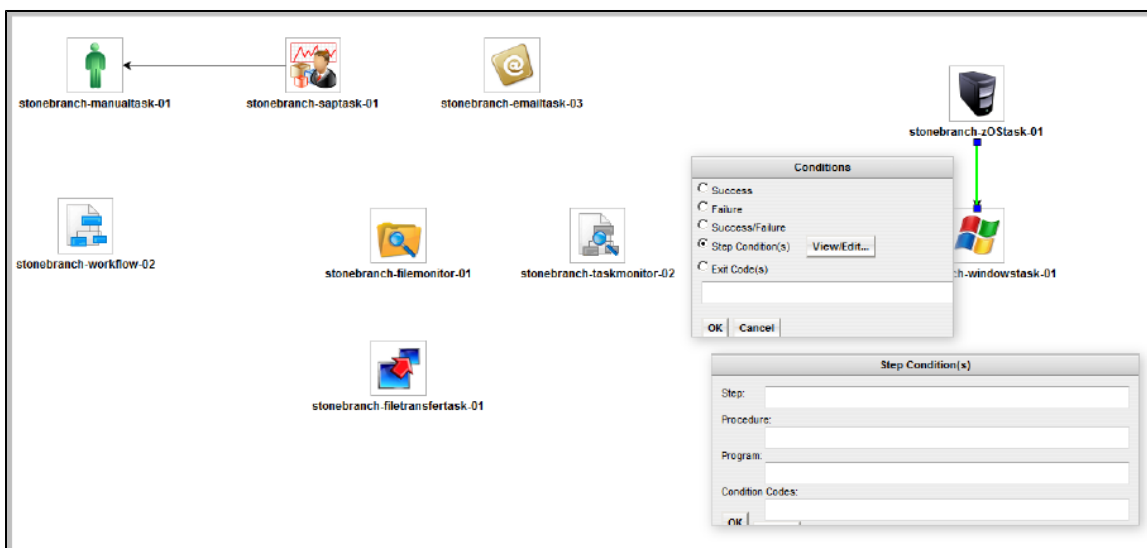
Step 1	Hover your mouse over the connection to view the currently selected condition. If you want to change the condition, right-click the connection to display a Connector Actions menu.
---------------	---

Step 2 Select **Conditions**. The Conditions dialog displays.



Step 3 Specify a condition (as [supported](#) for each task type).

Step 4 If you selected **Step Condition(s)** for a connection to a predecessor z/OS task, click the **View/Edit...** button to display a Step Conditions dialog that lets you define the step conditions.



Using the field descriptions [below](#) as a guide, complete the fields and click **OK**.

Step 5 Click **OK** on the Conditions dialog.

Step 6 To see the condition that is specified for a connection, hover your cursor over the connector (see [Connector Graphics for All Condition Types](#), below).

Step Conditions

A Step Condition - for a z/OS task only - is an intermediate condition that is evaluated immediately after each z/OS step end. In other words, it does not specify a condition to be met when the z/OS task completes.

In this way, Step Conditions allow a successor task to run before its predecessor task has completed. The successor task may require only that a step in the z/OS task completes, not the entire task.

Step Condition connectors display in gray to distinguish the difference between intermediate Step Conditions and task completion Conditions.

On z/OS task instance completion - specifically, Failed, Success or Finished - Step Conditions are not relevant and are not evaluated. If one or more non-Step Condition dependencies were satisfied, all remaining unsatisfied dependencies, including Step Condition dependencies, become unreachable and their associated paths are skipped.

Step Condition(s) Field Descriptions

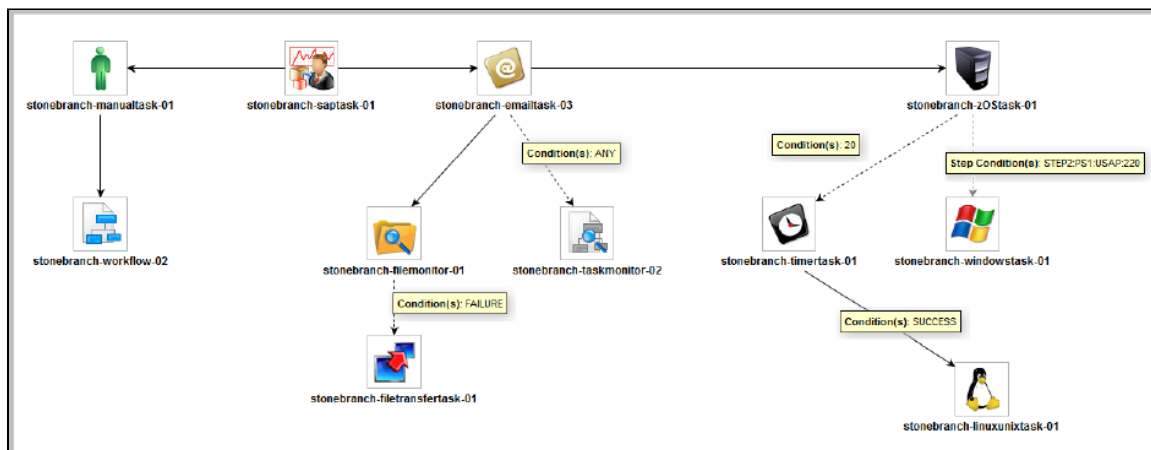
The following table describes the fields on the Step Condition(s) dialog.

Field Name	Description
Step	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	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	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. For example: 1,6-4095,Sxxx,Unnnn,JCLERR

Connector Graphics for All Condition Types

The following illustration is a composite Workflow showing the connector formats for all **condition types** and the information that displays for each condition type when you hover your cursor over the connector.

Condition	Hover Display
Success	Condition(s): SUCCESS
Failure	Condition(s): FAILURE
Success/Failure	Condition(s): ANY
Exit Code(s)	Condition(s): Exit Code(s)
Step Condition(s)	Step Condition(s): Step, Procedure, Program, Condition Codes



Creating Conditional Paths

The Controller allows you to specify a separate processing path for each possible condition between a predecessor task and a successor task. The Controller selects the appropriate conditional path when the predecessor task goes to a finished status (Success or Failure). As a result, at least one successor dependency is satisfied and at least one successor dependency is not satisfied.

For example, you could specify a conditional path containing a group of tasks that will run if the predecessor task goes to Success, and a second conditional path containing a group of tasks that will run if the predecessor task goes to Failure. When the predecessor task has run to completion, the tasks in the path whose condition meets the condition of the predecessor task will run, and the tasks in the path whose condition does not meet the condition of the predecessor task will go to a Skipped status.

Further examples:

Example 1

Task A is at the top of the Workflow. Three conditional paths have been specified for Task A: one for Failure and two for Success. Task A executes and goes to Success status. The Controller places all of the tasks in the Failure conditional path into Skipped status and the tasks in the two Success conditional paths begin running.

Example 2

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 tasks in the first path run and the second path is skipped.
- If Task A completes with exit code 15, the first path is skipped and the tasks in the second path run.
- If Task A completes with exit code 25, neither condition is satisfied and both paths remain in Waiting status.

Conditions for Multiple Connections

If a task is connected to multiple predecessor tasks, that successor task remains in Waiting status until the condition of each of its immediate predecessor tasks is evaluated. If all of its immediate predecessor tasks have been Skipped, the successor task is Skipped. If at least one of its immediate predecessor tasks has run to completion, the successor task will run if the specified dependency condition matches the condition of the predecessor task.

For example, if a task has four immediate predecessor tasks, and three of the immediate predecessor tasks have been Skipped and the fourth predecessor task has failed, and the dependency condition between the fourth predecessor task and the successor task is Failure, the successor task will run. (Likewise, if the fourth predecessor task has run to Success, and the dependency condition between the fourth predecessor task and the successor task is Success, the successor task will run.)

The status of the tasks in the conditional paths leading to the immediate predecessor tasks do not affect whether or not the successor tasks will run. All tasks leading to an immediate predecessor tasks do not have to have run in order for the successor task to run.

Moving Workflow Elements

After you have positioned tasks and connectors on the Workflow Editor canvas, you can move these elements as needed.

Move a Single Task

Click a task and drag it to a new location. If the task has any connectors attached, they remain connected, and lengthen or shorten as necessary.

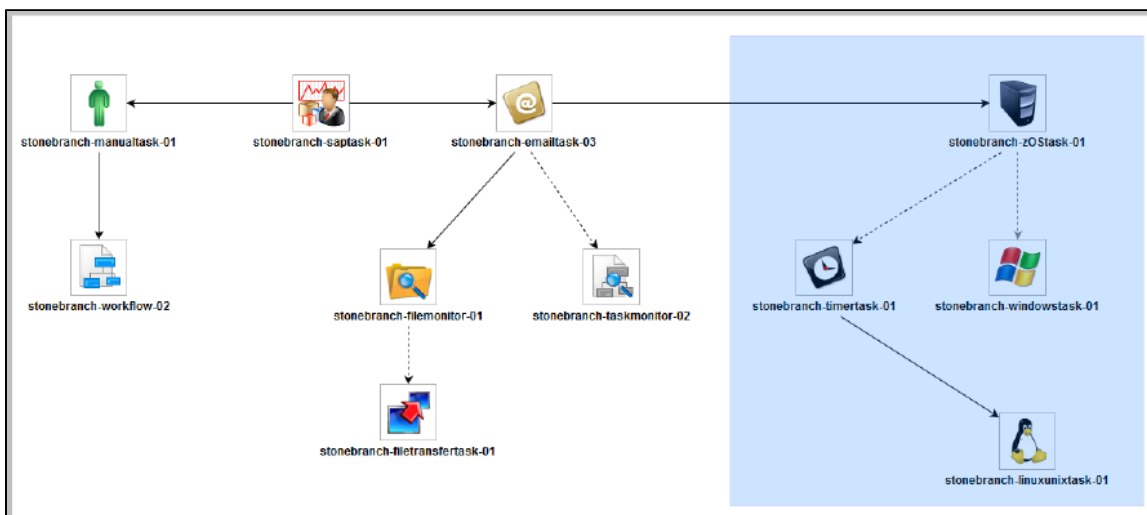
Move a Single Connector

Step 1	Click the connector you want to move.
Step 2	Position your cursor over either end of the connector (at the predecessor or successor task) and, when the cursor changes to a "pointing hand" icon, drag that end of the connector so that it connects to a different task.
Step 3	As desired, repeat Step 2 for the other end of the connector.

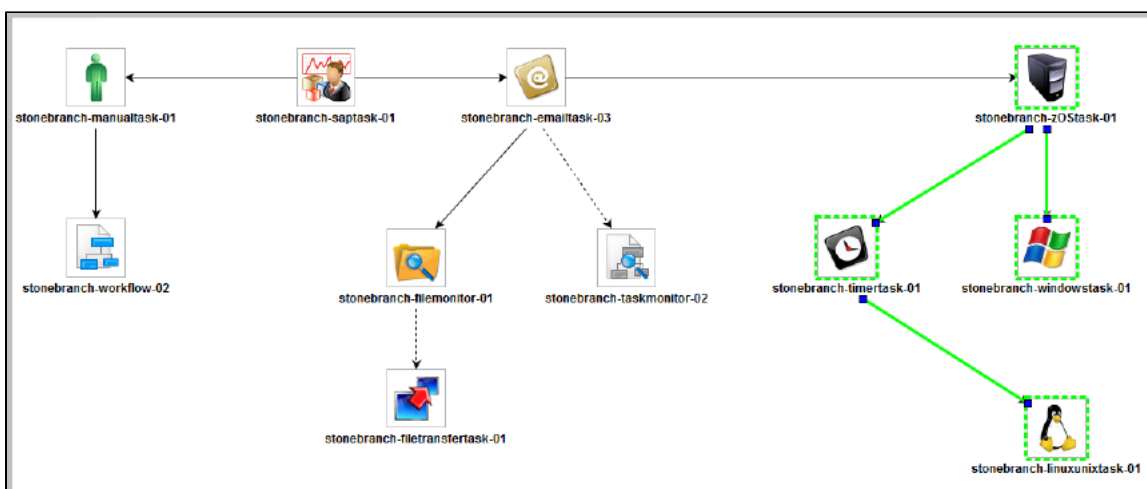
Move a Group of Tasks and Connectors

Step 1	Click the Select icon or either Connect icon.
Step 2	Position the cursor near the group that you want to move.
Step 3	Click the Workflow Editor canvas and begin dragging. A blue shaded box appears. This identifies the selection area.

Step 4 Drag the selection area over the tasks you want to select, as shown in the following example:



Step 5 When you have selected the group of tasks and connectors that you want to move, release the mouse. The tasks and connectors included in the group are highlighted, as shown:



Step 6 Hover your cursor over any element in the group until the Move cursor displays, then click that element and drag the group to the new location. All selected elements are moved.

Step 7 To deselect the group, click elsewhere in the Workflow Editor canvas.



Note

You also can use Ctrl+click to select a group of elements or a selection of individual elements and drag them to a new location.

Deleting Workflow Elements

Step 1 Click an element, drag your cursor across the Workflow Editor canvas to select a group of elements, or use Ctrl+click to select one element, a group of elements, or a selection of individual elements.

(You cannot drag your cursor across the Workflow Editor canvas in *Pan mode*.)

Step 2 Click the [Delete](#) icon. The selected element or group of elements is deleted.

Copying Workflow Elements

Step 1	Click an element, drag your cursor across the Workflow Editor canvas to select a group of elements, or use Ctrl+click to select one element, a group of elements, or a selection of individual elements. (You cannot drag your cursor across the Workflow Editor canvas in Pan mode .)
Step 2	Click the Copy icon to create a copy of the selected elements.
Step 3	Click the Paste icon to paste the copied element(s) into the Workflow next to the original element(s).
Step 4	Click the copied element (or any element in a copied group) and drag it to a new location.

Undoing and Redoing Workflow Changes

To undo the most recent change, click the [Undo](#) icon. To redo a change you just undid, click the [Redo](#) icon.

Zooming In and Out

Several features are available for zooming in and out on large Workflows:

- Click the [Fit](#) icon to fit the entire Workflow onto the display.
- Click the [Zoom In](#) icon to increase the size of the Workflow (to view details).
- Click the [Zoom Out](#) icon to decrease the size of the Workflow (to view the entire Workflow)
- Click the [Actual Size](#) icon to return the Workflow to its actual (original) size.
- Click the [Zoom](#) icon to enter a zoom percentage.

Panning Around in Large Workflows

For large Workflows that cannot be viewed entirely on your screen, you can pan around from area to area.

Two methods are provided:

- [Pan mode](#)
- [Outline](#) tool (in any mode)

Pan Mode Method

Step 1	Click the Pan icon to enter Pan mode .
Step 2	Click the Workflow Editor canvas and drag the Workflow so that it displays the area you want to work on.

Outline Tool Method

Step 1	Click the Outline icon icon to open the Outline window.
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...](#) tool.

Workflow Coordinates

If you use [Pan mode](#) (or [Find in Graph...](#)) to view different areas of a Workflow, the Controller preserves the coordinates of the last area that you viewed. If you leave the Workflow and return to it in the same browser session, the last viewed area of the Workflow displays.

To restore the displayed Workflow to its 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 the [Horizontal Layout](#) icon.
- To create a vertical layout, click the [Vertical Layout](#) icon.

Displaying Workflow Documentation

To display documentation on [monitoring Workflows](#), 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 (see [Designing, Modifying, or Viewing Multiple Workflows](#)).

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. The Controller 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.



Note

Calendar-based criteria are always evaluated using the task instance [Trigger Time](#) (or the task instance [Created time](#), if the workflow was launched manually).

When a workflow is launched or triggered, the Task Run Criteria provide a mechanism to decide whether a specific task (or sub-Workflow) within the Workflow should be run or skipped. If no Task Run Criteria are specified, a task will always run when the Workflow is launched or triggered. If Task Run Criteria are specified to run or skip a task, these will be evaluated as follows:

- Skip criteria will run the task unless the skip criteria is met. For example, a workflow is triggered every business day, and a task has a skip criteria of Tuesday. The task will run Monday, Wednesday, Thursday, and Friday. The task will not run on Tuesday.
- Run criteria will only run the task when the run criteria is met. For example, a workflow is triggered every business day and a task has a run criteria of Tuesday. The task will run on Tuesday. The task will be skipped on Monday, Wednesday, Thursday, and Friday.

Multiple conditions can be specified in a single Task Run Criteria definition. In this case, all of the conditions need to be met. For example, if in a single Task Run Criteria you specify for the task to run on a specific day of Tuesday, and if the variable RUN_ME resolves to the value YES at Run Time, the task will be skipped unless both of these conditions are true.

When multiple Task Run Criteria definitions are specified for a single task, these are evaluated independently.



Note

If skip criteria and run criteria both are met (true) on a given day, the skip criteria will take precedence.

Creating New Run Criteria

Step 1	Display the Details of the Workflow for which you want to specify run criteria.
Step 2	Click the Task Run Criteria tab. The Task Run Criteria list displays a list of any existing task run criteria.

Step 3 Click **New**. **Task Run Criteria Details** displays:



Step 4 Using the [field descriptions](#), below, as a guide, complete the fields as needed. For example, to tell the Controller 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 **Save** button to save the record.

Task Run Criteria Field Descriptions

The following table provides descriptions for each field in the Task Run Criteria Details.

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; task for which you are specifying run or skip criteria. Select a task from the drop-down list of all tasks in this Workflow.
Vertex ID	Vertex ID of the task selected in the Task field. 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.</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. Enter the name of an existing Custom Day, select a Custom Day from the drop-down list of all existing Custom Days, or click the De3tails icon to create a new Custom Day.</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.</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, Qualifier, and Nth Amount (if Adjective = Nth) fields. For example, to specify "the 15th business day of the month," select Adjective: Nth, Noun: Business Day, Qualifier: Month, Nth Amount: 15.</p> <p>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, Qualifier, and Nth Amount (if Adjective = Nth) fields. For example, to specify "the 1st business day of the month," select Adjective : 1st, Noun: Business Day, Qualifier: Month.</p> <p>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 Workflow definition Default Calendar field. Custom Days specified in the calendar selected in the Workflow definition Default 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 Adjective, Noun, and Nth Amount (if Adjective = Nth) fields. For example, to specify "the 1st business day of the month," select Adjective: 1st, Noun: Business Day, Qualifier: Month.</p> <p>Options:</p> <ul style="list-style-type: none"> Month Year January through December Custom Period (see Creating Custom Days)
Nth Amount	<p>If Adjective = Nth; Sequential occurrence of the day selected in Noun within the selected Qualifier.</p>
Variable	<p>If enabled, instructs the Controller 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.</p>

Evaluate At	<p>If Variable = enabled; Allows you to specify when you want the Controller to evaluate the variable.</p> <p>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. <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p> Note Evaluate At is applicable only to Variable criteria (it is hidden until the Variable field is enabled). Evaluate At has no impact on other criteria, which will resolve the same way whether the Evaluate At value is Trigger Time or Run Time; calendar-based criteria is always evaluated using the task instance Trigger Time (or the task instance Created time, if the workflow was launched manually).</p> </div>
Name	If Variable = enabled; Specifies the name of the variable being evaluated. The variable does not need to exist when this record is created.
Operator	If Variable = enabled; Allows you to select the operator to be used in the evaluation. Options: =, !=, >, >=, <, <=, regex (regular expression).
Value	If Variable = enabled; Allows you to specify the value or regular expression (up to 40 alphanumeric characters) that the Controller should look for when evaluating the variable.
Buttons	This section identifies the buttons displayed above and below the Task Run Criteria that let you perform various actions.
Save	Saves a new record in the Controller database.
Update	Saves updates to the record.
New	Displays empty (except for default values) Details for creating a new Database Connection.
Delete	Deletes the current record.
Refresh	Refreshes any dynamic data displayed in the Details.
Close	For pop-up view only; closes the pop-up view of this Database Connection.

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.

You also can specify [skip and run criteria](#) for specific tasks within the Workflow.

Monitoring Workflow Execution

You can monitor all Controller activity from the [Activity Monitor](#).

Modifying an Existing Workflow

Step 1	From the Automation Center navigation pane, select Tasks > Workflow Tasks . The Workflow Tasks list displays.
Step 2	Click the Task Name of the Workflow task that you want to modify. Workflow Task Details for that task displays.
Step 3	Click Edit Workflow to display the Workflow Editor .
Step 4	Modify the Workflow and click the Save icon.

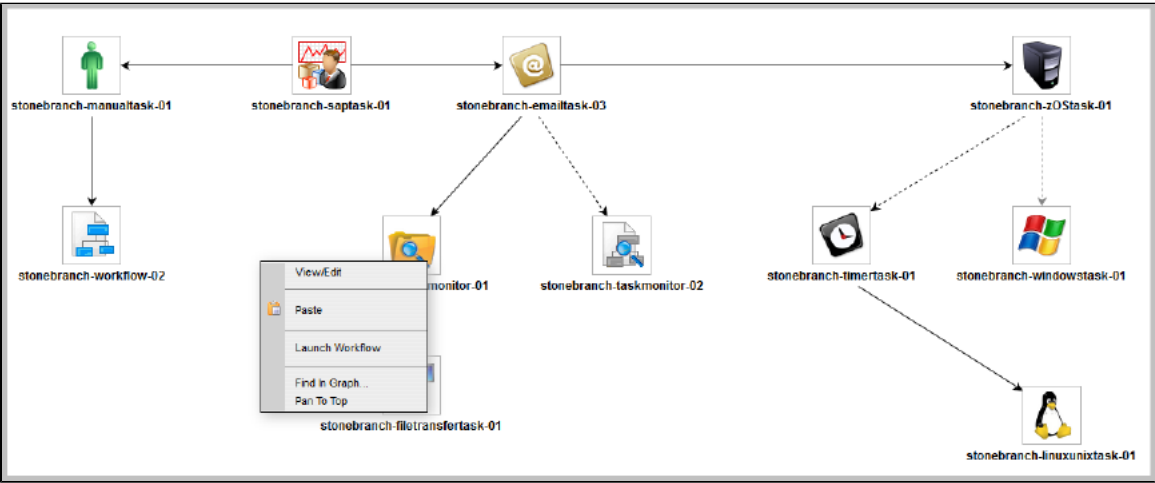
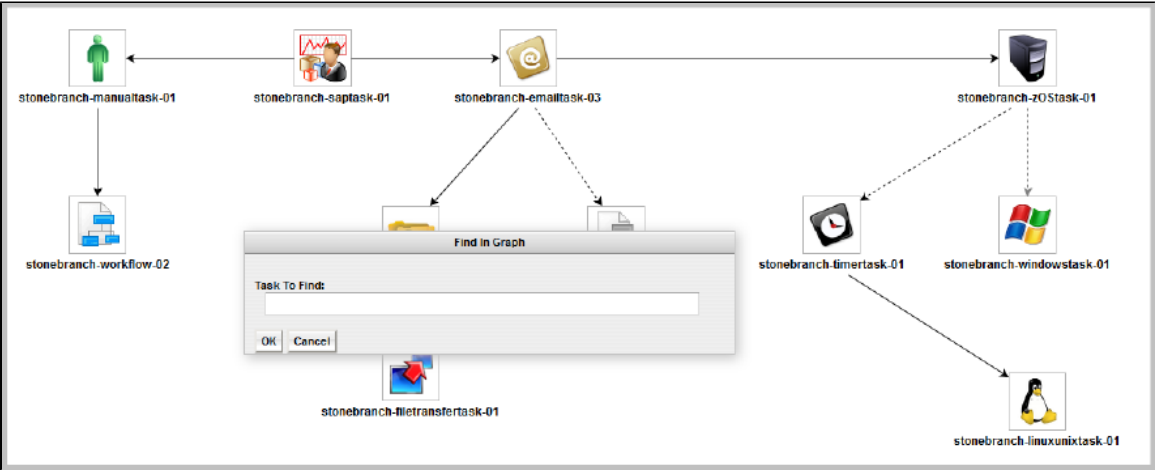
Deleting a Workflow

Step 1	From the Automation Center navigation pane, select Tasks > Workflow Tasks . The Workflow Tasks list displays.
---------------	--

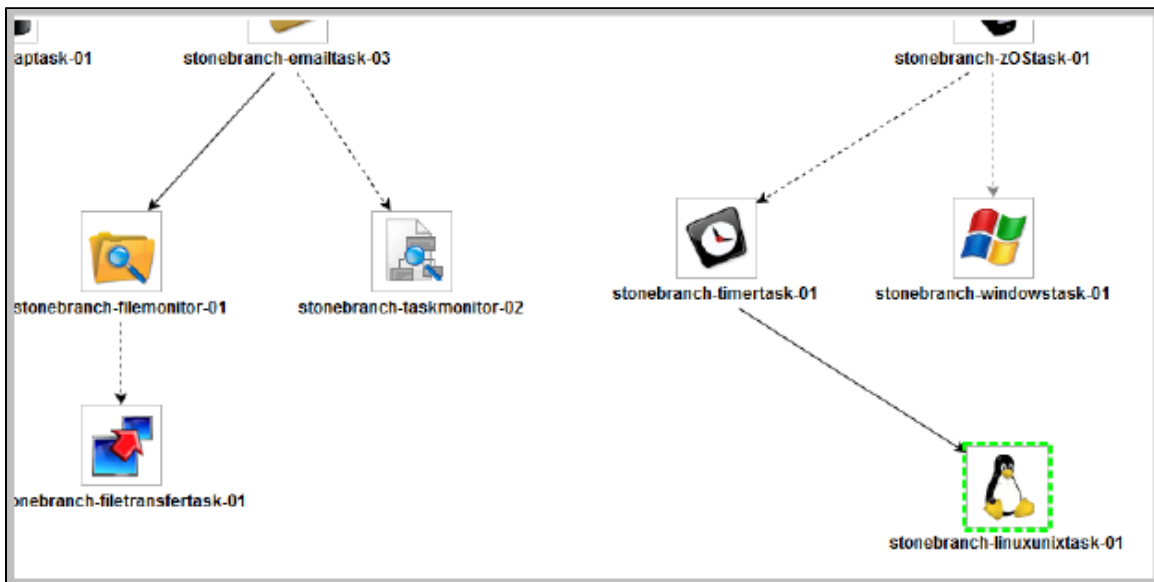
Step 2	Right-click the Workflow that you want to delete to display an Action menu . (If you want to delete multiple Workflows, use Ctrl+click to select each one and then right-click any of the selected Workflows.)
Step 3	Click Delete .

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	On the Workflow Tasks list, right-click a Workflow task to display an Action menu .
Step 2	Click Edit Workflow to display the Workflow Editor for that Workflow.
Step 3	Right-click in the Workflow Editor canvas. A pop-up menu displays.
	 <p>The screenshot shows a workflow graph with several tasks represented by icons: a person (manual), a person with a chart (saptask), an email (emailtask), a server (zostask), a folder (workflow), a monitor (taskmonitor), a clock (timertask), a Windows logo (windowstask), a file transfer (filetransfertask), and a penguin (linuxunixtask). A context menu is open over the 'stonebranch-taskmonitor-01' task, listing options: View/Edit, Paste, Launch Workflow, Find in Graph..., and Pan To Top.</p>
Step 4	Click Find in Graph... to display a Find in Graph dialog.
	 <p>The screenshot shows the same workflow graph as in Step 3. The 'Find in Graph' dialog box is open, featuring a text input field labeled 'Task To Find:', 'OK' and 'Cancel' buttons, and a search icon at the bottom.</p>

Step 5 Enter the name of the task you want to find and click OK. The Controller locates and displays the task within the Workflow.



Note

You also can locate a task in a Workflow by using the [Pan](#) tool.

Workflow Coordinates

If you use [Finding a Task...](#) (or [Pan mode](#)) to view different areas of a Workflow, the Controller preserves the coordinates of the last area that you viewed. If you leave the Workflow and return to it in the same browser session, the last viewed area of the Workflow displays.

To restore the displayed Workflow to its 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.

Viewing a Task Instance in a Workflow from the Activity Monitor or Task Instances List

From the Activity Monitor or Task Instances list, you can view the location and status of a task instance in a Workflow.

Step 1	Open the Activity Monitor or Task Instances List .
Step 2	Right-click a task instance to display the Action menu .
Step 3	Click View in Workflow . The parent Workflow of the selected task instance displays, with that task instance highlighted.



Note

You also can view any predecessor or successor task instance of a task instance in a Workflow from the [View Predecessors/Successors](#) pop-up dialog.

Inserting a Task in an Active 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	<p>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.</p> <p>The Controller 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.</p>

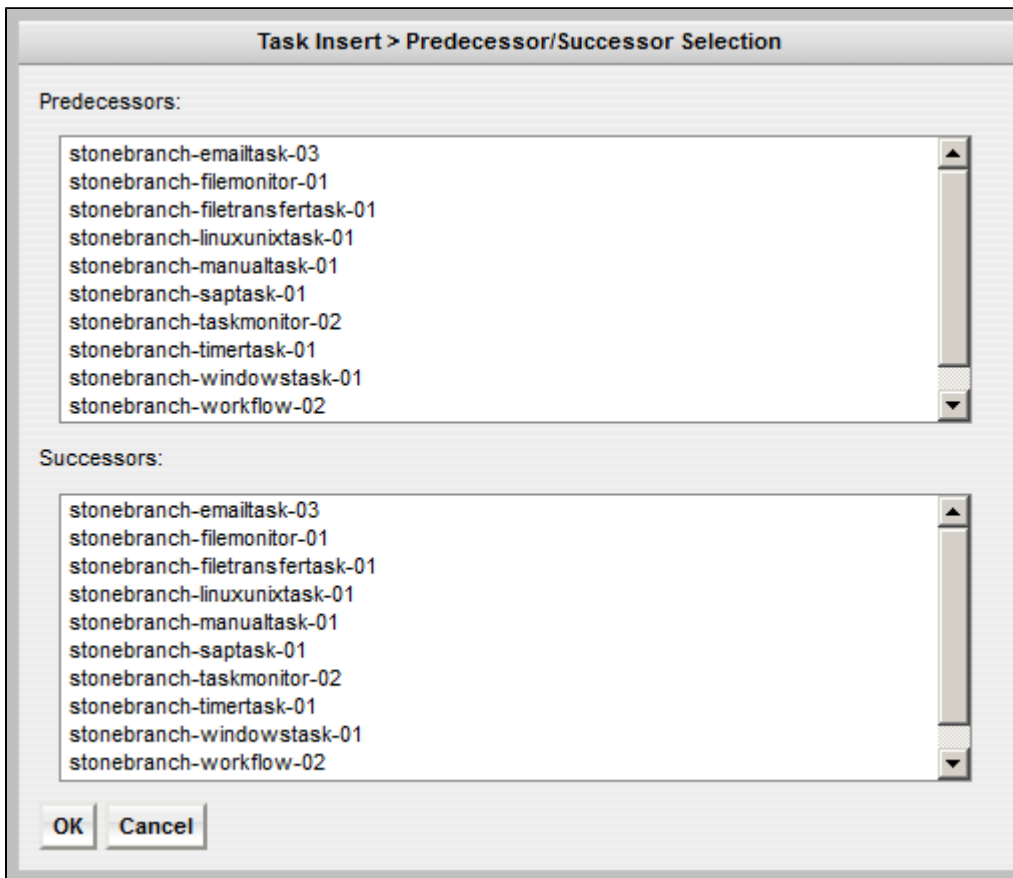
Insert Task as Successor

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	<p>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.</p> <p>The Controller 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.</p>

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**.

The Controller 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.

Viewing Task Predecessors / Successors

You can view all predecessors and successors of a task (or task instance) in a Workflow from:

- Activity Monitor
- Task Instances list
- Workflow Editor
- Workflow Monitor

Viewing Task Instance Predecessors / Successors from the Activity Monitor or Task Instances List

Step 1 Display the [Activity Monitor](#) or [Task Instances List](#).

Step 2 Right-click a task instance to display an **Action** menu.

Step 3 Click **View Predecessors/Successors** to display the View Predecessors/Successors pop-up dialog.

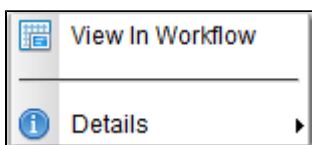
View Predecessors/Successors: Timer3			
1 Predecessor			
Task Instance	Condition	Satisfied	Inherited
Timer1	Success	Yes	No

2 Successors			
Task Instance	Condition	Satisfied	Inherited
Timer6	Success	Yes	No
Timer5	Success	Yes	No

The View Predecessors/Successors pop-up dialog provides separate lists of all predecessor and successor task instances of the selected task instance in the Workflow, and status information for each of those tasks instances.

Step 4 If you want to view the task instance Details of any predecessor or successor task instance, click that task instance.

Step 5 If you want to view a predecessor or successor task instance of the selected task instance within the Workflow, right-click that task instance to display an **Action** menu, and then click **View Workflow**.



The parent Workflow of the selected task instance displays, with the selected predecessor or successor task instance highlighted.

Viewing Task or Task Instance Predecessors / Successors from the Workflow Editor or Workflow Monitor



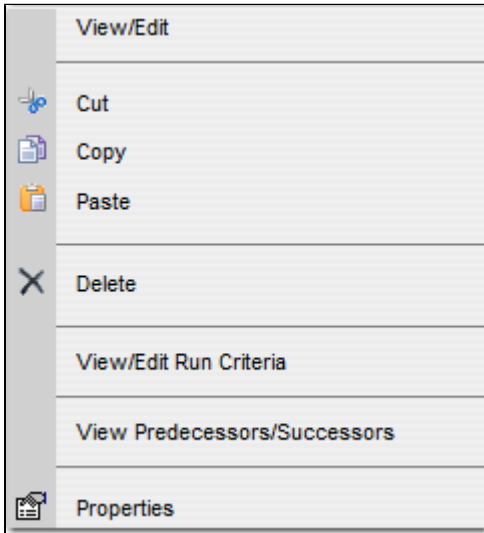
Note

To view more detailed task instance predecessor/successor information from the Workflow Monitor than is available via this procedure, see [Viewing Detailed Status of Task Instance Predecessors / Successors from Workflow Monitor](#), below.

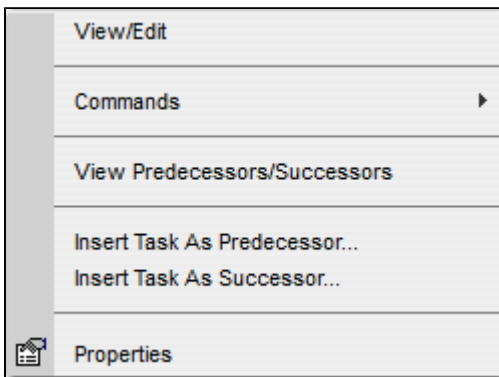
Step 1 Open the Workflow Editor or Workflow Monitor.

Step 2 Right-click a task to display an **Action menu**. (The menus are different for a task in a Workflow Editor and a task instance in a Workflow Monitor.)

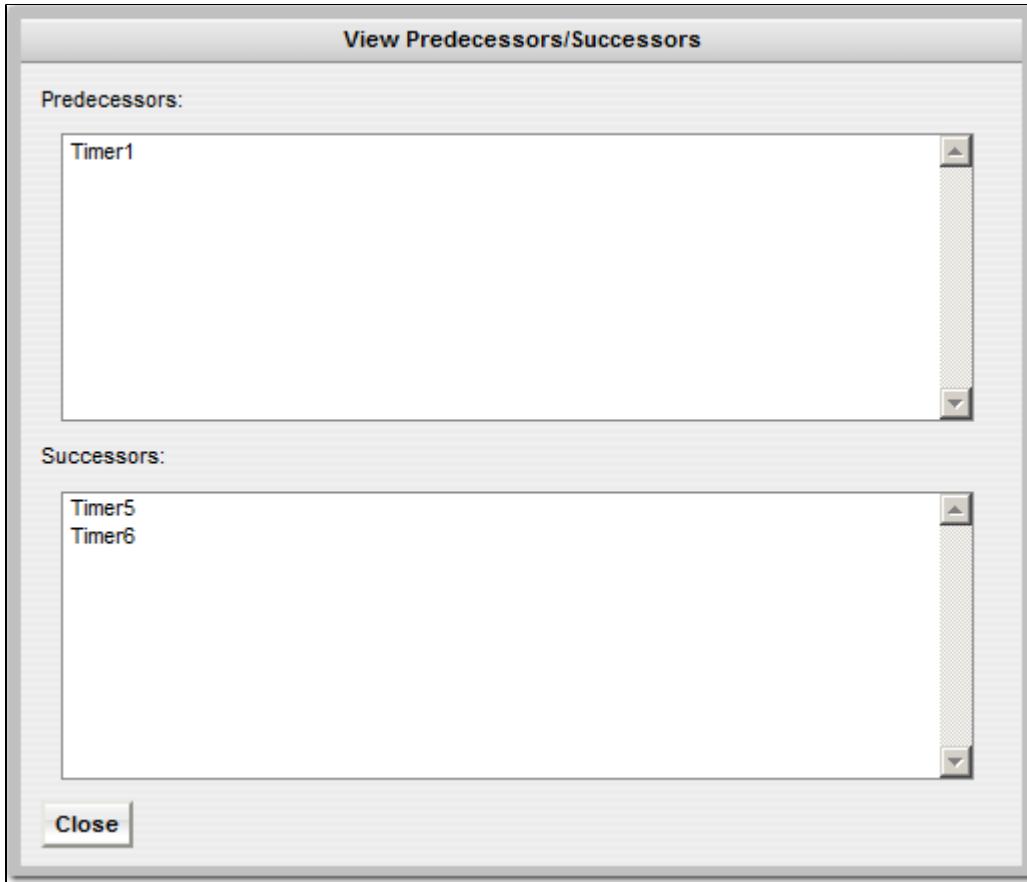
Workflow Editor Menu



Workflow Monitor Menu



Step 3 Right-click **View Predecessors/Successors** to display the View Predecessors/Successors pop-up.



The View Predecessors/Successors pop-up provides separate lists of all predecessor and successor task/task instances of the selected task/task instance in the Workflow.

- Click a task in either list to select and center it within the Workflow Editor / Monitor display.
- Double-click a task in either list to select and center it within the Workflow Editor / Monitor display and close the View Predecessors/Successors pop-up.

Viewing Detailed Status of Task Instance Predecessors / Successors from Workflow Monitor

Step 1	Open the Workflow Monitor.
Step 2	Double-click a task instance, or right-click and select View/Edit , to see its Task Instance Details.
Step 3	Right-click in the Task Instance Details to display an Action menu .

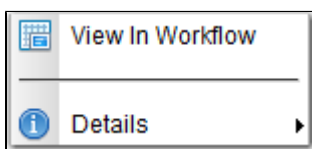
Step 4 Click **View Predecessors/Successors** to display a detailed view of the View Predecessors/Successors pop-up dialog.

View Predecessors/Successors: Timer3			
1 Predecessor			
Task Instance	Condition	Satisfied	Inherited
Timer1	Success	Yes	No
2 Successors			
Task Instance	Condition	Satisfied	Inherited
Timer6	Success	Yes	No
Timer5	Success	Yes	No

The View Predecessors/Successors pop-up dialog provides separate lists of all predecessor and successor task instances of the selected task instance in the Workflow, and status information for each of those tasks instances.

Step 5 If you want to view the task instance Details of any predecessor or successor task instance, click that task instance.

Step 6 If you want to view a predecessor or successor task instance of the selected task instance within the Workflow, right-click that task instance to display an **Action menu**, and then click **View Workflow**.



The parent Workflow of the selected task instance displays, with the selected predecessor or successor task instance highlighted.

Forecasting a Workflow for a Specific Date/Time

You can display forecast information for a Workflow for a specific date and time. This provides a view of how a Workflow will look based on the Run/Skip Criteria for that date and time.

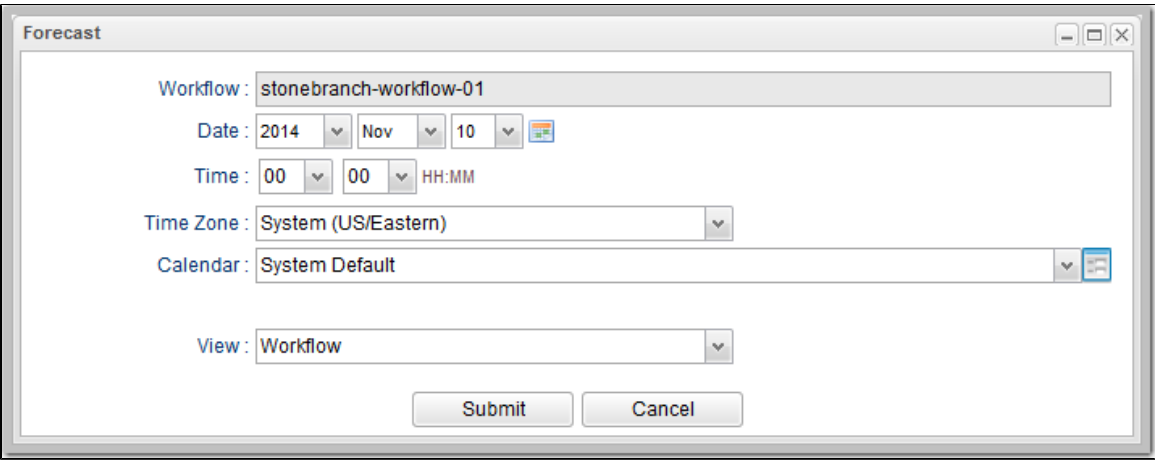
You can forecast a Workflow, without requiring a Trigger or Trigger forecast data, for any date/time within a span of five years in the past and 10 years in the future.

Step 1 From the **Automation Center** navigation pane, select **Tasks > All Tasks** or **Tasks > Workflow Tasks**.

Step 2 Display an **Action** menu either of three ways:

- Right-click the Workflow in the **All Tasks** or **Workflow Tasks** list.
- Open the Workflow and right-click in the Details.
- Open the Workflow, click the Triggers tab, and right-click a Trigger on the list.

Step 3 Click **Forecast...** to display a Forecast dialog.



Step 4 Select a Date, Time, Time Zone, and Calendar for when you want to forecast the Workflow (and all of its sub-Workflows, if any).

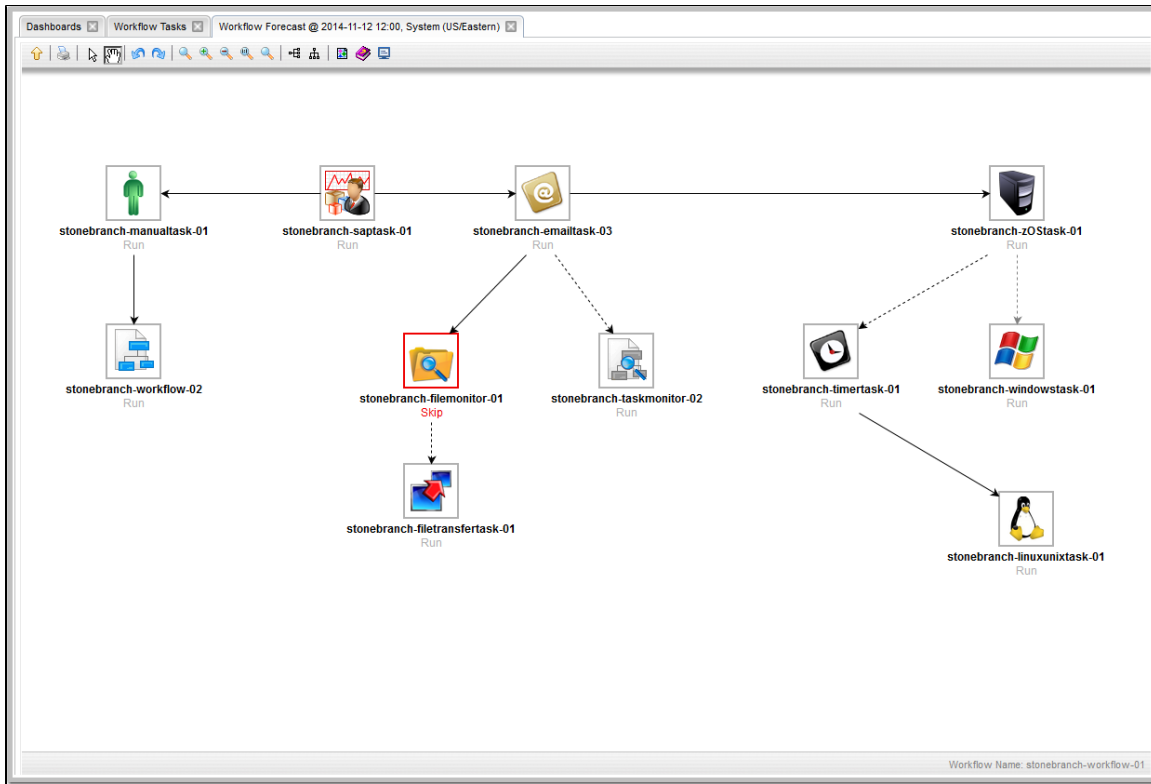
- If you selected Forecast from the **All Tasks** list, **Workflow Tasks** list, or Workflow Details, the default Calendar of the Workflow displays.
- If you selected Forecast from Triggers tab list, the default Calendar of the selected Trigger displays.

Step 5 Select a **View** in which you want to display the Forecast: Workflow or List.

Step 6

Click **Submit** to displays the Forecast in the selected View.

Workflow View



In the Workflow View, each task has a Run Criteria evaluation label:

- Run - Task border is gray and solid.
- Skip - Task border is red and solid.
- Not Evaluated - Task border is gray and dotted. (Run/skip criteria for this task is dependent on variable resolution; could not be evaluated.)



Note

You can choose to [show or hide skipped tasks](#) in the Workflow View of the Forecast just as you can on a Workflow Monitor.

The Workflow View lets you navigate into sub-Workflows (by double-clicking them) and back to the parent Workflow (via the [Go to Parent](#) icon).

List View

Task ^ 4	Task Type ^ 3	Workflow ^ 2	Launch Time ^ 1	End Time	Run Criteria Evaluation
stonebranch-workflow-01	Workflow		2014-11-12 12:00:00 -0500	2014-11-12 12:00:00 -0500	Run
stonebranch-timertask-01	Timer	stonebranch-workflow-01	2014-11-12 12:00:00 -0500	2014-11-12 12:00:00 -0500	Run
stonebranch-windowstask-01	Windows	stonebranch-workflow-01	2014-11-12 12:00:00 -0500	2014-11-12 12:00:00 -0500	Run
stonebranch-linuxunixtask-01	Linux/Unix	stonebranch-workflow-01	2014-11-12 12:00:00 -0500	2014-11-12 12:00:00 -0500	Run
stonebranch-zOSTask-01	z/OS	stonebranch-workflow-01	2014-11-12 12:00:00 -0500	2014-11-12 12:00:00 -0500	Run
stonebranch-filemonitor-01	File Monitor	stonebranch-workflow-01	2014-11-12 12:00:00 -0500	2014-11-12 12:00:00 -0500	Skip
stonebranch-filetransfertask-01	File Transfer	stonebranch-workflow-01	2014-11-12 12:00:00 -0500	2014-11-12 12:00:00 -0500	Run
stonebranch-saptask-01	SAP	stonebranch-workflow-01	2014-11-12 12:00:00 -0500	2014-11-12 12:00:00 -0500	Run
stonebranch-manualtask-01	Manual	stonebranch-workflow-01	2014-11-12 12:04:00 -0500	2014-11-12 12:04:00 -0500	Run
stonebranch-emailtask-03	Email	stonebranch-workflow-01	2014-11-12 12:04:00 -0500	2014-11-12 12:04:00 -0500	Run
stonebranch-taskmonitor-02	Task Monitor	stonebranch-workflow-01	2014-11-12 12:04:01 -0500	2014-11-12 12:04:01 -0500	Run
stonebranch-workflow-02	Workflow	stonebranch-workflow-01	2014-11-12 12:04:33 -0500	2014-11-12 12:04:33 -0500	Run

The Forecast View displays forecast data for the Workflow and all of its tasks, including all sub-Workflows and their tasks.

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 Instances
- Cancelling a Task Instances
- Force Finishing a Task Instances
- Putting a Task Instances on Hold
- Releasing a Task Instances from Hold
- Skipping a Task Instances
- Showing or Hiding Skipped Task Instances
- Unskipping a Task Instances
- Clearing All Predecessor Dependencies for a Task Instance
- Marking a Task Instances Dependency as Satisfied

Monitoring Task Activity

Opswise Controller lets you perform the following actions related to the monitoring of tasks and Workflows:

- [Monitoring Activity from the Activity Monitor](#)
- [Monitoring Activity from the Task Instances List](#)
- [Monitoring Activity History](#)
- [Monitoring Workflows](#)
- [Viewing Task Instances for a Specific Task](#)
- [Displaying Task Instance Status](#)
- [Retrieving Output](#)

Monitoring Activity from the Activity Monitor

- Overview
- Accessing the Activity Monitor
- Activity Monitor Column Descriptions
- Activity Monitor Task Bar
 - Time Constraint
 - Quick Filter
 - Custom Filter
- Activity Monitor Display Task Bar
 - Settings
 - Start and Stop Activity Monitor
 - Page Navigation Buttons
 - Page Numbers
- Displaying Task Instance Details
- Issuing Commands Against Task Instances

Overview

The Activity Monitor is the Opwise Controller central console of activity, a real-time display of task instance status. It displays a list of task instances contained in the Opwise Controller [All Task Instances](#) table (`ops_exec`), as controlled by your selections in the [Activity Monitor Selections](#) task bar at the top of the list.

A task bar below the list allows you to control displays settings for the selected task instances.

A task instance is the "run" version of a task. Each time a task runs, the Controller creates a task instance and monitors its activity on the Activity Monitor. Each task instance is a separate record.

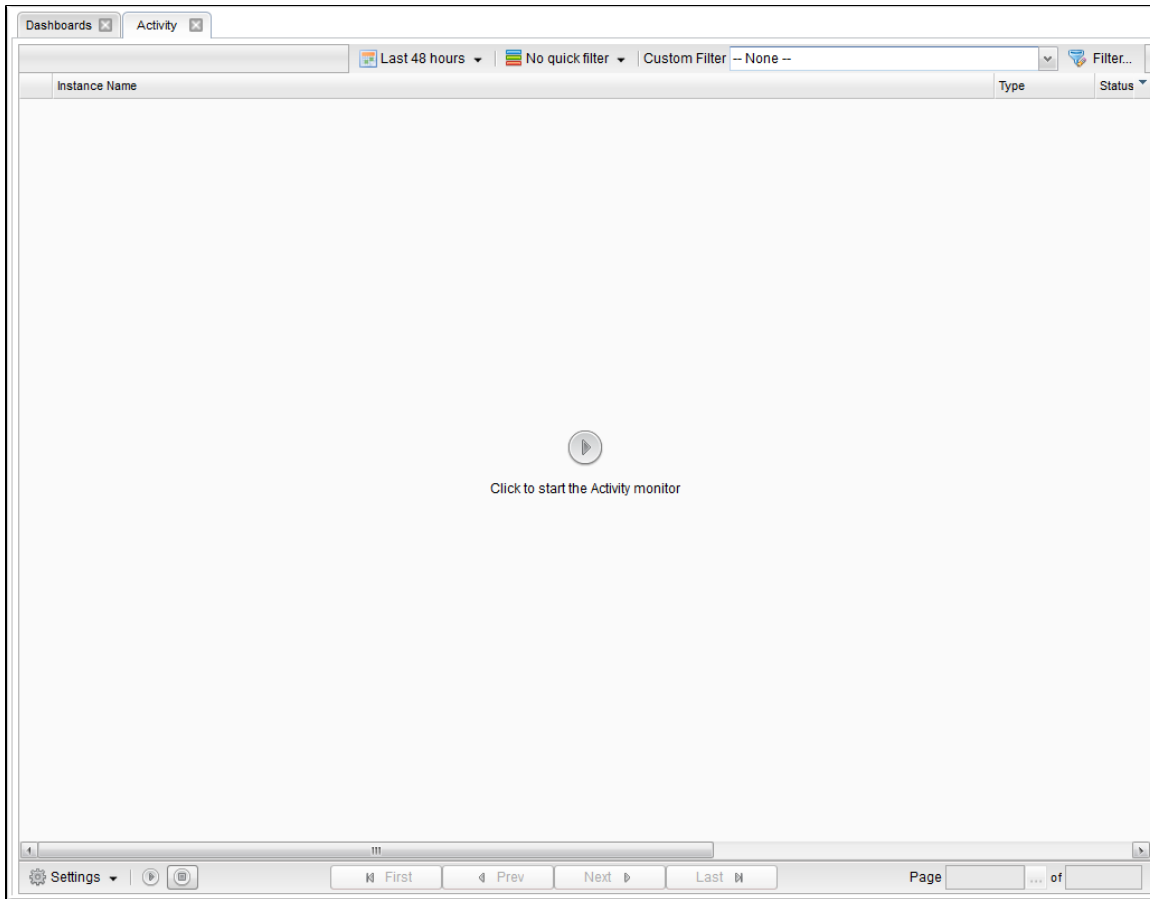
The Activity Monitor allows you to [issue commands against task instances](#). You can also issue commands from the [Task Instances](#) list (and the [Task Instances list for a specific task](#)). In cases where the task Details did not instruct the Controller to retrieve output automatically, you can [retrieve output manually](#) from any completed task.

Accessing the Activity Monitor

From the [Automation Center](#) navigation pane, select **Task Instances > Activity**. The Activity Monitor displays task status information based on the settings in the [Activity Monitor Selections Task Bar](#).

When you first log in to the Controller, the Activity Monitor does not automatically monitor Controller activity, since the [Activity Monitor Automatically](#) user preference value, by default, is set to **No**. You can change this value at any time.

The following is a sample Activity Monitor that is not monitoring activity.



You can manually start the Activity Monitor at any time by clicking the Start button in the center of the screen. You also can manually start or stop the Activity Monitor at any time from the Activity Monitor Display task bar. However, on your next login to the Controller, the Activity Monitor Automatically value will determine whether or not the Activity Monitor automatically starts monitoring the Controller.

The following is a sample Activity Monitor that is monitoring activity.

Instance Name	Type	Status	Invoked By	Start Time	End Time	Updated
lecu-wkfl-sleep	Workflow	Success	Manually Launched	2014-09-02 12:57:55 -0400	2014-09-02 12:58:36 -0400	2014-09-02 12:58:36 -0400
sleep 10	Timer	Success	Workflow: lecu-wkfl-sleep	2014-09-02 12:58:26 -0400	2014-09-02 12:58:36 -0400	2014-09-02 12:58:36 -0400
Sleep 30	Timer	Success	Workflow: lecu-wkfl-sleep	2014-09-02 12:57:56 -0400	2014-09-02 12:58:26 -0400	2014-09-02 12:58:26 -0400
Sleep 0	Timer	Success	Workflow: lecu-wkfl-sleep	2014-09-02 12:57:56 -0400	2014-09-02 12:57:56 -0400	2014-09-02 12:57:56 -0400
Sleep 60	Timer	Skipped	Workflow: lecu-wkfl-sleep		2014-09-02 12:57:55 -0400	2014-09-02 12:57:55 -0400
Sleep 60	Timer	Skipped	Workflow: lecu-wkfl-sleep		2014-09-02 12:57:55 -0400	2014-09-02 12:57:55 -0400
Sleep 60	Timer	Skipped	Workflow: lecu-wkfl-sleep		2014-09-02 12:57:55 -0400	2014-09-02 12:57:55 -0400
Sleep 30	Timer	Skipped	Workflow: lecu-wkfl-sleep		2014-09-02 12:57:55 -0400	2014-09-02 12:57:55 -0400
Sleep 30	Timer	Skipped	Workflow: lecu-wkfl-sleep		2014-09-02 12:57:55 -0400	2014-09-02 12:57:55 -0400
win-exit-code	Windows	Failed	Manually Launched	2014-09-02 12:44:03 -0400	2014-09-02 12:44:03 -0400	2014-09-02 12:44:03 -0400
zos-workflow-regression-test	Workflow	Success	Manually Launched	2014-09-02 11:52:47 -0400	2014-09-02 12:10:30 -0400	2014-09-02 12:10:30 -0400
zos-workflow-simple-load-test-01	Workflow	Success	Workflow: zos-workflow-regression-test	2014-09-02 11:56:39 -0400	2014-09-02 12:10:30 -0400	2014-09-02 12:10:30 -0400
zos-task-load-simple-01	z/OS	Success	Workflow: zos-workflow-simple-load-test-01	2014-09-02 12:10:29 -0400	2014-09-02 12:10:29 -0400	2014-09-02 12:10:30 -0400
zos-task-load-simple-01	z/OS	Success	Workflow: zos-workflow-simple-load-test-01	2014-09-02 12:10:25 -0400	2014-09-02 12:10:26 -0400	2014-09-02 12:10:27 -0400
zos-task-load-simple-01	z/OS	Success	Workflow: zos-workflow-simple-load-test-01	2014-09-02 12:10:23 -0400	2014-09-02 12:10:23 -0400	2014-09-02 12:10:24 -0400
zos-task-load-simple-01	z/OS	Success	Workflow: zos-workflow-simple-load-test-01	2014-09-02 12:10:20 -0400	2014-09-02 12:10:20 -0400	2014-09-02 12:10:21 -0400
zos-workflow-simple-load-test-02	Workflow	Success	Workflow: zos-workflow-regression-test	2014-09-02 11:56:39 -0400	2014-09-02 12:10:19 -0400	2014-09-02 12:10:19 -0400
zos-task-load-simple-02	z/OS	Success	Workflow: zos-workflow-simple-load-test-02	2014-09-02 12:10:17 -0400	2014-09-02 12:10:17 -0400	2014-09-02 12:10:19 -0400
zos-task-load-simple-01	z/OS	Success	Workflow: zos-workflow-simple-load-test-01	2014-09-02 12:10:15 -0400	2014-09-02 12:10:15 -0400	2014-09-02 12:10:19 -0400
zos-task-load-simple-02	z/OS	Success	Workflow: zos-workflow-simple-load-test-02	2014-09-02 12:10:12 -0400	2014-09-02 12:10:12 -0400	2014-09-02 12:10:15 -0400
zos-task-load-simple-01	z/OS	Success	Workflow: zos-workflow-simple-load-test-01	2014-09-02 12:10:11 -0400	2014-09-02 12:10:11 -0400	2014-09-02 12:10:14 -0400
zos-workflow-simple-load-test-03	Workflow	Success	Workflow: zos-workflow-regression-test	2014-09-02 11:56:39 -0400	2014-09-02 12:10:13 -0400	2014-09-02 12:10:13 -0400
zos-task-load-simple-03	z/OS	Success	Workflow: zos-workflow-simple-load-test-03	2014-09-02 12:10:09 -0400	2014-09-02 12:10:09 -0400	2014-09-02 12:10:13 -0400
zos-task-load-simple-02	z/OS	Success	Workflow: zos-workflow-simple-load-test-02	2014-09-02 12:10:07 -0400	2014-09-02 12:10:08 -0400	2014-09-02 12:10:11 -0400
zos-task-load-simple-01	z/OS	Success	Workflow: zos-workflow-simple-load-test-01	2014-09-02 12:10:06 -0400	2014-09-02 12:10:06 -0400	2014-09-02 12:10:10 -0400



Note

Once you have displayed the Activity Monitor, it remains open throughout your Controller session unless you manually close it by clicking the x icon in the Activity Monitor tab at the top of the page. (The same is true for the [Dashboards](#).)

If you click a page in the [Navigator](#) while viewing the Activity Monitor or Dashboards, a new tab for that page will open. The Activity Monitor and Dashboards tabs remain at the top of the screen if you navigate from them, providing for a quick return to those pages.

Activity Monitor Column Descriptions

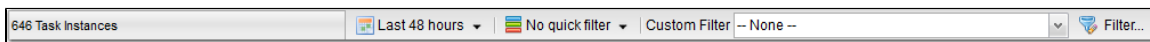
The following table describes the default columns of information displayed on the Activity Monitor.

Column Name	Description
Instance Name	Name of this task instance.
Type	Type of task instance.
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.
Updated	Date and time that this task instance was last updated.

Activity Monitor Task Bar

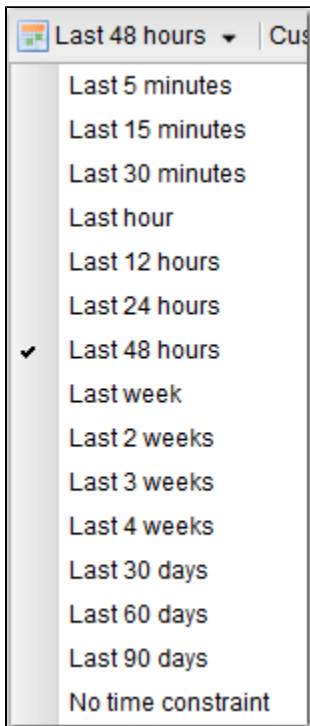
An Activity Monitor task bar displays across the top of the [Activity Monitor](#), which allows you to select which task instances display on the Activity Monitor.



Time Constraint

The Time Constraint drop-down list in the Activity Monitor Selections task bar allows you to select a time frame for which you want task instances to display on the Activity Monitor.

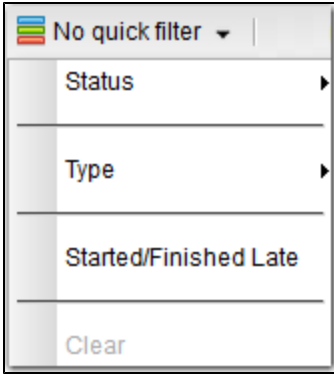
The default time constraint, as specified by the [Activity Time Constraint](#) user preference, is 48 hours.



Quick Filter

The Quick Filter menu in the Activity Monitor Selections task bar allows you to filter the list with simple, pre-defined criteria so that only task instances matching that criteria within the selected [Time Constraint](#) and any selected [Custom Filter](#) display on the Activity Monitor.

To display the Quick Filter menu, click the Quick Filter button on the task bar which, by default, is identified as **No quick filter**.

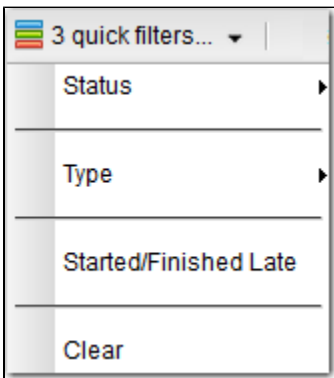


The Quick Filter menu allows you to select any combination of the following filters to the list:

- **Status:** Individual [task instance status](#) and the following composite statuses:
 - Active = All task instance statuses except Skipped (180), Finished (190), Success (200).
 - Blocked = Waiting (10), Held (20), Exclusive Wait (23), Resource Wait (30), Execution Wait (33), and Action Required (60) task instance statuses.
 - Completed = Skipped (180), Finished (190), and Success (200) task instance statuses.
 - Problem = Undeliverable (35), Running/Problems (81), Cancel Pending (99), In Doubt (110), Start Failure (120), Confirmation Required (125), Cancelled (130), and Failed (140) task instance statuses.
- **Type:** Individual [task type](#).
- **Started/Finished Late:** Task instance started and/or finish late according to the Start / Finish Late times specified in the task Details.
- **Clear =** Clears all Quick Filters from the list. (You can clear individual Quick Filters by deselecting them on the menu.)

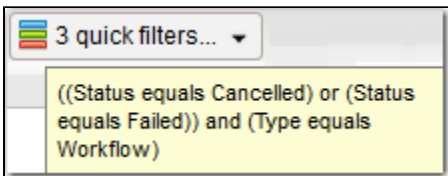
Note
 You can change the names of the composite statuses Quick Filters (Active, Blocked, Completed, and Problem), and the task instance statuses included in each one, by editing the [System Default Activity Quick Filters](#) Opswise Controller system property.

You must select each Quick Filter individually; you cannot use Ctrl-click to select multiple Quick Filters. The Quick Filter button provides a count of how many Quick Filters have been applied to the list. For example, the following Quick Filter button shows that three separate quick filters have been applied.



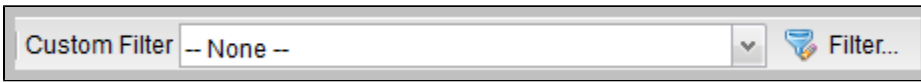
The **Clear** button now is enabled, allowing you to remove all Quick Filters from the list in one click.

To see which Quick Filters have been applied to the list, hover your cursor over the Quick Filter button.



Custom Filter

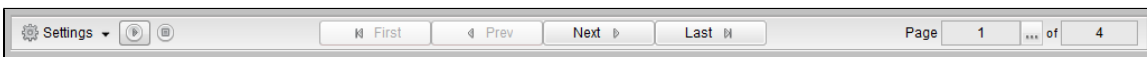
The **Custom Filter** field and **Filter...** button in the Activity Monitor Selections task bar allows you to filter the list with complex, user-defined criteria so that only task instances matching that specified criteria within the selected **Time Constraint** and any selected **Quick Filter** display on the Activity Monitor.



See [Filters](#) for detailed information on applying and saving filters.

Activity Monitor Display Task Bar

An Activity Monitor Display task bar displays across the bottom of the **Activity Monitor**, which allows you to select which task instances on the Activity Monitor to view and how often the data for those task instances is refreshed.



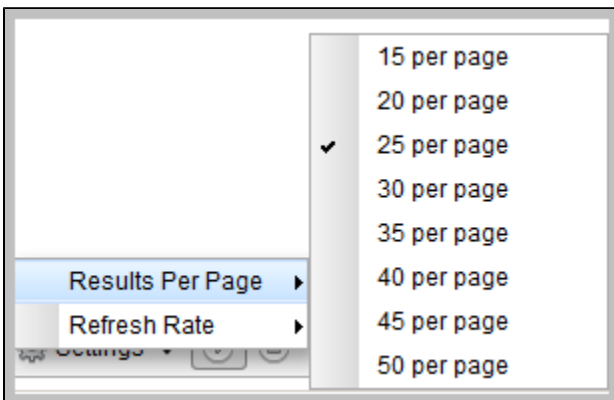
Settings

The Settings drop-down list provides the following selections:

- Results Per Page
- Refresh Rate

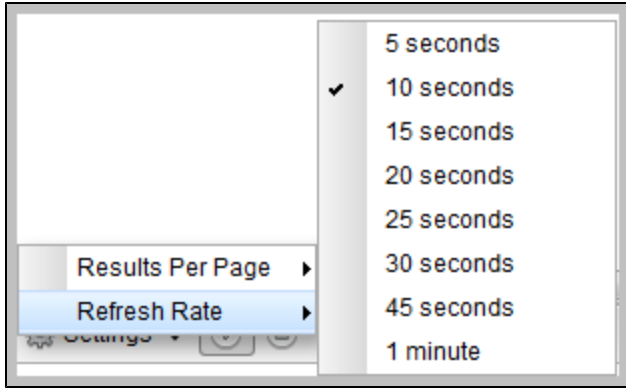
Results Per Page

Hover your cursor **Results Per Page** on the Settings drop-down list to select the number of task instances to be displayed on a page (default is 25).



Refresh Rate

Hover your cursor over **Refresh Rate** on the Settings drop-down list to select a refresh rate for the data displayed on the Activity Monitor (default is 10 seconds).



Start and Stop Activity Monitor

Start and Stop buttons on the Activity Monitor allow you to start and stop the monitoring of activity on the Controller.

The [Activity Monitor Automatically](#) value specifies whether or not the Activity Monitor automatically is monitoring Controller activity.

You can override that value by electing either of the following, as appropriate:

	Starts monitoring Controller activity. The Activity Monitor displays task instances according to the current Time Constraint setting and any applied Filters .
	Stops monitoring Controller activity. All task instances are removed from the Activity Monitor display.

Page Navigation Buttons

Page navigation buttons allow you to scroll through the pages of task instances that are displayed according to the current Activity Monitor settings.

	Displays the first page of the list.
	Displays the previous page of the list.
	Displays the next page of the list.
	Displays the last page of the list.



Note

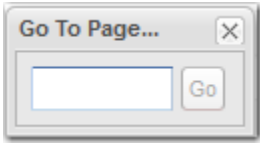
You also can specify a specific page number to go to (see [Page Numbering](#), below).

Page Numbers

Page Numbers buttons in the task bar identify the number of the currently displayed page and the total number of pages.



You can go to a specific page number by clicking the ellipse (...) icon that displays after the current page number. A Go To Page pop-up displays.



Enter a page number and click the **Go** button to navigate to that page.

Displaying Task Instance Details

To view details about any task instance displayed on the [Activity Monitor](#), click the **Details** icon next to the **Instance Name** of that task instance.

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](#).

Monitoring Activity from the Task Instances List

- Overview
- Displaying the Task Instances List
 - Task Instances List Column Descriptions
- Task Instances List Task Bar
 - Time Constraint
 - Filtering
- Accessing Task Instance Details
- Issuing Commands Against Task Instances

Overview

The Task Instances list displays the same task instance information as the [Activity Monitor](#), but only for task instances for which there has been a status change or a modification to the task instance record.

Also, unlike the Activity Monitor, the Task Instances list is not automatically refreshed.

You also can monitor activity for a specific task by displaying [task-specific Task Instances Details](#).

Displaying the Task Instances List

From the [Automation Center](#) navigation pane, select **Task Instances > All Task Instances**. The Task Instances list displays.

Instance Name	Type	Status	Invoked By	Start Time	End Time	Updated
ecu-uag-d03237rc	z/OS	Success	Manually Launched	2014-08-21 21:28:09 -0400	2014-08-21 21:28:10 -0400	2014-08-21 21:28:11 -0400
zos-task-run-simple	z/OS	Success	Workflow: zos-workflow-regression-test	2014-08-21 21:29:39 -0400	2014-08-21 21:29:39 -0400	2014-08-21 21:29:41 -0400
zos-task-security-auth-01	z/OS	Success	Workflow: zos-workflow-user-authentication	2014-08-21 21:29:42 -0400	2014-08-21 21:29:42 -0400	2014-08-21 21:29:44 -0400
zos-task-security-auth-02	z/OS	Success	Workflow: zos-workflow-user-authentication	2014-08-21 21:29:42 -0400	2014-08-21 21:29:43 -0400	2014-08-21 21:29:45 -0400
zos-task-security-auth-03	z/OS	Success	Workflow: zos-workflow-user-authentication	2014-08-21 21:29:43 -0400	2014-08-21 21:29:44 -0400	2014-08-21 21:29:47 -0400
Sleep 0	Timer	Success	Workflow: zos-workflow-user-authentication	2014-08-21 21:29:47 -0400	2014-08-21 21:29:47 -0400	2014-08-21 21:29:47 -0400
zos-task-security-auth-13	z/OS	Finished	Workflow: zos-workflow-user-authentication		2014-08-21 21:29:48 -0400	2014-08-21 21:29:48 -0400
zos-task-security-auth-10	z/OS	Finished	Workflow: zos-workflow-user-authentication		2014-08-21 21:29:48 -0400	2014-08-21 21:29:48 -0400
zos-task-security-auth-12	z/OS	Finished	Workflow: zos-workflow-user-authentication		2014-08-21 21:29:48 -0400	2014-08-21 21:29:48 -0400
zos-task-security-auth-11	z/OS	Finished	Workflow: zos-workflow-user-authentication		2014-08-21 21:29:48 -0400	2014-08-21 21:29:49 -0400
zos-task-failure	z/OS	Skipped	Workflow: zos-workflow-user-authentication		2014-08-21 21:29:48 -0400	2014-08-21 21:29:49 -0400
Sleep 0	Timer	Success	Workflow: zos-workflow-user-authentication	2014-08-21 21:29:49 -0400	2014-08-21 21:29:49 -0400	2014-08-21 21:29:49 -0400
zos-workflow-user-authenti...	Work...	Success	Workflow: zos-workflow-regression-test	2014-08-21 21:29:41 -0400	2014-08-21 21:29:49 -0400	2014-08-21 21:29:49 -0400
zos-step-action-setup	Timer	Success	Workflow: zos-workflow-step-actions	2014-08-21 21:29:49 -0400	2014-08-21 21:29:50 -0400	2014-08-21 21:29:50 -0400

Task Instances List Column Descriptions

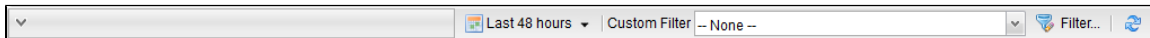
The following table describes the default columns of information displayed on the Task Instances list.

Column	Description
Instance Name	Name of this 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.
Start Time	Date and time the task instance started.
End Time	Date and time the task instance ended.
Status	Current status of the task instance.
Type	Type of task instance.
Updated	Date and time this task instance ended or was last updated.

Task Instances List Task Bar

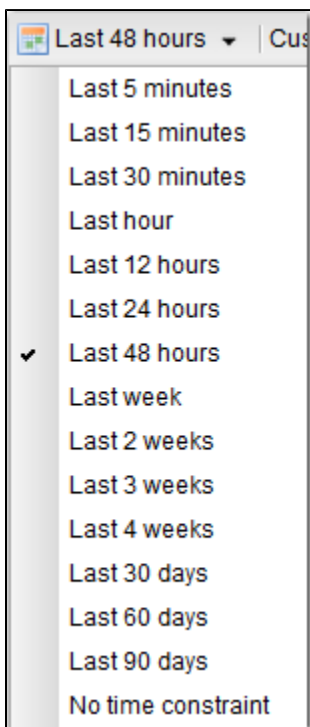
A Task Instances List task bar displays across the top of the [Task Instances list](#), which allows you to select which task instances display on the list.



Time Constraint

The Time Constraint drop-down list in the Task Instance Selections task bar allows you to select a time frame for which you want task instances to display on the list.

The default time constraint, as specified by the [Task Instances Time Constraint](#) user preference, is 48 hours.



Filtering

The **Custom Filter** field and **Filter...** button in the Task Instance Selections task bar allows you to filter the list so that only task instances

matching the specified criteria within the selected time constraint display on the list.



See [Filters](#) for detailed information on applying and saving filters.

Accessing Task Instance Details

To display Details for a task instance, either:

- Click the **Instance Name** of the task instance to display the Details below the list.
- Click the **Details** icon next to the **Instance Name** of the task instance to display a Details pop-up.

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](#).

Monitoring Activity History

- Overview
- Displaying the History List
- History List Task Bar
 - Time Constraint
 - Filtering
- Displaying History Details

Overview

The History list 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.

The list 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 list, 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 Monitor](#) and/or [Task Instances list](#)

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 List

Step 1 From the Automation Center navigation pane, select **Task Instances > History**. The History list displays.

Instance Name	Instance Id	Type	Status	Start Time	End Time	Duration	Waited for Resources	Waited for Exclusive	Updated By	Updated
zos-task-load-simple-02	93	z/OS	Success	2014-08-21 21:35:04 -0400	2014-08-21 21:35:04 -0400	0 Seconds	No	No	ops.system	2014-08-21 21:35:08 -0400
zos-task-load-simple-05	31	z/OS	Success	2014-08-21 21:42:50 -0400	2014-08-21 21:42:51 -0400	0 Seconds	No	No	ops.system	2014-08-21 21:42:54 -0400
zos-task-load-simple-04	25	z/OS	Success	2014-08-21 21:37:06 -0400	2014-08-21 21:37:07 -0400	0 Seconds	No	No	ops.system	2014-08-21 21:37:10 -0400
zos-task-security-auth-11	1	z/OS	Start Failure				No	No	ops.system	2014-08-21 21:29:48 -0400
zos-task-load-simple-01	66	z/OS	Success	2014-08-21 21:35:42 -0400	2014-08-21 21:35:42 -0400	0 Seconds	No	No	ops.system	2014-08-21 21:35:45 -0400
zos-task-load-simple-04	14	z/OS	Success	2014-08-21 21:45:20 -0400	2014-08-21 21:45:20 -0400	0 Seconds	No	No	ops.system	2014-08-21 21:45:23 -0400
zos-task-rerun-auto-02	1	z/OS	Step Restarted				No	No	ops.system	2014-08-21 21:30:24 -0400
zos-task-load-simple-02	33	z/OS	Success	2014-08-21 21:34:43 -0400	2014-08-21 21:34:45 -0400	1 Seconds	No	No	ops.system	2014-08-21 21:34:48 -0400
zos-task-load-simple-03	19	z/OS	Success	2014-08-21 21:34:47 -0400	2014-08-21 21:34:48 -0400	0 Seconds	No	No	ops.system	2014-08-21 21:34:50 -0400
zos-task-load-simple-04	69	z/OS	Success	2014-08-21 21:40:19 -0400	2014-08-21 21:40:20 -0400	0 Seconds	No	No	ops.system	2014-08-21 21:40:23 -0400
zos-task-load-simple-05	95	z/OS	Success	2014-08-21 21:41:15 -0400	2014-08-21 21:41:16 -0400	0 Seconds	No	No	ops.system	2014-08-21 21:41:19 -0400
zos-task-load-simple-02	10	z/OS	Success	2014-08-21 21:35:43 -0400	2014-08-21 21:35:44 -0400	0 Seconds	No	No	ops.system	2014-08-21 21:35:47 -0400
zos-task-steppcond-abend...	1	z/OS	Failed	2014-08-21 21:30:30 -0400	2014-08-21 21:30:43 -0400		No	No	ops.system	2014-08-21 21:30:43 -0400
zos-task-load-simple-01	82	z/OS	Success	2014-08-21 21:34:13 -0400	2014-08-21 21:34:13 -0400	0 Seconds	No	No	ops.system	2014-08-21 21:34:17 -0400
zos-task-load-simple-05	2	z/OS	Success	2014-08-21 21:36:09 -0400	2014-08-21 21:36:09 -0400	0 Seconds	No	No	ops.system	2014-08-21 21:36:13 -0400
zos-task-load-simple-02	54	z/OS	Success	2014-08-21 21:42:29 -0400	2014-08-21 21:42:34 -0400	5 Seconds	No	No	ops.system	2014-08-21 21:42:37 -0400
zos-task-steppcond-abend...	1	z/OS	Finished	2014-08-21 21:31:26 -0400	2014-08-21 21:31:36 -0400	9 Seconds	No	No	ops.system	2014-08-21 21:31:36 -0400
zos-task-load-simple-02	28	z/OS	Success	2014-08-21 21:36:23 -0400	2014-08-21 21:36:24 -0400	1 Seconds	No	No	ops.system	2014-08-21 21:36:28 -0400
zos-task-load-simple-03	51	z/OS	Success	2014-08-21 21:47:02 -0400	2014-08-21 21:47:03 -0400	0 Seconds	No	No	ops.system	2014-08-21 21:47:06 -0400
zos-task-rerun-auto-02	1	z/OS	Step Restarted				No	No	ops.system	2014-08-21 21:30:47 -0400
zos-task-load-simple-02	46	z/OS	Success	2014-08-21 21:41:35 -0400	2014-08-21 21:41:35 -0400	0 Seconds	No	No	ops.system	2014-08-21 21:41:38 -0400
zos-task-load-simple-04	17	z/OS	Success	2014-08-21 21:41:38 -0400	2014-08-21 21:41:45 -0400	5 Seconds	No	No	ops.system	2014-08-21 21:41:47 -0400

Step 2 You can modify the display by filtering, sorting, adding, and removing columns (see [Record Lists](#)).

History List Task Bar

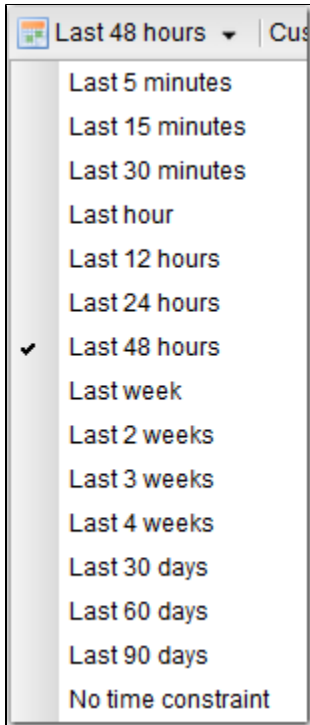
A History List task bar displays across the top of the [History list](#), which allows you to select which task instances display on the list.



Time Constraint

The Time Constraint drop-down list in the History List Selections task bar allows you to select a time frame for which you want task instances to display on the list.

The default time constraint, as specified by the [History Time Constraint](#) user preference, is 48 hours.



Filtering

The **Custom Filter** field and **Filter...** button in the History List Selections task bar allows you to filter the list so that only task instances matching the specified criteria within the selected time constraint display on the list.



See [Filters](#) for detailed information on applying and saving filters.

Displaying History Details

To display execution details about any task instance on the History list, either:

- Click anywhere in the task instance row to display the Details for that task instance below the list.
- Click the icon next to any **Instance Name** to display a pop-up version of the Details.

History Details: stonebranch-saptask-01

History

Details

Instance Name: stonebranch-saptask-01

Task Description:

Task Instance Name: stonebranch-saptask-01 Instance Id: 6

Task Name: stonebranch-saptask-01 Invoked By: Workflow: stonebranch-workflow-01

Type: SAP agentsim1 Trigger Name:

Status: Finished Agent Name: stonebranch-agent-00

Exit Code: 0

Start Time:

End Time: 2014-07-09 13:03:05 -0400

Waited for Resources:

Waited for Exclusive:

Duration:

Status Description: Agent is not Active -> State was forced from UNDELIVERABLE to FINISHED

Member of Business Services:

Delete Close

Delete Close

For a description of the fields displayed on the History Details, see the Task Instance Details field descriptions for that task type.

Monitoring Workflows

- Overview
- Monitoring a Running Workflow
- Workflow Monitor Display Mode
 - Task Name
 - Task Type
 - Task Status
 - Displaying Task Status Icons
- Manually Intervening in a Workflow

Overview

Opswise Controller 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 Automation Center navigation pane, select Task Instances > Activity .
Step 2	Click the Details icon for the Workflow you want to monitor. The Workflow Task Instance Details for that Workflow displays.
Step 3	Click the View Workflow button to display the Workflow Monitor for that Workflow, as shown in the following example.

The diagram illustrates a workflow monitor with the following task instances and their statuses:

- stonebranch-manualtask-01: Action Required (Red person icon)
- stonebranch-saptask-01: Finished (Green checkmark icon)
- stonebranch-emailtask-03: Finished (Yellow envelope icon)
- stonebranch-zOSTask-01: Skipped (Grey server icon)
- stonebranch-workflow-02: Waiting (Yellow person icon)
- stonebranch-filemonitor-01: Skipped (Blue folder icon)
- stonebranch-taskmonitor-02: Running (Green document icon)
- stonebranch-timertask-01: Skipped (Black clock icon)
- stonebranch-windowstask-01: Skipped (Windows logo icon)
- stonebranch-filetransfertask-01: Skipped (Blue folder with red X icon)
- stonebranch-linuxunixtask-01: Skipped (Linux penguin icon)

Arrows indicate dependencies between tasks, showing a flow from manualtask-01 to saptask-01, emailtask-03, and zOSTask-01. saptask-01 and emailtask-03 both lead to filemonitor-01 and taskmonitor-02. zOSTask-01 leads to timertask-01 and windowstask-01. timertask-01 leads to linuxunixtask-01. filemonitor-01 leads to filetransfertask-01.

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	 Timer	 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.

The Controller assigns a default color for each status, which you can change. See [Colors](#) for a list all statuses and the default colors assigned to them, and for information on how to change any of the colors.

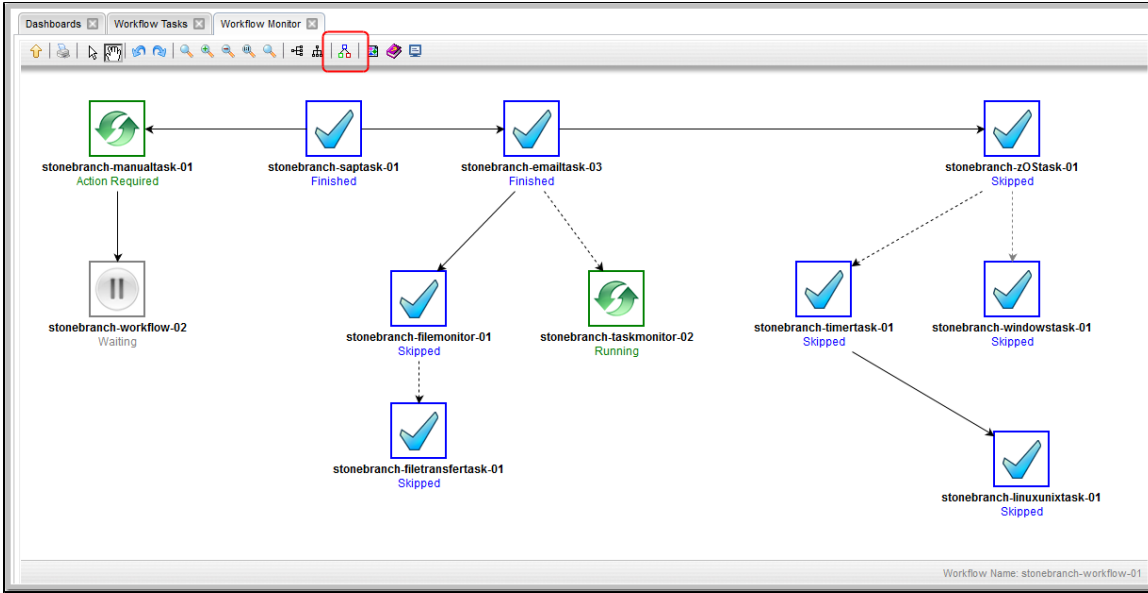
Displaying Task Status Icons


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.

 Running	 Complete	 Not Running	 Problem	 Status Unknown
---	--	---	---	--

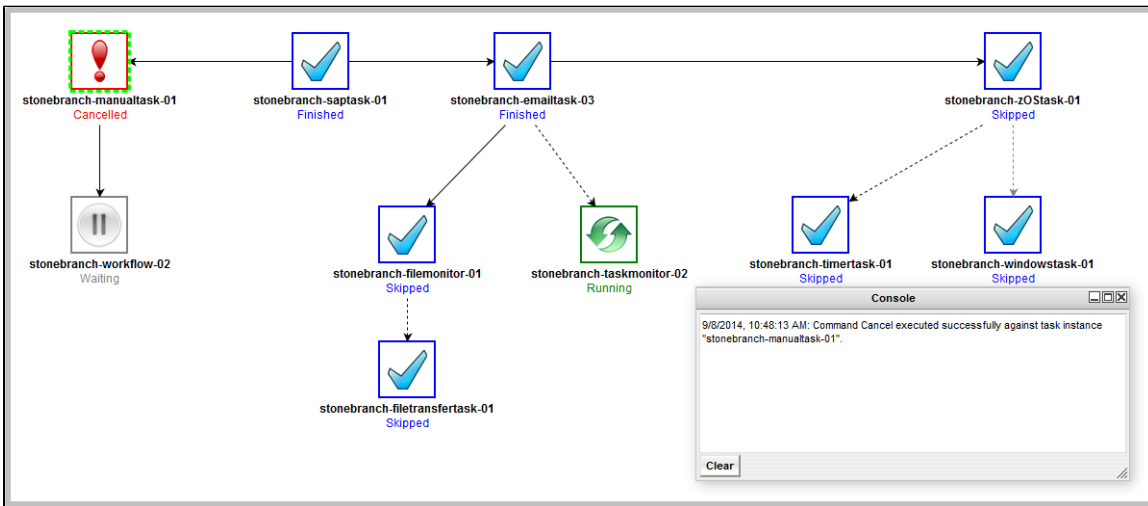
You can switch the Workflow Monitor display mode, which by default shows icons that represent [task types](#), so that the icons represent task status types, as shown in the following example.



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.



Viewing Task Instances for a Specific Task

From the Task Details of any task, you can display a list of task instances for that task and the Details of any specific task instance.

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.

(You also can display Details of any specific task instance from the [Task Instances list](#).)

Step 1 From the [Automation Center](#) navigation pane, select the task from the **Tasks > All Tasks** or **Tasks > <task type> Tasks** list. The Task Details for that task displays.

For example:

The screenshot shows the 'Linux/Unix Task Details' window for the task 'stonebranch-linuxunixtask-01'. The window has a title bar with standard OS controls and a toolbar with 'Update', 'Launch Task', 'Copy', 'Delete', 'Refresh', and 'Close' buttons. Below the toolbar is a tabbed interface with tabs for 'Linux/Unix Task', 'Variables', 'Actions', 'Virtual Resources', 'Mutually Exclusive', 'Instances', 'Triggers', 'Notes', and 'Versions'. The 'Linux/Unix Task' tab is active, showing several sections:

- General:** Task Name: stonebranch-linuxunixtask-01, Version: 1. Fields for Task Description, Member of Business Services, Hold on Start, Virtual Resource Priority (10), and Hold Resources on Failure.
- Linux/Unix Details:** Agent: stonebranch-linuxunixagent-01, Agent Cluster, Agent Variable, Agent Cluster Variable, Credentials, Cluster Broadcast, Credentials Variable, Run as sudo, Command or Script (Command), Command (dir), Parameters, Runtime Directory, Environment Variables (table with Name and Value columns, currently empty), Exit Code Processing (Success Exitcode Range), Exit Codes (0), and Automatic Output Retrieval (-- None --).
- Retry Options:** Maximum Retries (0), Retry Interval (Seconds) (60), and Retry Indefinitely checkbox.
- Time Options:** Late Start, Late Finish, Early Finish checkboxes, and User Estimated Duration (HH:MM:SS).

At the bottom of the window is another toolbar with 'Update', 'Launch Task', 'Copy', 'Delete', 'Refresh', and 'Close' buttons.

Step 2 Click the **Instances** tab. A list of instances for that task displays.

Instance Name	Type	Status	Invoked By	Start Time	End Time	Updated
stonebranch-linuxunixtask-01	Linux/Unix	Finished	Workflow: stonebranch-workflow-01	2014-07-09 13:03:05 -0400	2014-07-09 13:03:05 -0400	2014-07-09 13:03:05 -0400
stonebranch-linuxunixtask-01	Linux/Unix	Finished	Workflow: stonebranch-workflow-01	2014-07-09 10:52:20 -0400	2014-07-09 10:52:20 -0400	2014-07-09 10:52:20 -0400
stonebranch-linuxunixtask-01	Linux/Unix	Finished	Workflow: stonebranch-workflow-01	2014-07-09 10:47:34 -0400	2014-07-09 10:47:34 -0400	2014-07-09 10:47:34 -0400
stonebranch-linuxunixtask-01	Linux/Unix	Finished	Workflow: stonebranch-workflow-01	2014-07-09 09:44:59 -0400	2014-07-09 09:45:00 -0400	2014-07-09 09:45:00 -0400

Step 3 Click the Details icon next to any **Instance Name** on the list to displays the Details for that task instance.

Update Re-run View Parent Retrieve Output... Delete Refresh Close

Linux/Unix Task Instance Virtual Resources Exclusive Requests Output Notes

General

Instance Name: Reference Id:

Task: Invoked By:

Task Description:

Member of Business Services: Execution User:

Virtual Resource Priority: Hold Resources on Failure:

Status

Status: Exit Code:

Status Description:

Queued Time:

Start Time: End Time:

Duration: CPU Time:

Process ID:

Linux/Unix Details

Agent: Agent Cluster:

Agent Variable: Agent Cluster Variable:

Credentials: Credentials Variable:

Run as sudo:

Command or Script:

Command:

Parameters:

Runtime Directory:

Name	Value
No items to show.	

Exit Code Processing:

Exit Codes:

Automatic Output Retrieval:

Start Line: Number of Lines:

Scan Text:

Retry Options

Maximum Retries: Retry Indefinitely:

Retry Interval (Seconds): Current Retry Count:

Statistics

User Estimated End Time: Average Estimated End Time:

Shortest Estimated End Time: Longest Estimated End Time:

Update Re-run View Parent Retrieve Output... Delete Refresh Close

Displaying Task Instance Status

- [Displaying Task Instance Status](#)
- [Task Instance Status Types](#)
 - [Agent-Based Task Types](#)
- [Task Instance Status Colors](#)

Displaying Task Instance Status

You can display the status of one or more task instances from the Activity Monitor, Task Instances list, History list, or Command Line Interface (CLI).

Activity Monitor	To display the status of one or more task instances on the Activity Monitor: <ol style="list-style-type: none"> 1. From the Automation Center navigation pane, select Task Instances > Activity. The Activity Monitor contains a Status column that identifies the current status of every task instance on the list.
Task Instances list	To display the status of one or more task instances on the Task Instances list: <ul style="list-style-type: none"> • From the Automation Center navigation pane, select Task Instances > All Task Instances. The Task Instances list contains a Status column that identifies the current status of every task instance on the list.
History list	To display the status of one or more task instances on the History list: <ul style="list-style-type: none"> • From the Automation Center navigation pane, select Task Instances > History. The History list contains a Status column that identifies the current status of every task instance on the list.
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.

For a list of commands that you can issue against a task instance in each status, see [Commands Supported for Task Instance Statuses](#).

For a description of each command, see [Issuing Commands Against Task Instances](#).



Note

The format of multi-word task status names used in the Controller user interface differs from the format of task status names specified in the `Display Task Instance Status` CLI function. In the user interface, they are separated by a space; in the `Display Task Instance Status` CLI function, the words are separated by an underscore character.

Status Name	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 Controller 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"> • Cancelled • Confirmation Required • Failure • In Doubt • Running/Problems (for sub-workflows) • Start Failure • Undeliverable
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 re-run starting 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 on a predecessor.
----------------	----	-----	---

* Workflows cannot go to Failed status.

Agent-Based Task Types

The following task types are Agent-based task types:

- Linux/Unix
- Windows
- z/OS
- Indesca
- SAP
- File Transfer
- File Monitor
- FTP File Monitor
- System Monitor

Task Instance Status Colors

You can change the default color assigned to each task instance status via the [Colors](#), in the **Reporting** navigation pane.

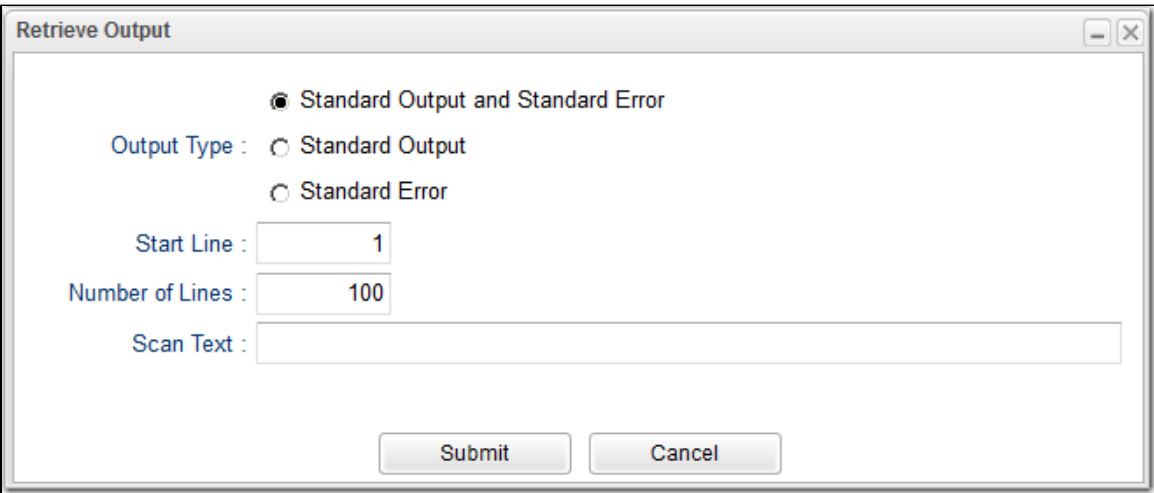

Retrieving Output

- Overview
- Retrieve the Output
- Retrieve Output Field Descriptions

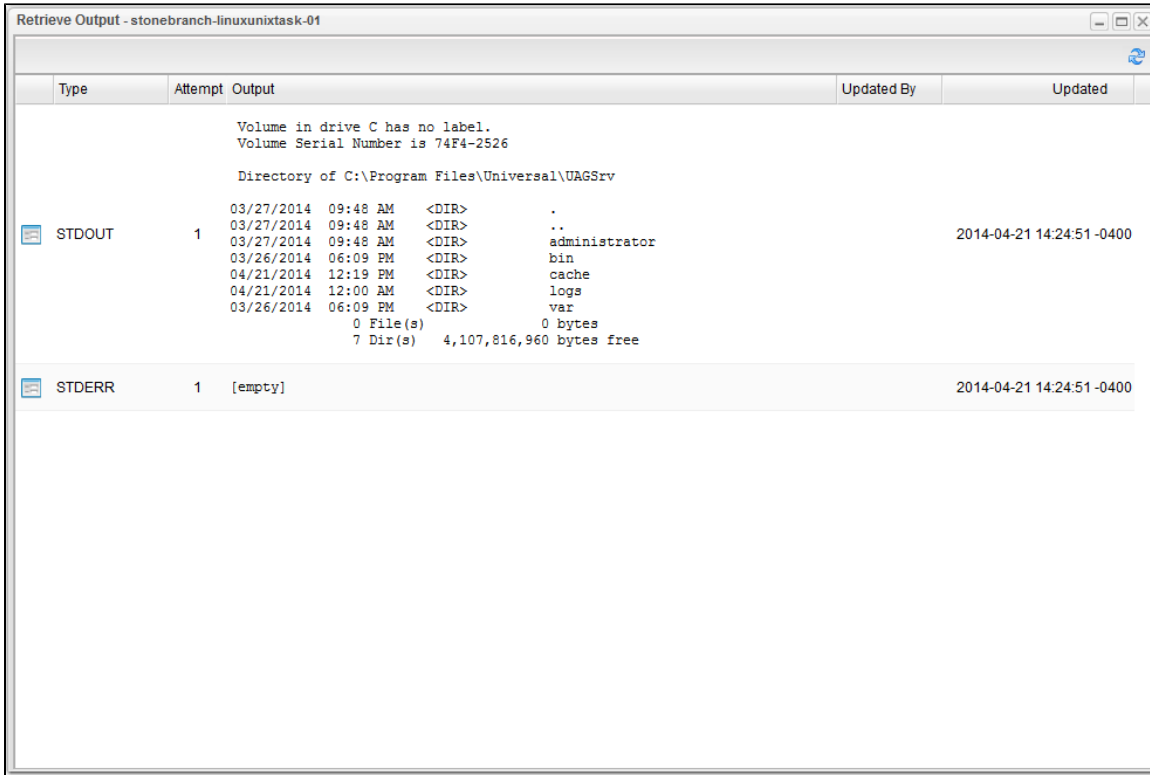
Overview

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 Opwise Controller to retrieve the output.

Retrieve the Output

Step 1	From the Automation Center navigation pane, select Task Instances > Activity or Task Instances > Task Instances to display the Activity Monitor or Task Instances list .
Step 2	Display the task instance Details for the task whose output you want to retrieve.
Step 3	<p>Access the Action menu. If output is available for the task instance, click Retrieve Output.... The Retrieve Output dialog displays.</p>  <p>If the selected task is a z/OS task, the Retrieve Output dialog contains a single Output Type selection:</p> 
Step 4	<p>Using the field descriptions below as a guide, make your selection. Positioning can be accomplished by using Start Line or Scan Text fields to see the Number of Lines both before and after the first match of text.</p> <p>If you want to retrieve data starting at the end of a file, enter -1 in the Start Line field and the number of lines to retrieve in the Number of Lines field.</p>

Step 5 Click the **Submit** button. The Controller retrieves the output you specified and displays a Retrieve Output pop-up containing the requested type of output. (It also writes the output to the **Output** tab of the task instance.)



Retrieve Output Field Descriptions

Field Name	Description
Standard Output and Standard Error	Retrieve both standard output and standard error information returned by the program.
Standard Output	Retrieve standard output only.
Standard Error	Retrieve standard error output only.
z/OS Job Log	Z/OS tasks only; Retrieve information from the z/OS job log.
Start Line	Retrieve data beginning at the line indicated. <ul style="list-style-type: none"> • If a Start Line value is not specified in the task instance Details, the default is 1. • If the Start Line value is -1, data will be retrieved starting at the end of the file.
Number of Lines	Limit the retrieved data to the Number of Lines value in the task instance Details. If a Number of Lines value is not specified, default is the value of the Retrieve Output Default Maximum Lines Opwise Controller system property.
Scan Text:	Regex pattern that the Controller will search for a match for in STDOUT/STDERR (or z/OS job log). The Controller will include the Number of Lines above and below the first line matched. if the Regex pattern is not found, the following message is returned: OPSWISE WARNING - Scan text string not found.