Universal Controller 6.4.x

Troubleshooting and Tutorials

© 2018 by Stonebranch, Inc. All Rights Reserved.

1. Troubleshooting . . . 3
1.1 Problem Resolution . . . 4
1.1.1 Error in your SQL syntax . . . 5
1.1.2 Maximum open cursors have been exceeded . . . 6
1.1.3 Out-of-Range Value during Database Initialization . . . 7
1.1.4 JDBC Connections Time Out . . . 8
1.1.5 DB2 JDBC License Error . . . 9
1.1.6 Processes Will Not Start Automatically (Debian Linux) . . . 10
1.1.7 Error when Starting Controller . . . 11
1.1.8 Tomcat Post Limit - STATUS_MAX_POST_SIZE_EXCEEDED . . . 12
1.1.9 Special Characters not Displayed Correctly . . . 13
1.1.10 Cannot launch a task . . . 14
1.1.11 VBScript stuck in "Running" state . . .
1.1.12 My Universal Controller License has Expired . . . 16
1.1.13 Packet for query is too large . . . 18
1.1.14 Error when refreshing target agents . . . 19
1.1.15 Invalid Call Error . . . 20
1.1.16 Permanent Generation (PermGen) space removed in Java 8 . . . 21
1.2 Error Messages . . . 22
2. Tutorials . . . 24
2.1 Tutorial - Creating and Manually Launching a Task . . . 26
2.2 Tutorial - Creating and Manually Launching a Universal Task . . . 28
2.3 Tutorial - Running a Windows Task . . . 37
2.4 Tutorial - Launching a Task Automatically Using a Simple Time Trigger . . . 41
2.5 Tutorial - Launching a Task Every Monday Except Holidays . . . 44
2.6 Tutorial - Launching a Task Every Two Hours During Workday . . . 50
2.7 Tutorial - Launching Tasks at a Future Time . . . 54
2.8 Tutorial - Launching an Email Task Based on a File Monitor . . . 55
2.9 Tutorial - Launching an Email Task Based on a Task Monitor . . . 61
2.10 Tutorial - Launching Tasks Using a Cron Trigger . . . 69
2.11 Tutorial - Aborting a Process Launched by a Task . . . 70
2.12 Tutorial - Force Finishing, Force Finish-Cancelling, and Cancelling a Task . . . 73
2.13 Tutorial - Accessing Task Instance Details . . . 75
2.14 Tutorial - Monitoring Task Activity . . . 81
2.15 Tutorial - Creating a Simple Workflow . . . 84
2.16 Tutorial - Running a Workflow with a Conditional Path . . . 92
2.17 Tutorial - Running a Workflow with Multiple Conditional Paths . . . 105
2.18 Tutorial - Running a Workflow with Skipped Criteria . . . 108
2.19 Tutorial - Finding and Inserting Tasks in an Active Workflow . . . 112
2.20 Tutorial - Skipping, Unskipping, and Showing-Hiding Skipped Task Instances . . . 120
2.21 Tutorial - Using Variables in a Simple Task . . . 125
2.22 Tutorial - Using Variables in a Workflow . . . 130
2.23 Tutorial - Creating Custom Days and Periods . . . 132
2.24 Tutorial - Generating Forecast Data . . . 141
2.25 Tutorial - Setting Up a Virtual Resource . . . 145
2.26 Tutorial - Creating a Widget . . . 148
2.27 Tutorial - Creating a Dashboard and Adding Widgets . . . 149
2.28 Tutorial - Creating Business Services . . . 152
2.29 Tutorial - Assigning Records to Business Services . . . 153
2.30 Tutorial - Viewing Activity by Business Service . . . 154
2.31 Tutorial - Creating a Report . . . 155
2.32 Tutorial - Creating a Report Based on Business Services . . . 159
2.33 Tutorial - Scheduling a Report . . . 161
2.34 Tutorial - Creating Users and Assigning Roles and Permissions . . . 164
2.35 Tutorial - Creating User Groups and Assigning Permissions . . . 167
2.36 Tutorial - Creating and Promoting a Bundle . . . 169
2.37 Tutorial - Scheduling the Promotion of a Bundle . . . 175

Troubleshooting

Troubleshooting

Troubleshooting information is categorized into two areas:

- Problem Resolution
- Error Messages

Problem Resolution

- Problem Resolution
  - Database
  - Installation
  - Operations

Problem Resolution

This page provides links to problems, and their solutions, that you might encounter with Universal Controller.

Database

- Error in your SQL syntax
- Maximum open cursors have been exceeded
- Out-of-Range Value during Database Initialization
- JDBC Connections Time Out
- DB2 JDBC License Error

Installation

- Processes Will Not Start Automatically (Debian Linux)
- Error when Starting Controller
- Tomcat Post Limit: STATUS_MAX_POST_SIZE_EXCEEDED
- Special Characters not Displayed Correctly

Operations

- Cannot launch a task
- VBScript stuck in "Running" state
- My Universal Controller License has Expired
- Packet for query is too large
- Invalid login credentials for refreshing target agents
- Invalid Call Error: Invalid call to setDataSource()
- Permanent Generation (PermGen) space removed in Java 8

Error in your SQL syntax

Problem

When you execute an SQL task that includes multiple SQL commands, the following error message (for example) may display:

```
INSERT INTO opswise_demo (name, value) values ('A', 'F');
INSERT INTO opswise_demo (name, value) values ('B', 'S');
INSERT INTO opswise_demo (name, value) values ('C', 'F');
```

Solution

Multiple queries, by default, are disabled for MySQL. To enable multiple queries, append the following string to the Connection URL field in the Database Connections resource definition:

```
?allowMultiQueries=true
```

The following example is a URL connection string for a MySQL Database Connection resource definition:

```
jdbc:mysql://localhost:3306/opswise?allowMultiQueries=true
```

Maximum open cursors have been exceeded

Problem

During large imports on Oracle, you could receive following error message:

```
ORA-01000: maximum open cursors exceeded
```

(The cursors are used only during the import; they then are closed.)

Issue the following **sql*plus** utility command to check the current value for maximum open cursors:

```
show parameter open_cursors
```

A listing similar to the following will display:

```
SQL> show parameter open_cursors;

NAME                                 TYPE        VALUE
------------------------------------ ----------- ------------------------------
open_cursors                         integer     1000
```

Solution

An **open_cursors** value of 1000 should be sufficient for all large imports.

You can temporarily set the **open_cursors** value with the following SQL:

```
alter system set open_cursors=1000
```

To make a permanent change, you must set the **open_cursors** value in the initialization parameters file.

Out-of-Range Value during Database Initialization

Problem

During the database initialization performed on initial start-up, you could receive the following message:

```
The conversion of a varchar data type to a datetime data type of the value is out of range.
```

The problem likely is that the database was created in SQL SERVER Management Studio with a user that has other than English as the default language.

Solution

Verify the installed default language and set the language to U.S. English.

To check what default language a server has installed, use the following SQL command:

```
sp_configure 'default language'
```

If the resulting value is not 0, the default language is not U.S. English. Run the following SQL command to find the installed default language setting and date format used:

```
select name ,alias, dateformat
from syslanguages
   where langid =
   (select value from master..sysconfigures
      where comment = 'default language')
```

To set the default language to U.S. English, use the following SQL statements:

```
sp_configure 'default language', 0
reconfigure with override
```

For further details, refer to this Microsoft Support page.

JDBC Connections Time Out

Problem

JDBC connections from Linux to MS SQL Server 2008 R2/Windows 2008 R2 time out after 40 seconds causing SQL/Stored Procedure Tasks that take longer than 40 seconds to fail with the following exception:

```
2014-09-22-14:51:37:034 -0400  ERROR
[Ops.General.15.EP.SqlHandler.ecd8ab62183f4b9dbf32d3ea4ad0a126.74b824ad1ca84142a40d3ec1f84d4d2b.0]
SQLServerException - Connection reset
com.microsoft.sqlserver.jdbc.SQLServerException: Connection reset
          at com.microsoft.sqlserver.jdbc.SQLServerConnection.terminate(SQLServerConnection.java:1667)
          at com.microsoft.sqlserver.jdbc.SQLServerConnection.terminate(SQLServerConnection.java:1654)
          at com.microsoft.sqlserver.jdbc.TDSChannel.read(IOBuffer.java:1789)
          at com.microsoft.sqlserver.jdbc.TDSReader.readPacket(IOBuffer.java:4838)
          at com.microsoft.sqlserver.jdbc.TDSCommand.startResponse(IOBuffer.java:6150)
          at
com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.doExecutePreparedStatement(SQLServerPreparedState
at
com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement$PrepStmtExecCmd.doExecute(SQLServerPreparedStatem
at com.microsoft.sqlserver.jdbc.TDSCommand.execute(IOBuffer.java:5696)
          at
com.microsoft.sqlserver.jdbc.SQLServerConnection.executeCommand(SQLServerConnection.java:1715)
          at
com.microsoft.sqlserver.jdbc.SQLServerStatement.executeCommand(SQLServerStatement.java:180)
          at
com.microsoft.sqlserver.jdbc.SQLServerStatement.executeStatement(SQLServerStatement.java:155)
          at
com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.execute(SQLServerPreparedStatement.java:332)
          at
com.stonebranch.opswise.server.events.SqlEventHandler.storedProc(SqlEventHandler.java:266)
```

Resolution

To disable TCP Chimney Offload, follow these steps:

Step 1	Use administrative credentials to open a command prompt.
Step 2	At the command prompt, enter the following command: `netsh int tcp set global chimney=disabled`
Step 3	Press ENTER:.

For additional information, see:

http://support.microsoft.com/kb/951037

DB2 JDBC License Error

Problem

A SQL or Stored Procedure task using a DB2 Database Connection may fail with the following error message:

```
The IBM Data Server for JDBC and SQLJ license was invalid or was not activated for the DB2 for z/OS
subsystem. If you are connecting
directly to the data server and using DB2 Connect Unlimited Edition for System z, perform the
activation step by running the activation
program in the license activation kit. If you are using any other edition of DB2 Connect, obtain the
license file, db2jcc_license_cisuz.jar,
from the license activation kit, and follow the installation directions to include the license file in
the class path.
```

Solution

The **`db2jcc_license_cisuz.jar`** file needs to be included in the classpath for Universal Controller.

Step 1	Copy the **`db2jcc_license_cisuz.jar`** file to the following directory: **Windows** **`[tomcat-install]\webapps\opswise\WEB-INF\lib`** **UNIX** **`[tomcat-install]/webapps/opswise/WEB-INF/lib`**
Step 2	Restart Universal Controller.

Processes Will Not Start Automatically (Debian Linux)

Problem

For Debian Linux environment: Outboard 5.1.0 processes will not start automatically at boot time.

Debian Linux does not provide the **chkconfig** command and therefore cannot work with the runlevels specified in the opsagent, opstransport, opsmsghub scripts provided in `/etc/init.d`.

This is a known problem; we are working on a solution.

Error when Starting Controller

Problem

Upon starting the Universal Controller, the `opswise.log` shows ERR:

```
SQLSTATE: HY000, SQLERR: 1040, ERRMSG: [unixODBC][MySQL][ODBC 5.1 Driver]Too many connections
```

Solution

You must set additional connections in your database server:

```
MySQL - /etc/my.conf - max_connections=500 (default is either 100 or 150)
```

Tomcat Post Limit - STATUS_MAX_POST_SIZE_EXCEEDED

Problem

The following error message displays:

```
The server did not receive the data that was sent to it.  Please see the documentation for
isc.RPCResponse.STATUS_MAX_POST_SIZE_EXCEEDED
```

Resolution

Remove the post limit by specifying the following attribute on the **<Connector>** element in **`conf/server.xml`**:

`maxPostSize="-1"`

Special Characters not Displayed Correctly

Problem

Some special characters not getting displayed correctly in your browser GUI.

Resolution

Tomcat on Windows requires you to define code page UTF-8 as the default code page for war files.

To do this, add the following to the Java options statement just as you did with the memory parameter:

```
-Dfile.encoding=UTF8
```

Cannot launch a task

Problem

You cannot launch a task.

The problem may be with your credentials. Check the stderr for the following message:

```
ops_suexec: Not enough privileges. Check SUID bit and binary owner.
```

Solution

If the error message is present, issue the following commands as **root** in the `$WMS_HOME/bin` directory:

```
chown root ops_suexec
chmod 4755 ops_suexec
```

VBScript stuck in "Running" state

Problem

By default, Windows uses a GUI-based VBScript interpreter (`wscript.exe`). With this interpreter, if your script tries to display an error message that requires a user-response (for example, **Click OK**), you will never see the dialog box. The script therefore gets stuck in the "Running" state.

Solution

To avoid this, we recommend you use the console version of the VBScript interpreter (`cscript.exe`). To do so, specify `cscript.exe` before the script name in a task definition, as shown in the following example:

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

My Universal Controller License has Expired

Problem

If your Universal Controller license has expired, you will not be able to run any tasks. If you attempt to run a task under an expired license, the Controller will place the task in the Defined state.

When you log in to the Controller, the Universal Automation Center Console will displays a license expiration informational message if your license will expire within a week, and an error message if your license already has expired.

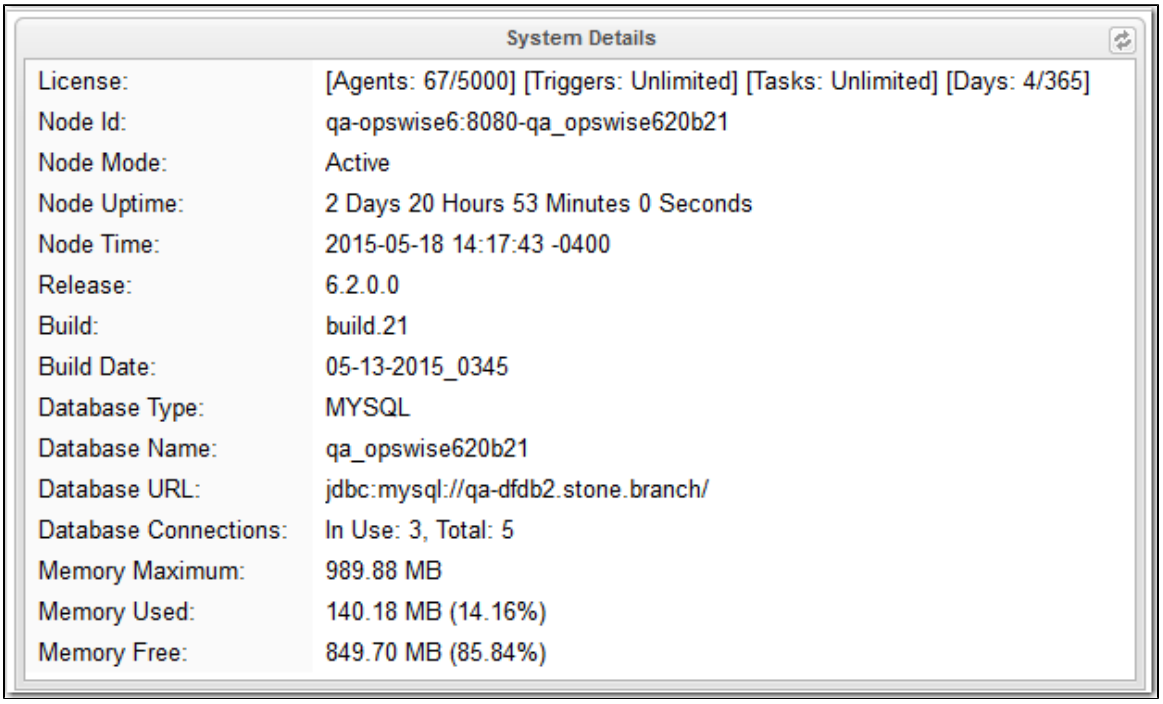
Additionally, if you have configured the Controller for System Notifications, system notifications are sent when the Controller license will expire in seven days and if the license already has expired.

There are two other methods you can use to check on license expiration.

Method 1

Check the System Details widget (view the system-defined Home Dashboard or, on the Reporting navigation pane, click **Widgets**) to see how many days are left on your license.

The **Days: #/#** entry in the **License** field indicates the current day of your current license and the total number of days in the license. If the numbers are identical, your license has expired.



Method 2

Check the Controller log file.

If your license has expired, the following two messages should appear in the log at the midnight roll-over:

```
2015-01-07-00:00:00:006   WARN [Ops.Timer.Forecast_Refresh.0] License Violation: Number of Days has
exceeded # suspending system
2015-01-07-00:00:00:006   INFO [Ops.Timer.Forecast_Refresh.0] Pausing the server.
```

Additionally, this message should appear in the log if you try to run a task under an expired license:

```
2015-01-07-09:32:27:728   INFO [Ops.Available.2367.0] System paused, waiting for resume
```

Solution

Contact Stonebranch Customer Support.

Packet for query is too large

Problem

During operations, the following message may appear in the Universal Controller log:

```
Packet for query is too large (1084852 > 1048576).
```

Solution

Change this value on the database server by setting the MySQL **max_allowed_packet** configuration variable.

For detailed information about this variable, refer to:

- MySQL 5.6.x reference manual
- MySQL 5.7.x reference manual

Error when refreshing target agents

Problem

An error occurs when you click **Refresh Target Agents** on a Promotion Target record and you are using invalid login credentials for the target Universal Controller instance.

The user interface on source machine will show the following error:

```
GET http://NN.NNN.NN.N:8080/opswise/resources/agents/list returned a response status of 401
Unauthorized
```

The the Controller log on the source machine will show the following error:

```
ERROR [http-8080-10] com.sun.jersey.api.client.UniformInterfaceException:
GET http:///opswise/resources/agents/list returned a response status of 401 Unauthorized"
```

The target machine will return the following error:

```
ERROR [http--#] *** ERROR *** Login using Basic Authentication failed for:
```

Solution

Update the Promotion Target record with valid login credentials and try the promotion again.

Invalid Call Error

Problem

The following error message displays:

```
onUncaughtException: Exception caught: Invalid call to setDataSource() passing null.
```

Resolution

You may need to update the **LimitRequestLine** property in the Apache `httpd.conf` file to its default value, **8190**.

Permanent Generation (PermGen) space removed in Java 8

Problem

The Permanent Generation (PermGen) space has been completely removed in Java 8.

If you specify the Maximum PermGen Size option (`-XX:MaxPermSize=<NNN>`) for a Java 8 VM, you may see a warning message similar to the following.

```
Java HotSpot(TM) 64-Bit Server VM warning: ignoring option MaxPermSize=512m; support was removed in
8.0
```

Resolution

The PermGen space is superseded by a new space called Metaspace in Java 8.

You can specify the Maximum Metaspace Size with the analogous `-XX:MaxMetaspaceSize=<NNN>` option; however, the default (no limit) is recommended for most deployments.

Error Messages

Error Messages

This page identifies error messages (in alphabetical order) that you may receive for Universal Controller.

For each error, there is a link to the cause problem, and its solution, in Problem Resolution.

```
GET http://NN.NNN.NN.N:8080/opswise/resources/agents/list returned a response status of 401
Unauthorized
```

(in user interface on source machine)

```
ERROR [http-8080-10] com.sun.jersey.api.client.UniformInterfaceException:
GET http:///opswise/resources/agents/list returned a response status of 401 Unauthorized"
```

(in Controller log on source machine)

```
ERROR [http--#] *** ERROR *** Login using Basic Authentication failed for:
```

(on Target machine)

See Error when refreshing target agents.

```
INSERT INTO opswise_demo (name, value) values ('A', 'F');
```

See Error in your SQL syntax.

```
Java HotSpot(TM) 64-Bit Server VM warning: ignoring option MaxPermSize=512m; support was removed in
8.0
```

See Permanent Generation (PermGen) space removed in Java 8

```
onUncaughtException: Exception caught: Invalid call to setDataSource() passing null.
```

See Invalid Call Error.

```
ops_suexec___Not enough privileges. Check SUID bit and binary owner
```

See Cannot launch a task.

```
ORA-01000: maximum open cursors exceeded
```

See Maximum open cursors have been exceeded.

```
Packet for query is too large (1084852 > 1048576
```

See Packet for query is too large.

```
SQLSTATE: HY000, SQLERR. 1040, ERRMSG. unixODBC MySQL ODBC 5.1 Driver Too many connections
```

See Error when Starting Controller.

```
The conversion of a varchar data type to a datetime data type resulted in an out-of-range value.
```

See Out-of-Range Value during Database Initialization.

```
The IBM Data Server for JDBC and SQLJ license was invalid or was not activated for the DB2 for z/OS
subsystem. If you are connecting directly to the data server and using DB2 Connect Unlimited Edition
for System z, perform the activation step by running the activation program in the license activation
kit. If you are using any other edition of DB2 Connect, obtain the license file,
db2jcc_license_cisuz.jar, from the license activation kit, and follow the installation directions to
include the license file in the class path.
```

See DB2 JDBC License Error.

```
The server did not receive the data that was sent to it. Please see the documentation for
isc.RPCResponse.STATUS_MAX_POST_SIZE_EXCEEDED
```

See Tomcat Post Limit - STATUS_MAX_POST_SIZE_EXCEEDED.

Tutorials

Introduction

The tutorials guide you through features of Universal Controller. They also provide links to more detailed information about each aspect of the Controller.

Once you have completed the tutorials, you should have a basic understanding of how the features work together to automate your environment.

Before you begin, we recommend that you read the Setting up Universal Controller and User Interface sections of this documentation to familiarize yourself with user interface navigation and terminology.

> **Note**
> During the course of performing these tutorials, you will build up a small database of demonstration records that you will re-use in subsequent exercises. To avoid having to reenter data, do not delete the records.

The navigation pane on the left lists the tutorials in the sequence they should be read; many of them refer to information created in one or more previous tutorials.

The following table lists the tutorials by topic, rather than sequence.

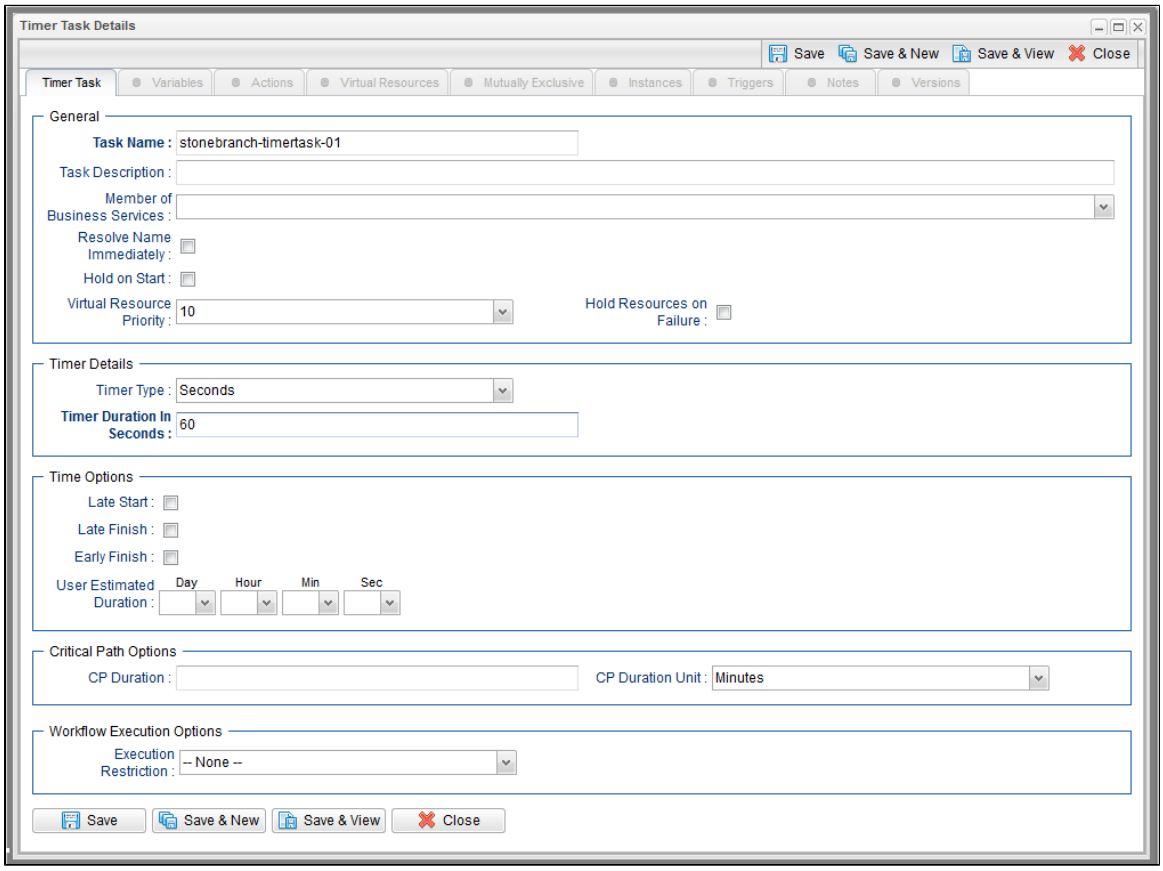
Tasks
Creating and Manually Launching a Simple Task
Creating and Manually Launching a Universal Task
Running a Windows Task
Launching a Task Automatically Using a Simple Time Trigger
Launching a Task Every Monday Except Holidays
Launching a Task Every Two Hours During Workday
Launching Tasks at a Future Time
Launching an Email Task Based on a File Monitor
Launching an Email Task Based on a Task Monitor
Launching Tasks Using a Cron Trigger
Aborting a Process Launched by a Task
Force Finishing, Force Finish-Cancelling, and Cancelling a Task
Accessing Task Instance Details
Monitoring Task Activity
Workflows
Creating a Simple Workflow
Running a Workflow with a Conditional Path
Running a Workflow with Multiple Conditional Paths

Running a Workflow with Skipped Criteria
Finding and Inserting Tasks in an Active Workflow
Skipping, Unskipping, and Showing-Hiding Skipped Task Instances
Variables
Using Variables in a Simple Task
Using Variables in a Workflow
Custom Days
Creating Custom Days and Periods
Forecasting
Generating Forecast Data
Virtual Resources
Setting Up a Virtual Resource
User Interface
Creating a Widget
Creating a Dashboard and Selecting Widgets
Business Services
Creating Business Services
Assigning Records to Business Services
Viewing Activity by Business Service
Reports
Creating a Report
Creating a Report Based on Business Services
Scheduling a Report
Security
Creating Users and Assigning Permissions
Creating User Groups and Assigning Permissions
Bundles and Promotion
Creating and Promoting a Bundle
Scheduling the Promotion of a Bundle

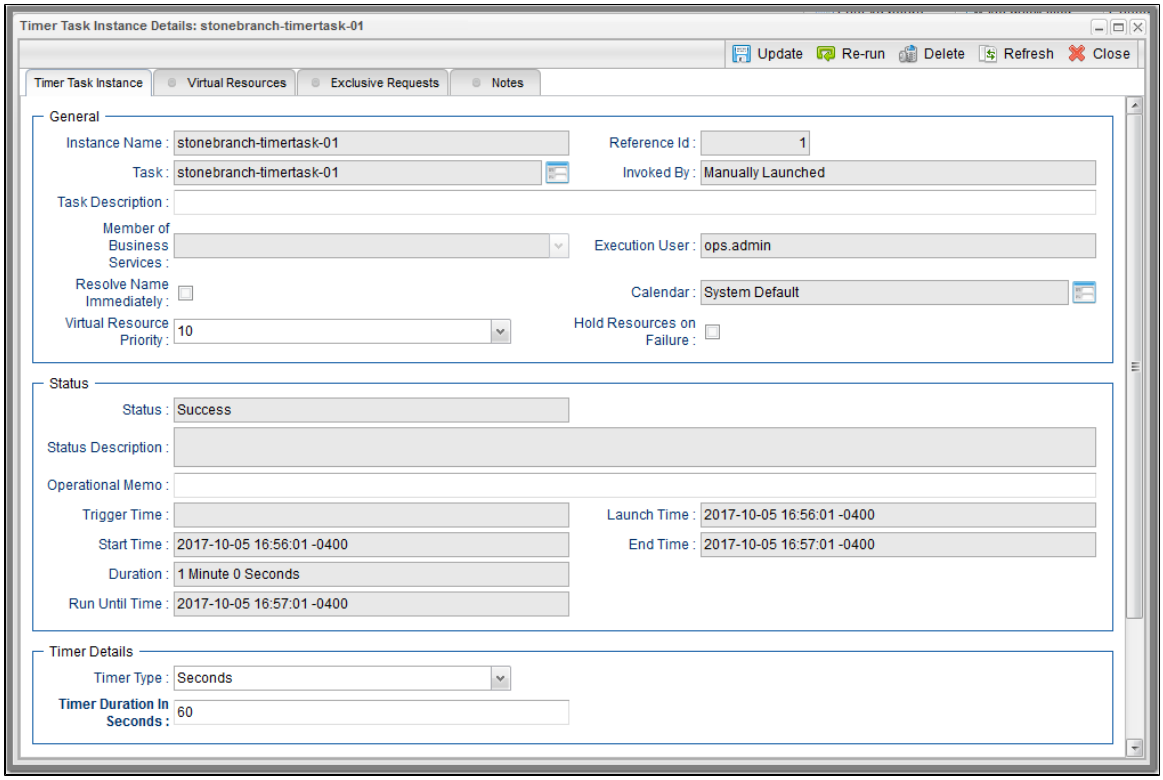
Tutorial - Creating and Manually Launching a Task

In this tutorial, you will:

- Create a Timer task.
- Manually launch the task.
- View task instance Details of the task.

Step 1	From the Automation Center navigation pane, select **Tasks > Timer Task**. The Timer Tasks list displays.
Step 2	Click **New**. The Timer Task Details displays. • In the Task Name field, enter **stonebranch-timertask-01**. • In the Time Duration in Seconds field, enter **60**. 
Step 3	Click **Save**.
Step 4	On the Timer Tasks list, right-click **stonebranch-timertask-01** task to display an Action menu.
Step 5	Click **Launch Task**.
Step 6	From the **Automation Center** navigation pane, select **Task Instances > Activity**. The Activity Monitor displays.
Step 7	Locate the **stonebranch-timertask-01** task instance. When the task instance completes, the status changes from **Running** to **Success**.

Step 8 Click the Details icon next to the **Instance Name** of **stonebranch-timertask-01** to display Details of the task instance.



For additional information, see:

- Creating Tasks

Tutorial - Creating and Manually Launching a Universal Task

- Overview
  - Before You Begin
- Create a Universal Template
- Create Universal Template Fields
- Create a Universal Task

Overview

To run a Universal Task, you must:

- Create a Universal Template, for which Universal Controller automatically creates a Universal Task type. This includes:
  - Entering a script that any Universal Task created for that Universal Task type will execute.
  - Creating fields, for which the Controller will both automatically assign variables (to be used in the script) and add to any Universal Task created for that Universal Task type.
- Replace variables in the script with the variables assigned to the user-defined Fields that you created in the Universal Template.
- Create a Universal Task for the Universal Task type based on the Universal Template that you created.
- Enter/select values for the user-defined fields in the Universal Task that match the fields you created in the Universal Template.
- Run the Universal Task.
- Retrieve the task output and verify that the script variables have been resolved to the Universal Task field values.

In this tutorial, you will:

- Create a Universal Template.
- Define fields for a Universal Task in the Universal Template.
- Create a Universal task.

Before You Begin

When you create a Universal Template, you create fields that will display in the Details of the Universal Task (based on this Universal Template) that you will create. These Universal Template fields are assigned variables to be used in the script. The script should contain these variables in a specific format based on information that you provide when creating the Universal Template.

Create a Universal Template

First, create a Universal Template:

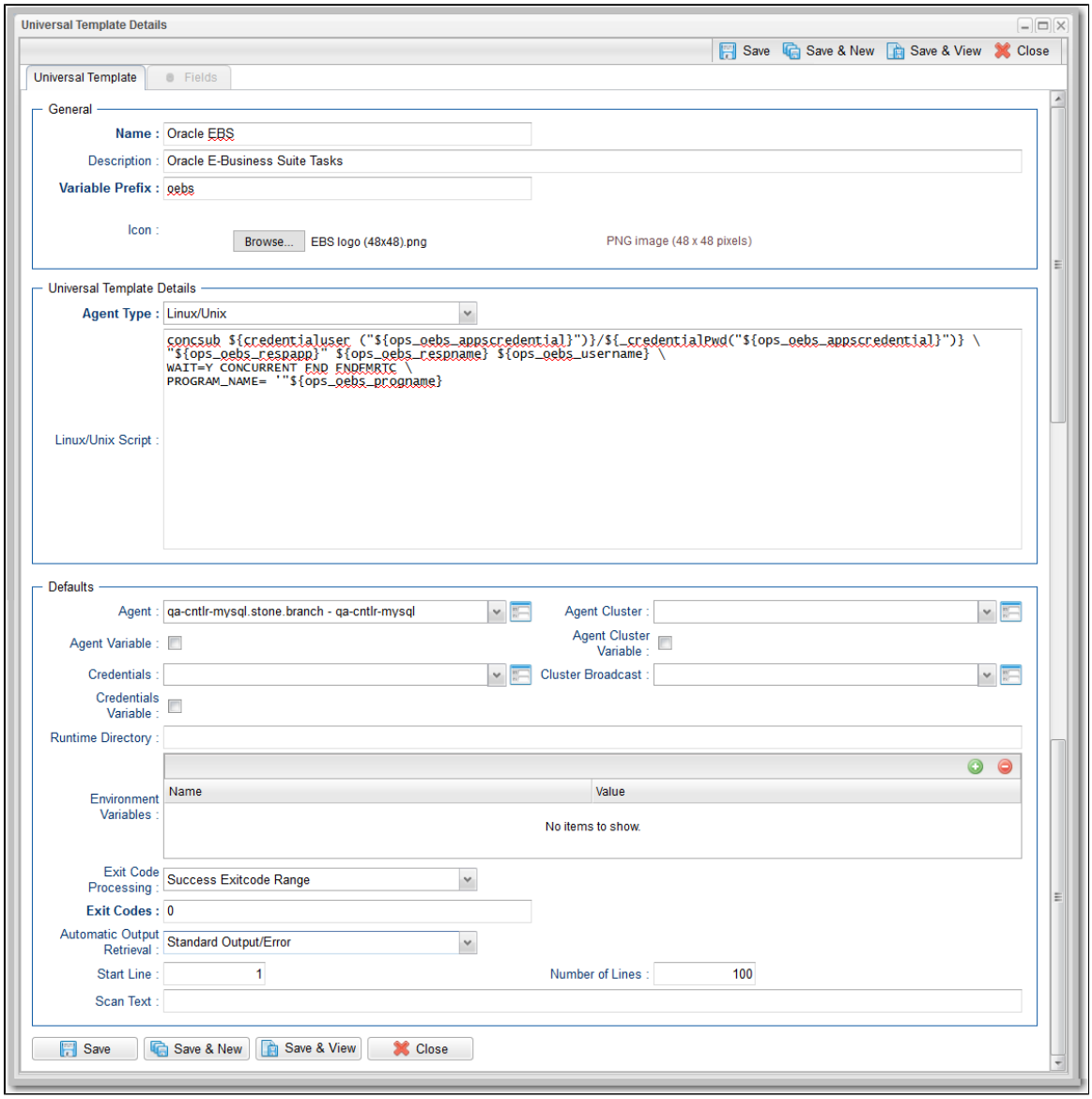
Step 1	From the Administration navigation pane, select **Configuration> Universal Templates**. The Universal Templates list displays.

Step 2 Click **New**. The Universal Template Details displays.

- In the Name field, enter **Oracle EBS**.
- In the Description field, enter **Oracle E-Business Suite Tasks**.
- In the Variable Prefix field, enter: **oebs**.
- In the Agent Type field, select **Linux/Unix**.
- In the Linux/Unix Script field, enter:

```
concsub ${credentialuser
("${ops_oebs_appscredential}")}/${_credentialPwd("${ops_oebs_appscredential}")} \
"${ops_oebs_respapp}" ${ops_oebs_respname} ${ops_oebs_username} \
WAIT=Y CONCURRENT FND FNDFMRTC \
PROGRAM_NAME= '"${ops_oebs_progname}
```

- In the Agent field, select a Linux/Unix agent on which the Universal Task based on this agent will run.



Step 3 Optionally, in the Icon field, browse for an icon (PNG image, 48x48 pixels) to be used instead of the default icon for any Universal Tasks based on this Universal Template.

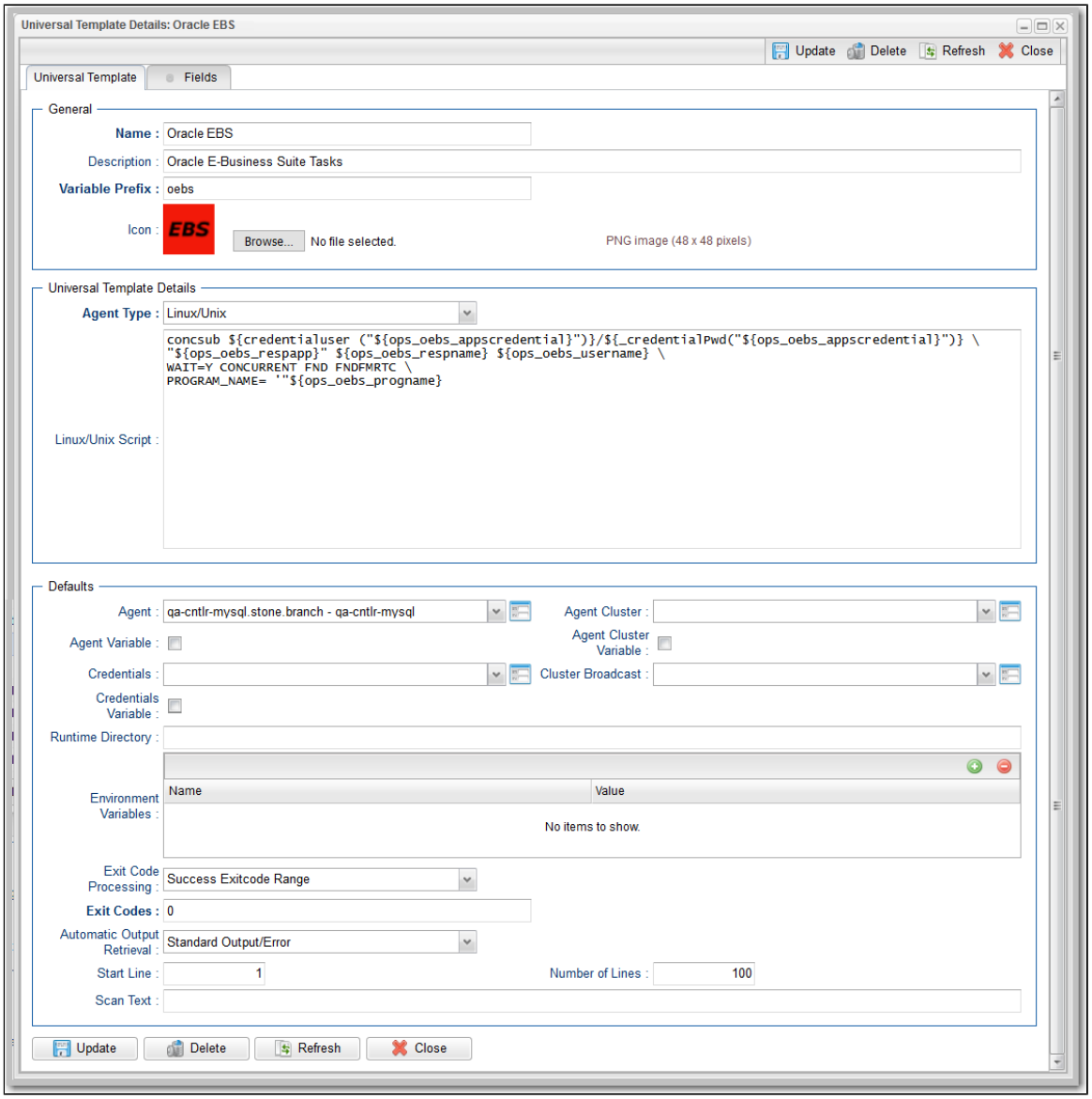
If you select an icon, the file name display next to the Browse button. When you save the Universal Template, the logo itself displays in the Icon field (see Create Universal Template Fields, below).

Step 4 Click **Save**.

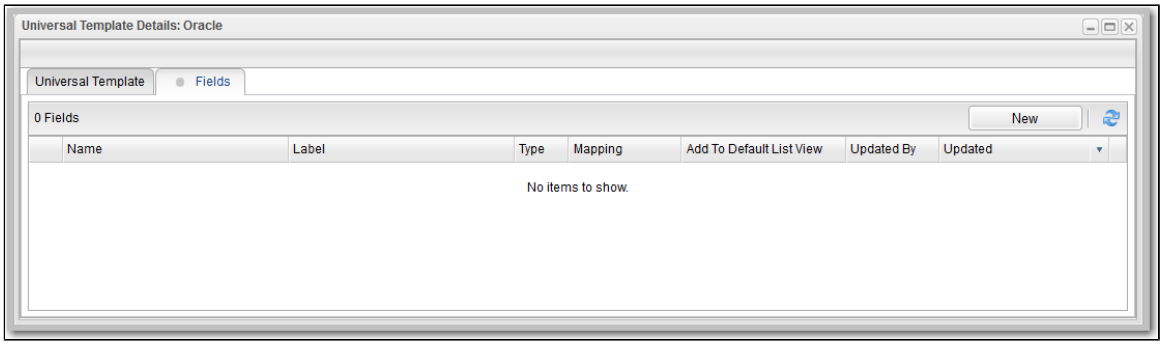
Create Universal Template Fields

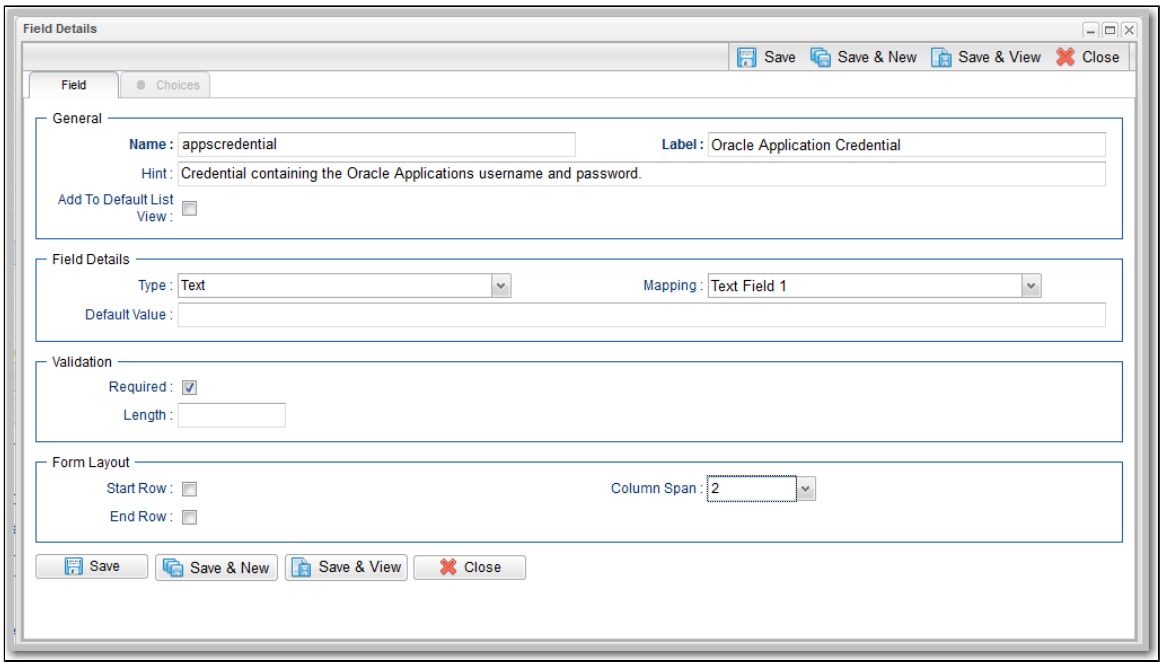
Next, create fields in the template:

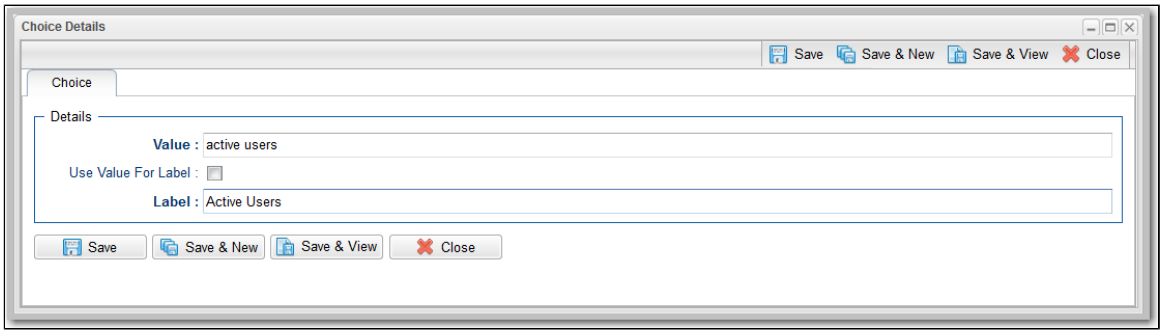
Step 1 Open the **Oracle EBS** template that you just created.



Step 2 Click the **Fields** tab to display an empty Fields list.



Step 3	Click **New**. The Field Details for a new Field displays. • In the Name field, enter **appscredential**. • In the Label field, enter **Oracle Application Credential**. • In the Hint field, enter **Credential containing the Oracle Applications username and password.**. • In the Type field, select **Text**. • In the Required field, enter a check mark. 
Step 4	Click **Save** to save the Field and re-display the Fields list.
Step 5	Click **New** and create four more Fields. • Name = **respapp**. • Label = **Responsibility Application**. • Type = **Text**. • Name = **respname**. • Label = **Responsibility Name**. • Type = **Text**. • Name = **username**. • Label = **Application User**. • Type = **Text**. • Name = **progname**. • Label = **Program Name**. • Type = **Choice**. • Default = **active users** Click **Save** to save each Field.

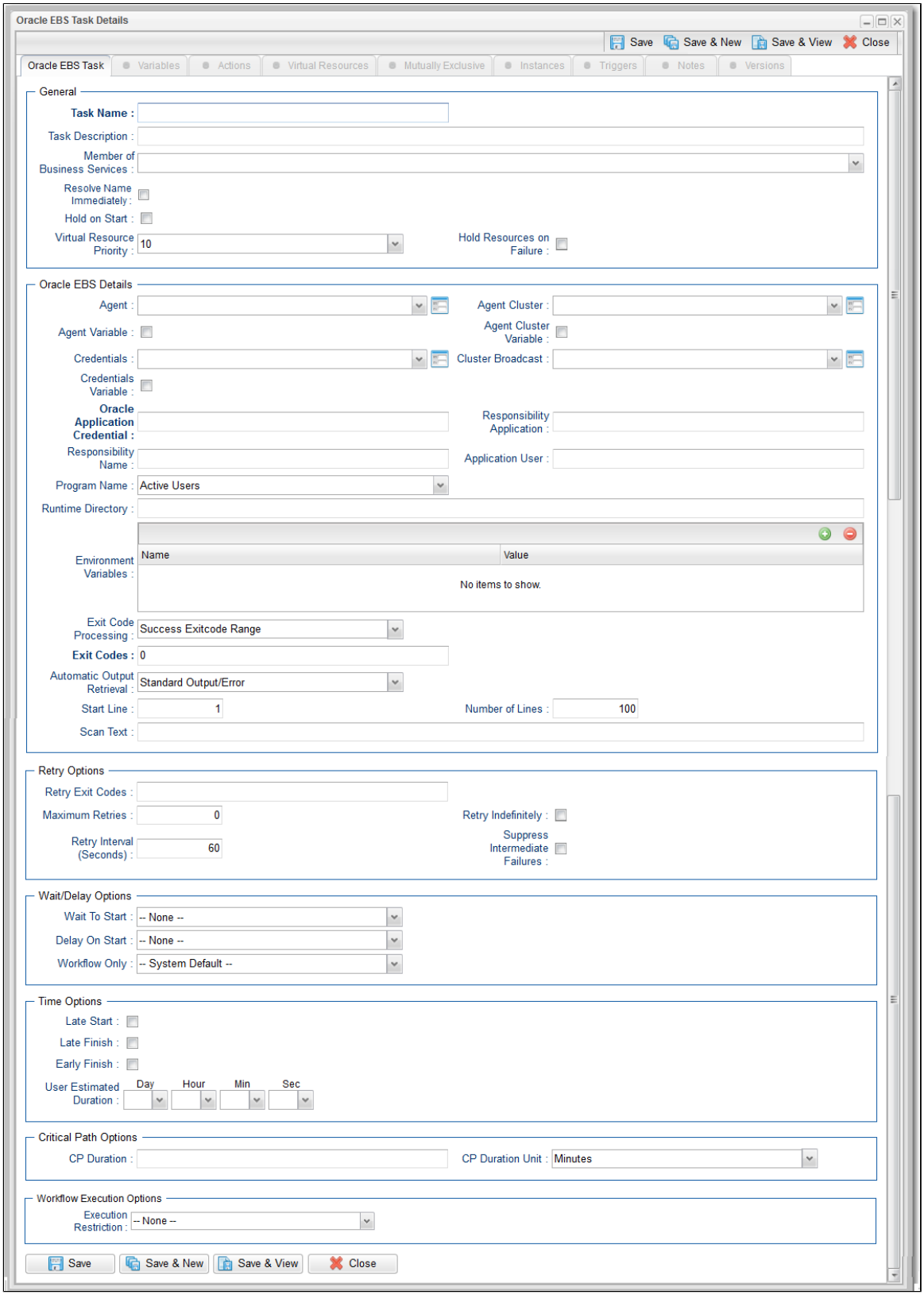
Step 6	When you save the **progname** field, the Choices tab is enabled so that you can define user-selectable Choices for that Field. An empty Choices list displays under the enabled Choices tab, and an empty Choice Details for a new Choice automatically displays. • In the Value field, enter **active users**. • In the Label field, enter **Active Users**. 
Step 7	Click **Save** to save the Choice and then create two more Choices for the **progname** field: • Value = **inactive users**. • Label = **Inactive Users**. • Name = **temporary users**. • Label = **Temporary Users**. Click **Save** to save each Field.
Step 8	Click the **Field** tab to re-display the Field Details, click the **Close** to return to the Fields list, and then click click the **Universal Template** tab to return to the Universal Template Details.
Step 9	Check the script to make sure that the variables for the five Fields that you created are in the following format: `ops_<Variable Prefix>_<Field Name>`. For example: `ops_oebs_appscredential`.
Step 10	Click **Save** to save the template.

Create a Universal Task

You now can create a Universal Task based on Stonebranch Template One.

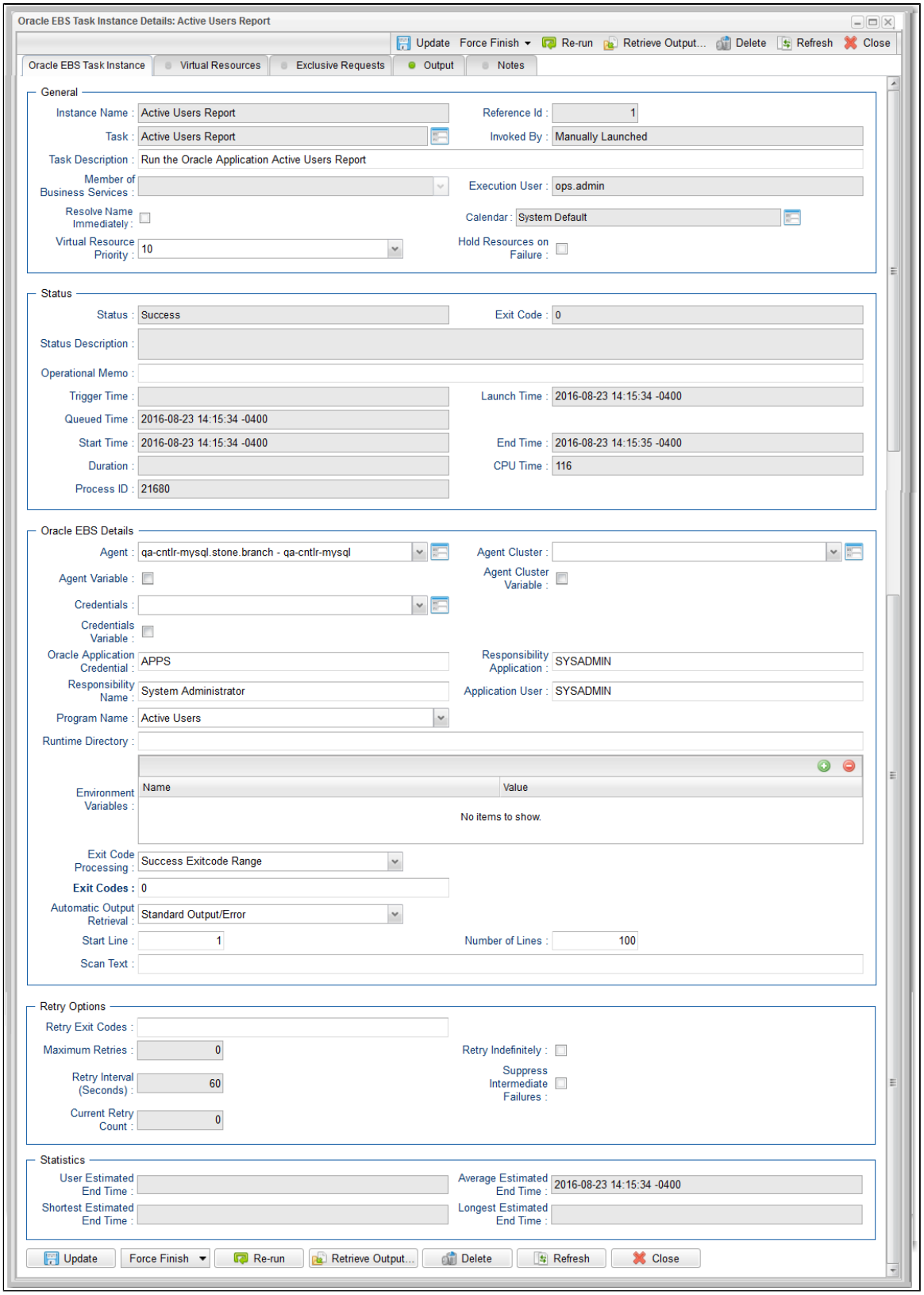
Step 1	Select the Automation Center navigation pane.
Step 2	Right-click inside the navigation pane and, on the Action menu that displays, click **Refresh Navigation Tree**. An **Oracle EBS Tasks** Universal Task task type now displays in the Universal Tasks folder.
Step 3	Click **Oracle EBS Tasks** to display an empty tasks list for this Universal Task type.

Step 4	Click **New** to display Details for a new Oracle EBS Tasks task.
Step 5	In the Task Name field, enter: Active Users Report.
Step 6	Review the Details section fields that you defined in the Oracle EBS Universal Template to make sure that their names, defaults, and requirement settings are correct.



Step 7	Enter the following field values: • Oracle Application Credential: APPS • Responsibility Application: SYSADMIN • Responsibility Name: System Administrator • Application User: SYSADMIN
Step 8	Click the **Save** button to save the task.
Step 9	Right-click Active Users Report on the Oracle EBS Task list and, on the Action menu that displays, click Launch Task.

Step 10 Click **Activity** in the Automation Center navigation pane and open the Active Users Report task instance:



Step 11 In the **Status** section of the task instance Details, check the Status and Exit Code fields to verify that the task ran to success. Optionally, retrieve output from the task instance to view details about the run.

For additional information, see:

- Universal Templates
- Universal Tasks

Tutorial - Running a Windows Task

- Creating a Simple Windows Task
- Manually Retrieving Output from a Windows Task
- Attaching Output to an Email Notification

To perform this tutorial, you need a running Universal Agent for Windows.

Creating a Simple Windows Task

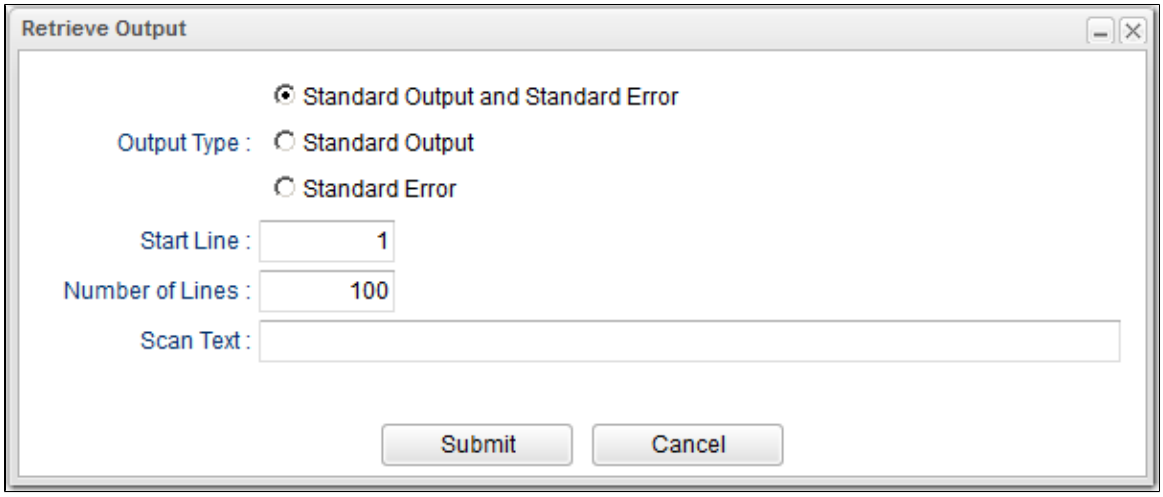
Step 1	From the Automation Center navigation pane, select **Tasks > Windows Tasks**. The Windows Tasks list displays.
Step 2	In the empty Windows task Details below the list: • In the Task Name field, enter **stonebranch-windowstask-01**. • In the Agent field, select your Windows agent. • In the Command field, enter **md c:\tutorial**.
Step 3	Click **Save**.
Step 4	On the Windows Tasks list, right-click **stonebranch-windowstask-01** to display an Action menu.
Step 5	Click **Launch Task**.
Step 7	Check the Activity Monitor for the task instance.

Manually Retrieving Output from a Windows Task

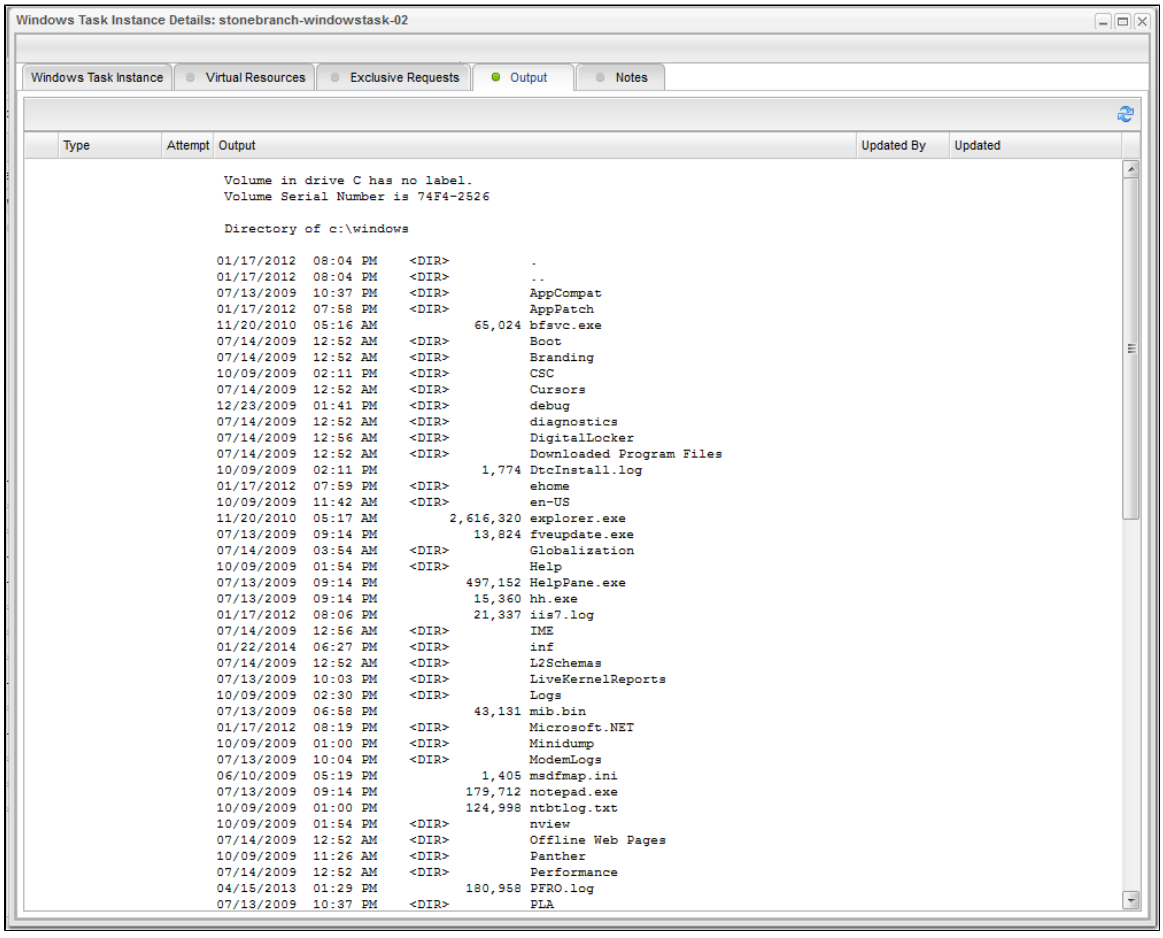
In this Windows task, we will run a DIR command. Normally, you would use the Automatic Output Retrieval field to specify that any output generated by the command be attached to the task instance after the task completes. However, if you did not specify that output be attached, you can manually retrieve it after the task instance has run. In this exercise, we will manually retrieve and display the output.

Step 1	Use the same steps to create another Windows task called **stonebranch-windowstask-02**. Do not specify Automatic Output Retrieval, and use the following command: `dir c:\windows`
Step 2	Save and launch the task.
Step 3	Locate and open the task instance on the Activity Monitor.

Step 4	Note that the Output tab is empty. Click **Retrieve Output**. The Retrieve Output window appears:



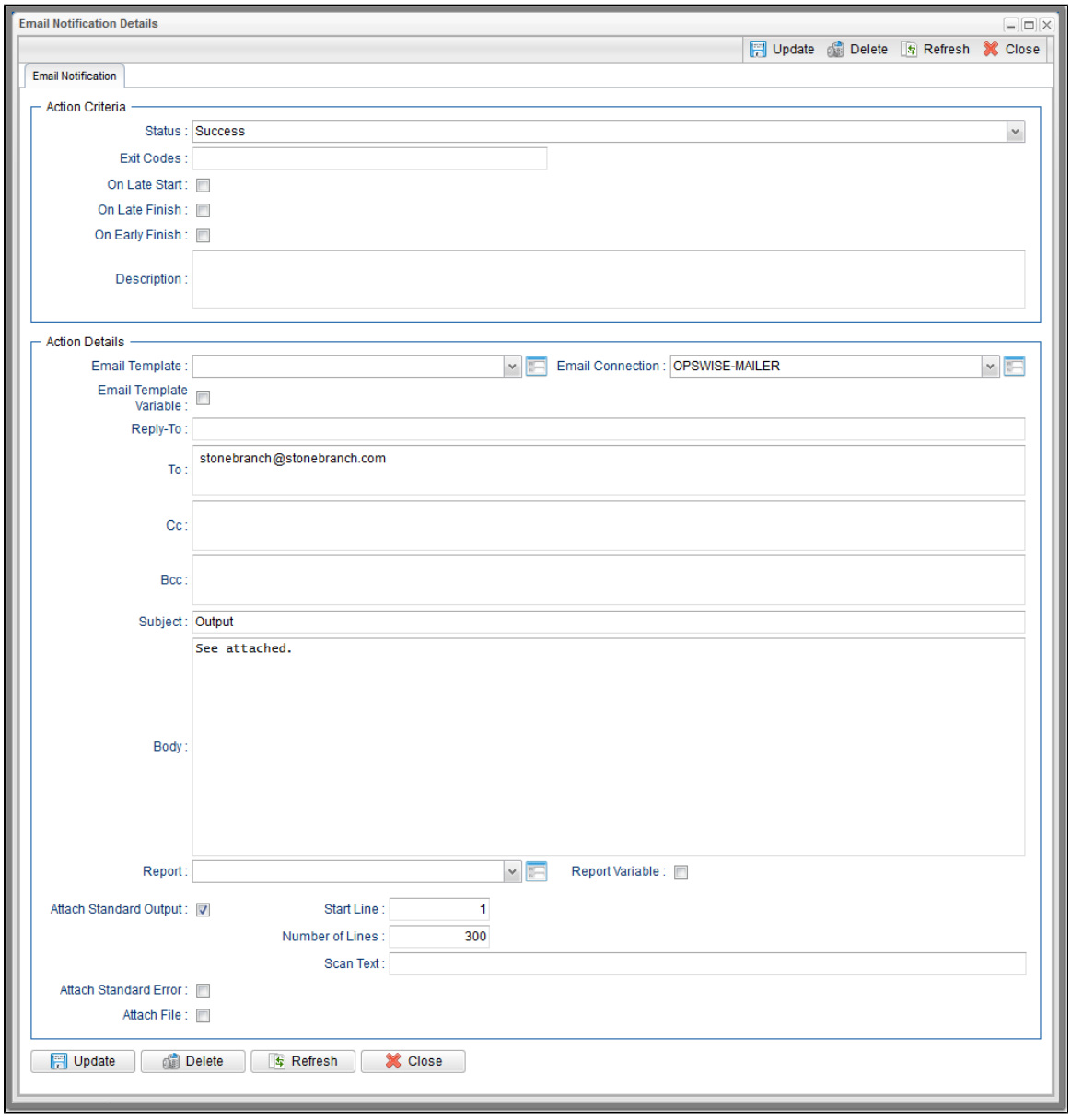
Step 5	Change the number of lines to 300 and click **Submit**.
Step 6	Click the Output tab. The STDOUT (standard output) displays on the output list, as shown in the following example:



Attaching Output to an Email Notification

In this exercise, we will modify the **stonebranch-windowstask-02** task with an Email notification that includes the output from the DIR command.

Step 1	Open the **stonebranch-windowstask-02** task.

Step 2	Click the **Actions** tab.
Step 3	Click **Email Notification** and then click the **New** button.
Step 4	In the Email Notification Details, specify the following: • Status=Success • Email Connection=Your email connection • To=Your email address • Subject=Output • Body=See attached. • Attach Standard Output=enabled • Start Line=1 • Number of lines=300 
Step 5	Click **Save**.
Step 6	Launch the **stonebranch-windowstask-02** task.

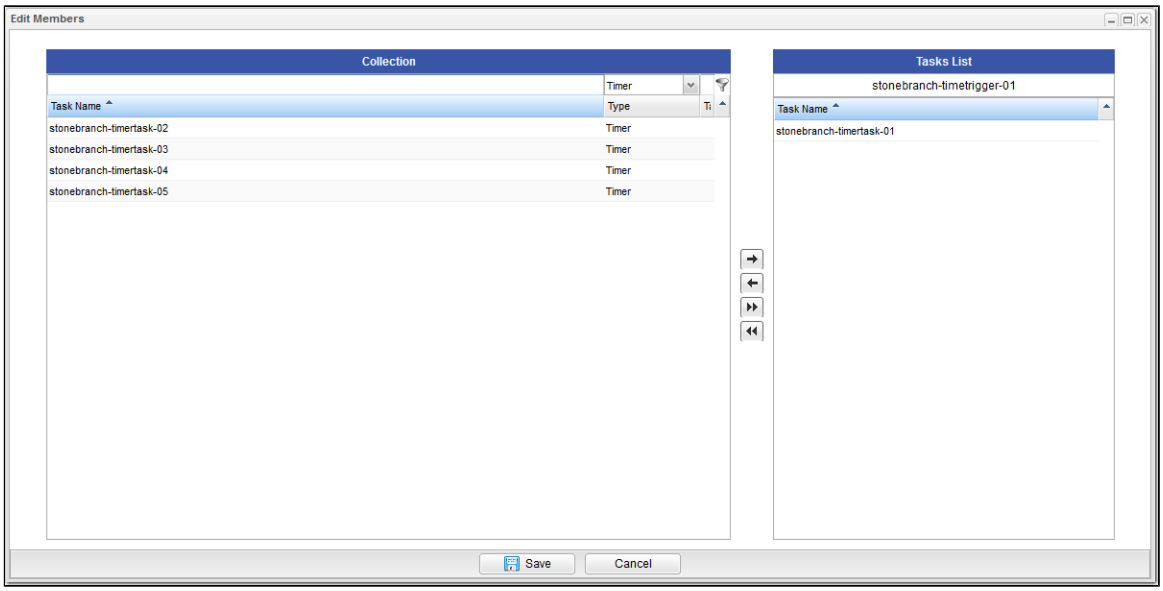
Step 7	Once the Windows task goes to success, check your inbox for the email. The output is attached, as shown in the following example.

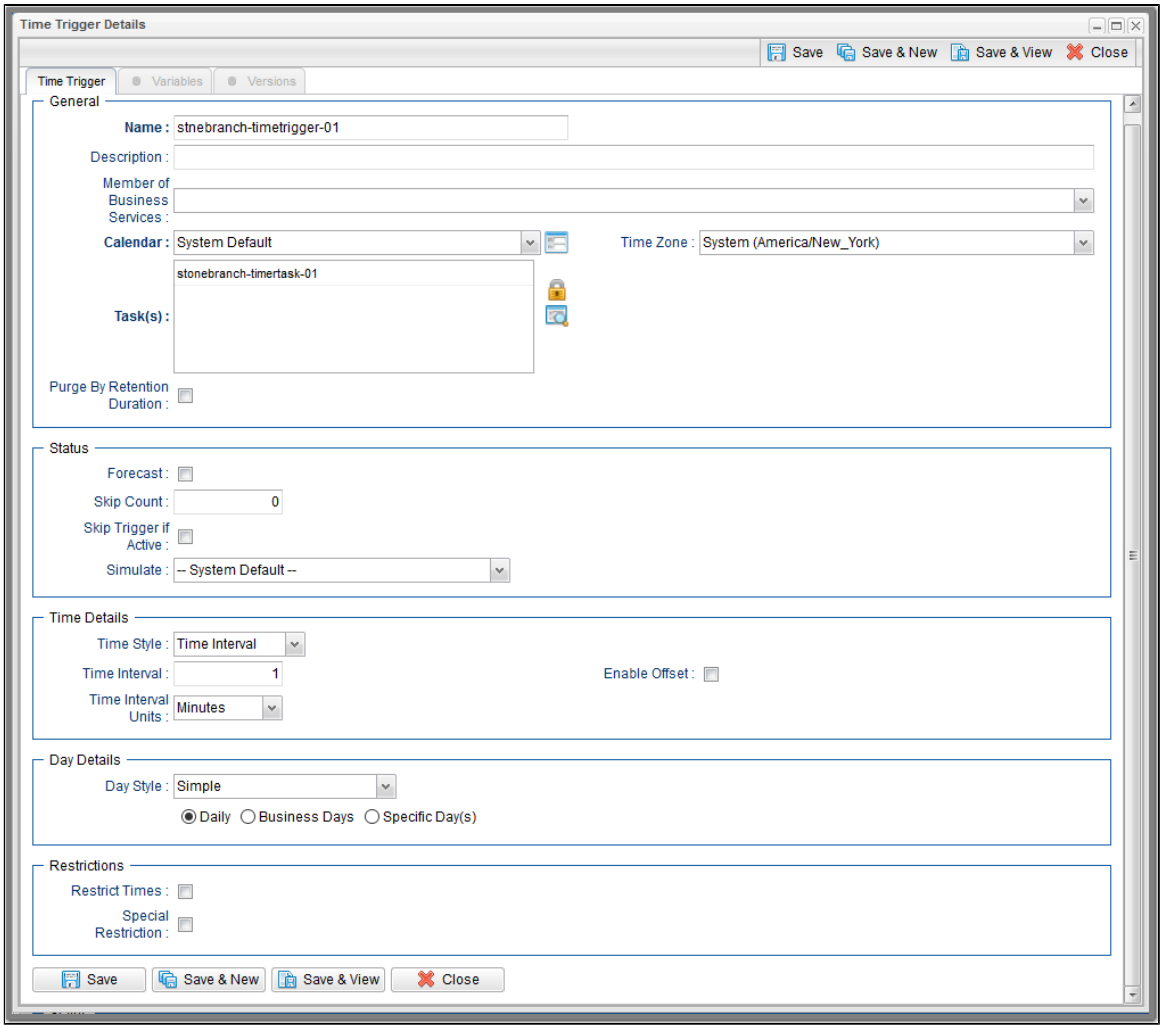
For additional information, see:

- Creating Tasks
- Windows Task

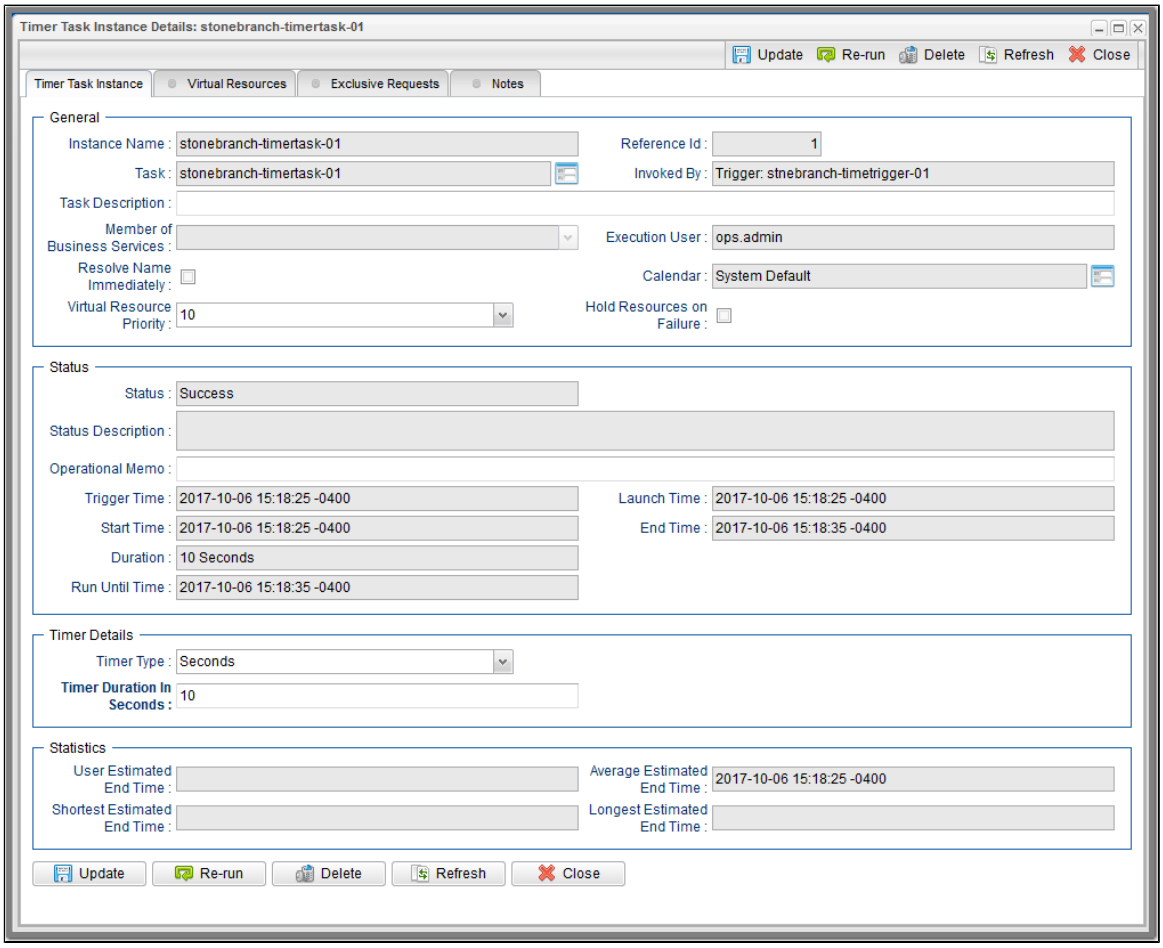
Tutorial - Launching a Task Automatically Using a Simple Time Trigger

In this exercise, we will define a time trigger that launches our Timer task every one minute.

Step 1	From the Automation Center navigation pane, select **Triggers > Time Triggers**.
Step 2	Click **New**.
Step 3	In the **Name** field, enter **stonebranch-timetrigger-01**.
Step 4	In the **Task(s)** field, click the **Add-Remove Multiple** icon.
Step 5	In the Collection window, locate the Timer task created in the Creating and Manually Launching a Task tutorial, **stonebranch-timertask-01**, move it to the Tasks List window, and click **Save**. 
Step 6	In the **Task(s)** field, click the **lock** icon.
Step 7	In the **Time Style** field, select **Time Interval**.
Step 8	In the **Time Interval** field, enter **1**.

Step 9	In the **Time Interval Units** field, select **Minutes**. 
Step 10	Click **Save**.
Step 11	By default, triggers are disabled. To enable this trigger: 1. On the Time Triggers list, right-click **stonebranch-timetrigger-01** to display an Action menu. 2. Click **Enable** to enable the trigger. Note that the **Enabled** column on the trigger list now displays a green check-mark for this trigger.
Step 12	From the **Automation Center** navigation pane, select **Activity** to display the Activity Monitor. Note that a new instance of **stonebranch-timertask-01** appears every one minute.

Step 13 Click the most recent instance of **stonebranch-timertask-01** to view its details, and note that the **Invoked By** field contains the name of the trigger that launched this task.



For additional information, see:

- Triggers
- Time Trigger
- Enabling and Disabling Triggers

Tutorial - Launching a Task Every Monday Except Holidays

- Introduction
- Create Calendar and Custom Days
- Create a Time Trigger
- Adding a Complex Restriction

Introduction

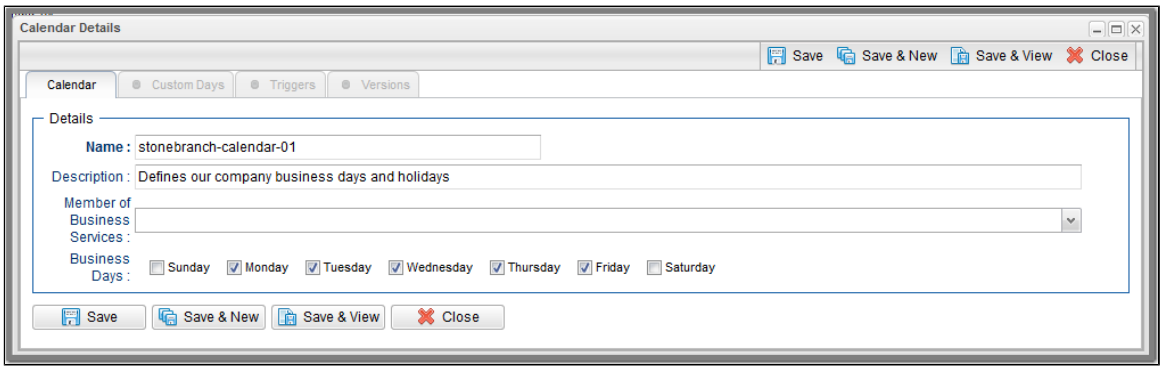
In this exercise, we will define a trigger that runs the **stonebranch-timertask-01** task (created in the Creating and Manually Launching a Simple Task tutorial) automatically every Monday at 1 p.m., except holidays.

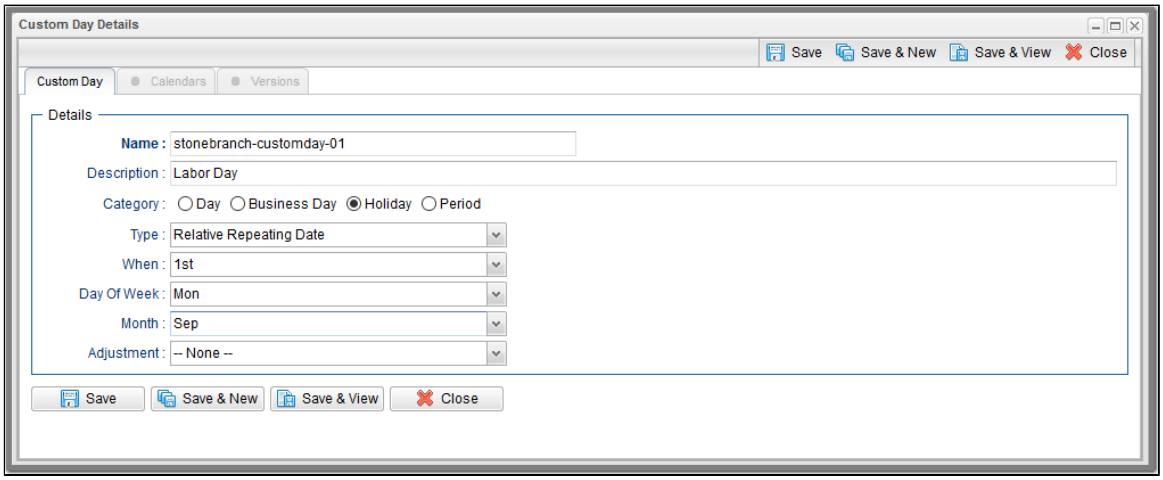
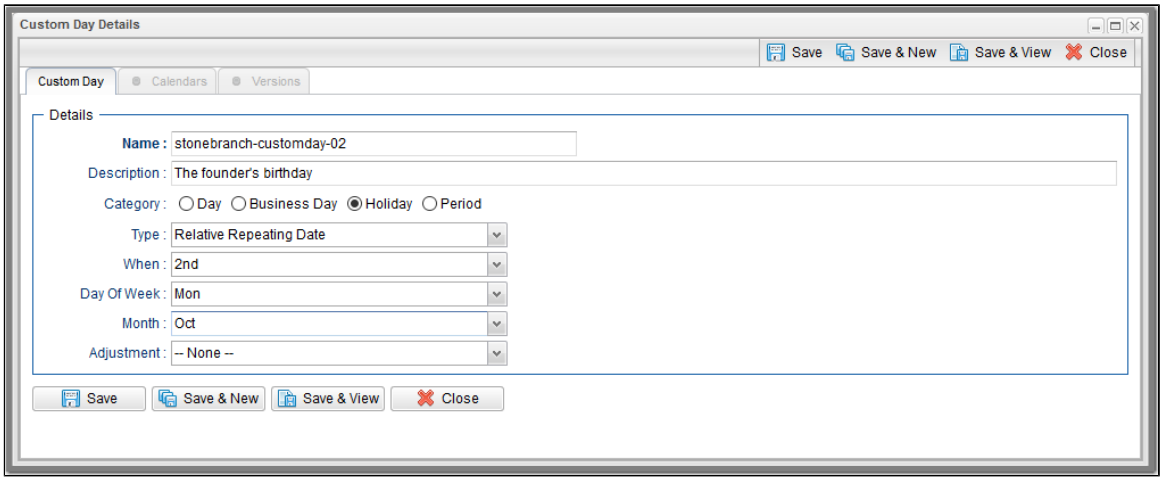
For cases where Monday falls on a holiday, we will define a special restriction in the trigger that instructs the Controller to run the task on the next business day.

We will define the business days and holidays in a calendar.

Create Calendar and Custom Days

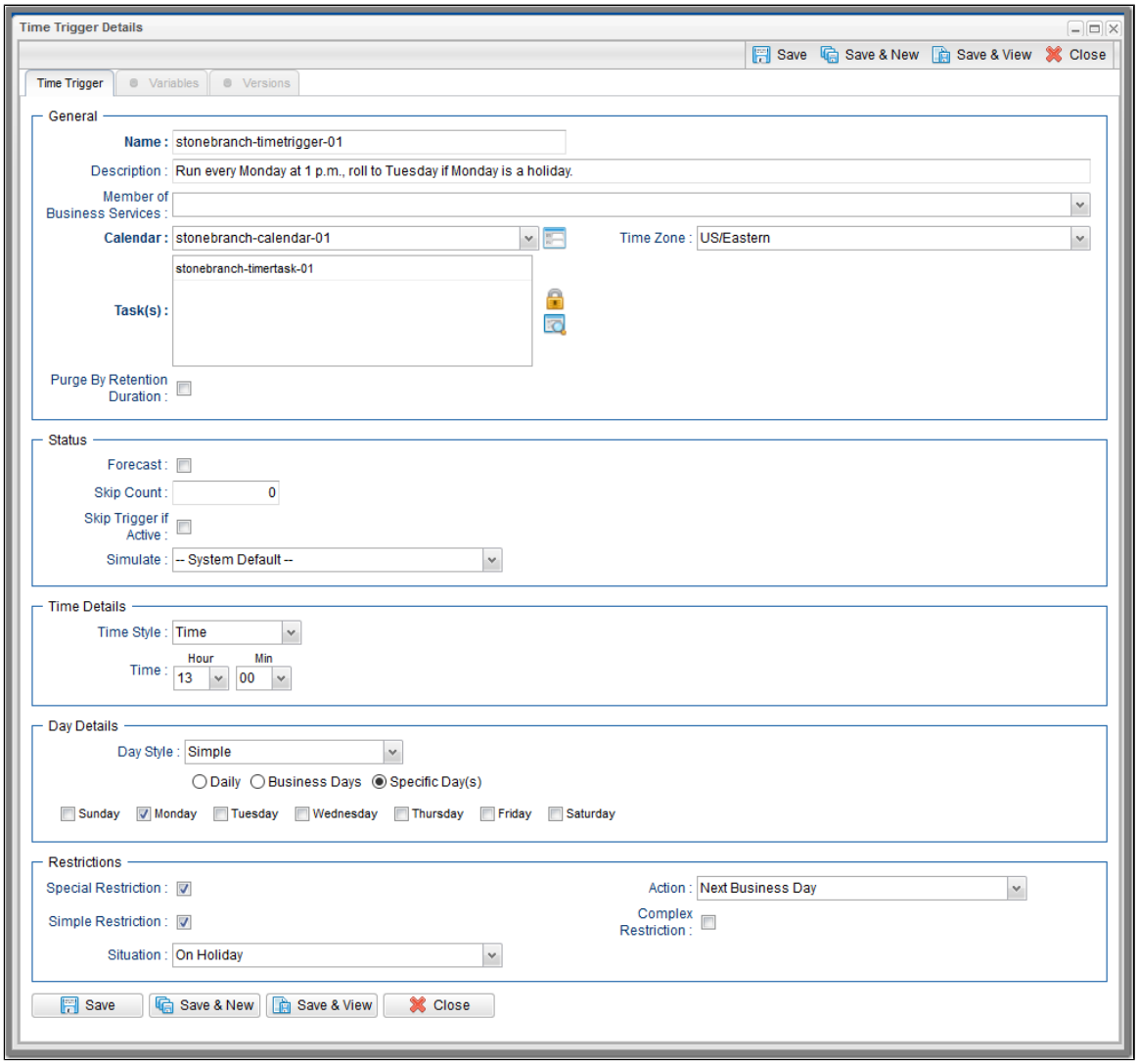
Before building the trigger, we will create the calendar:

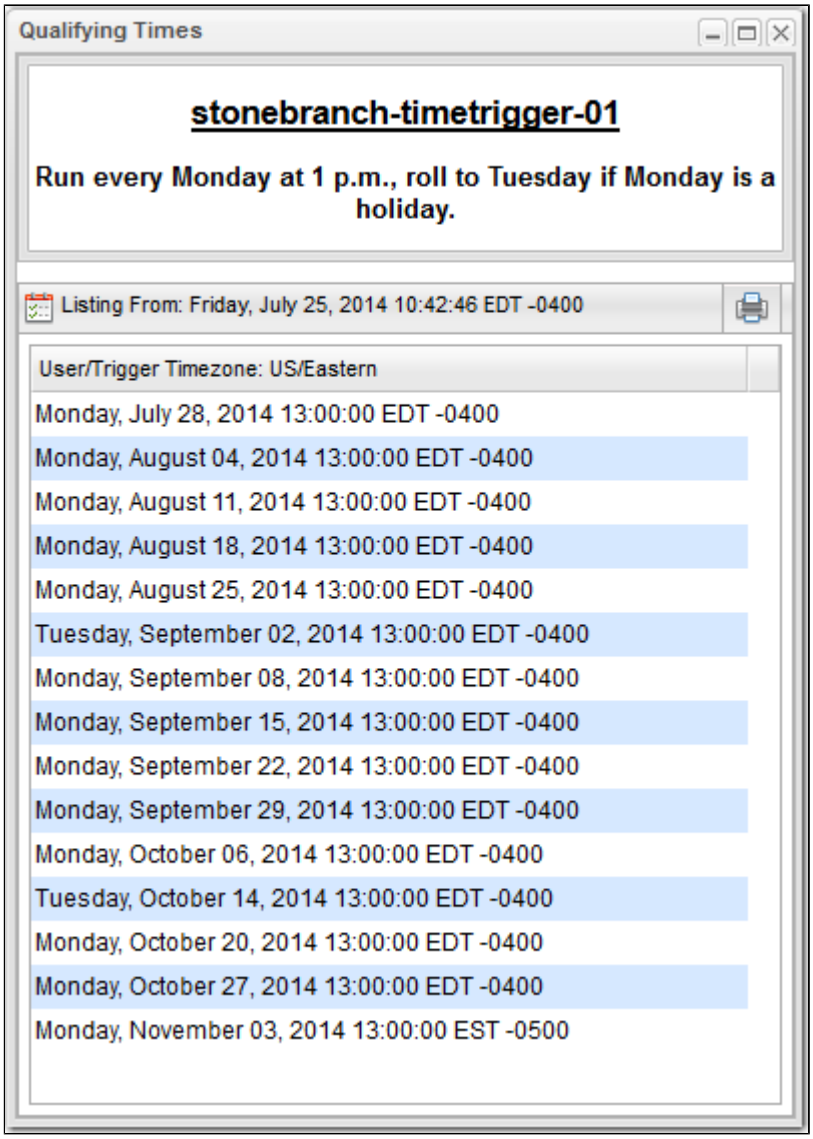
Step 1	From the Automation Center navigation pane, **Others > Calendars** to display the Calendars list.
Step 2	Click **New** to display an empty Calendar Details.
Step 3	Enter **stonebranch-calendar-01** in the **Name** field.
Step 4	In the **Description** field, enter **Defines our company business days and holidays**.
Step 5	Do not change the default selections for **Business Days** (Monday through Friday). Click **Save**. 
Step 6	Open **stonebranch-calendar-01** and click the **Custom Days** tab.
Step 7	On the **Custom Days** list, click the **New** button. A Custom Day Details for a new Custom Day displays.
Step 8	In the **Name** field, enter **stonebranch-customday-01**.
Step 9	In the **Description** field, enter **Labor Day**.
Step 9	In the **Category** field, select **Holiday**.

Step 10	In the **Type** field, select **Relative Repeating Date**. In the three additional fields that appear, make the following selections: • When = 1st • Day of Week = Mon • Month = Sep 
Step 11	Click **Save** to add this Custom Day to the Calendar.
Step 12	On the **Custom Days** list, click the **New** button. A Custom Day Details for a new Custom Day displays.
Step 13	In the **Name** field, enter **stonebranch-customday-02**.
Step 14	In the **Description** field, enter **The founder's birthday.**
Step 15	In the **Category** field, select **Holiday**.
Step 16	In the **Type** field, select **Relative Repeating Date**. In the three additional fields that appear, make the following selections: • When = 2nd • Day of Week = Mon • Month = Oct 
Step 17	Click **Save** to add this Custom Day to the Calendar.

Create a Time Trigger

Step 1	From the **Automation Center** navigation pane, select **Triggers > Time Trigger** to display the Time Triggers list.

Step 2	Click **New** to display an empty Time Trigger Details.
Step 3	In the **Name** field, enter **stonebranch-timetrigger-01**.
Step 4	In the **Description** field, enter **Run every Monday at 1 p.m., roll to Tuesday if Monday is a holiday**.
Step 5	In the **Calendar** field, select **stonebranch-calendar-01**.
Step 6	In the **Task(s)** field, select **stonebranch-timertask-01**.
Step 7	In the **Time Style** field, keep the default, **Time**.
Step 8	In the **Time** field, enter **13:00** (1 p.m. in 24-hour time).
Step 9	In the **Day Style** field, keep the default, **Simple**.
Step 10	Enable **Specific Day(s)** field and select **Monday**.
Step 11	Enable **Special Restriction** and select: • **On Holiday** in the **Situation** field. • **Next Business Day** in the **Action** field.
Step 12	Click **Save**. 
Step 13	Click the **List Qualifying Times** button to display the *List Qualifying Times Input pop-up dialog.
Step 14	In the **Number of Dates/Times** field, enter **15**.

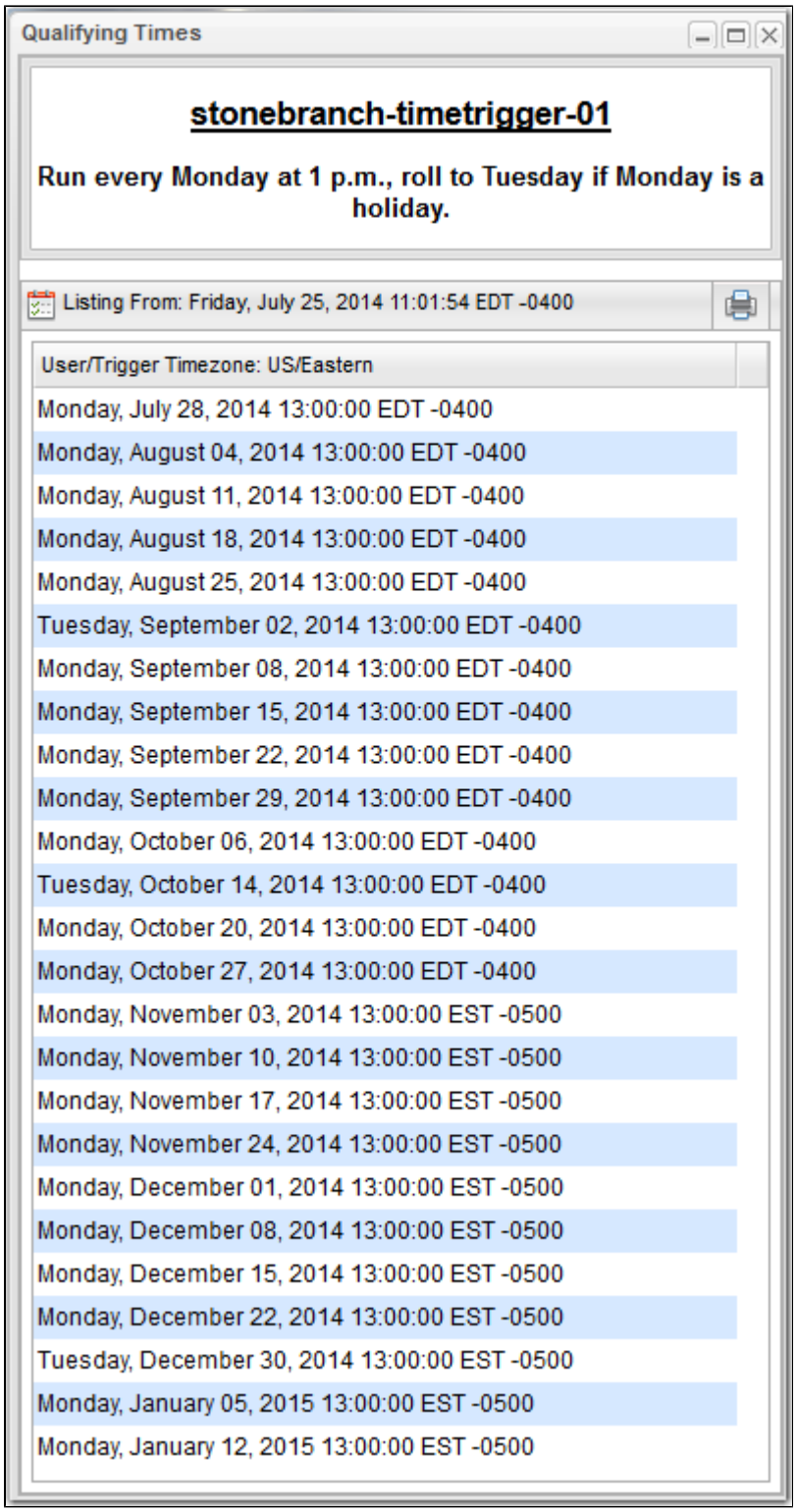
Step 15	In the **Start Date** fields, Select **July 25 2014**.
Step 16	Click **Submit** to display a list of **Qualifying Times** when the trigger will launch the task.  As shown in the list, the task will not run on the two Mondays that you have defined as holidays in the calendar, but instead will run the the following Tuesday.

Adding a Complex Restriction

The following steps show you how to add a complex restriction to the **stonebranch-timetrigger-01** trigger. In this case, you will add a restriction that skips the last Monday of the year and instead triggers the task on the following Tuesday, just as it does for Mondays that are holidays.

Step 1	Enable **Complex Restriction**.
Step 2	In the **Restriction Mode** field, select **Or**.
Step 3	In the **Restriction Adjective** field, select **Last**.
Step 4	In the **Restriction Noun** field, select **Monday**.
Step 5	In the **Restriction Qualifier** field, select **Year**.
Step 6	Click **Update**.

Step 7 Re-open the trigger, click the **List Qualifying Times** button, enter **25** in the **Number of Dates/Times** field, and click **Submit**. The **Qualifying Times** list now shows an additional Monday (the last Monday of the year) on which the task will not be run.



For additional information, see:

- Triggers
- Time Trigger
- Calendars
- Displaying Trigger Forecast Information

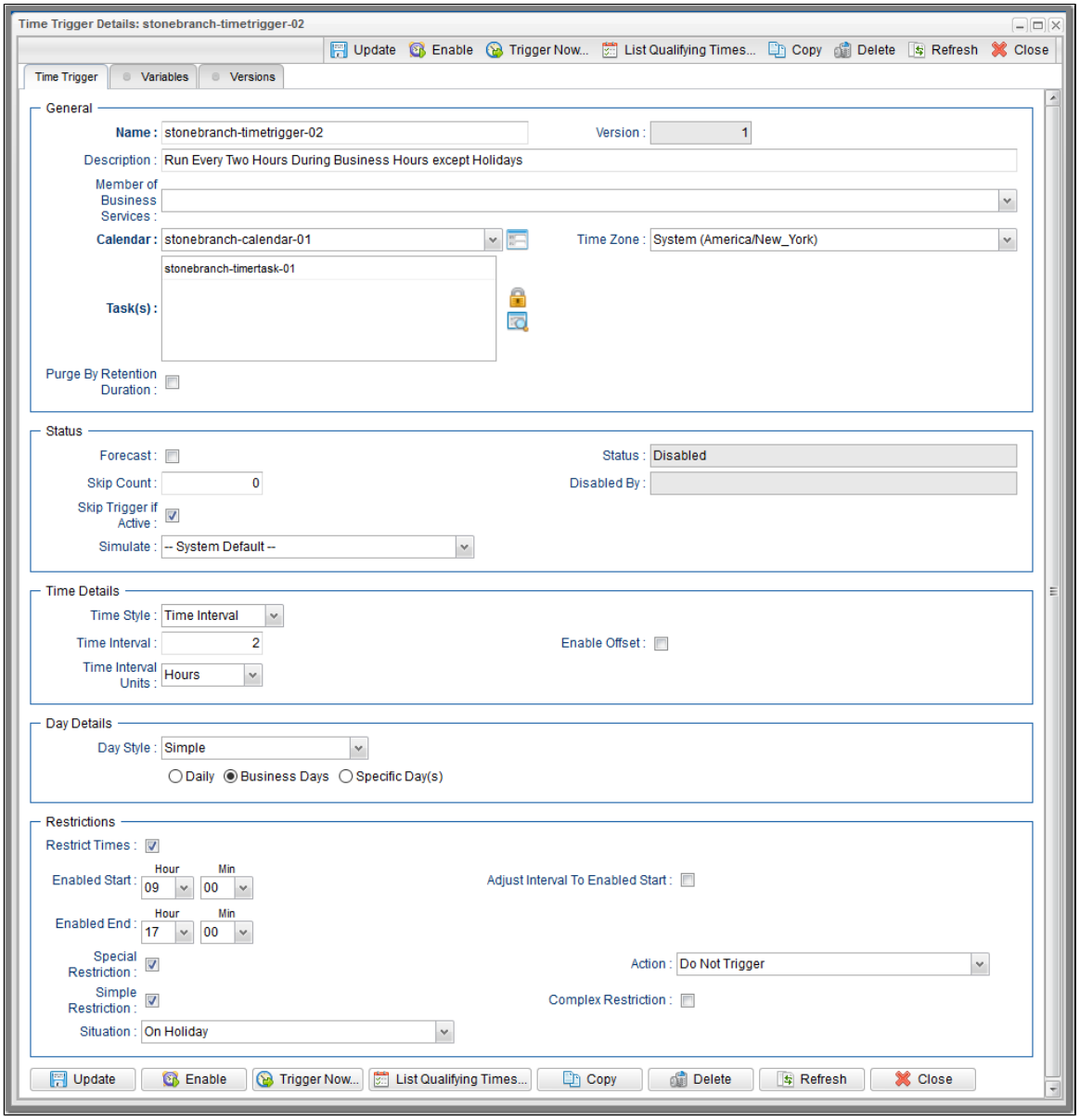
Tutorial - Launching a Task Every Two Hours During Workday

In this exercise, we will define a recurring task that runs every two hours, limited to business hours – Monday through Friday from 9 a.m. to 5 p.m. We will also instruct Universal Controller not to run the task on holidays.

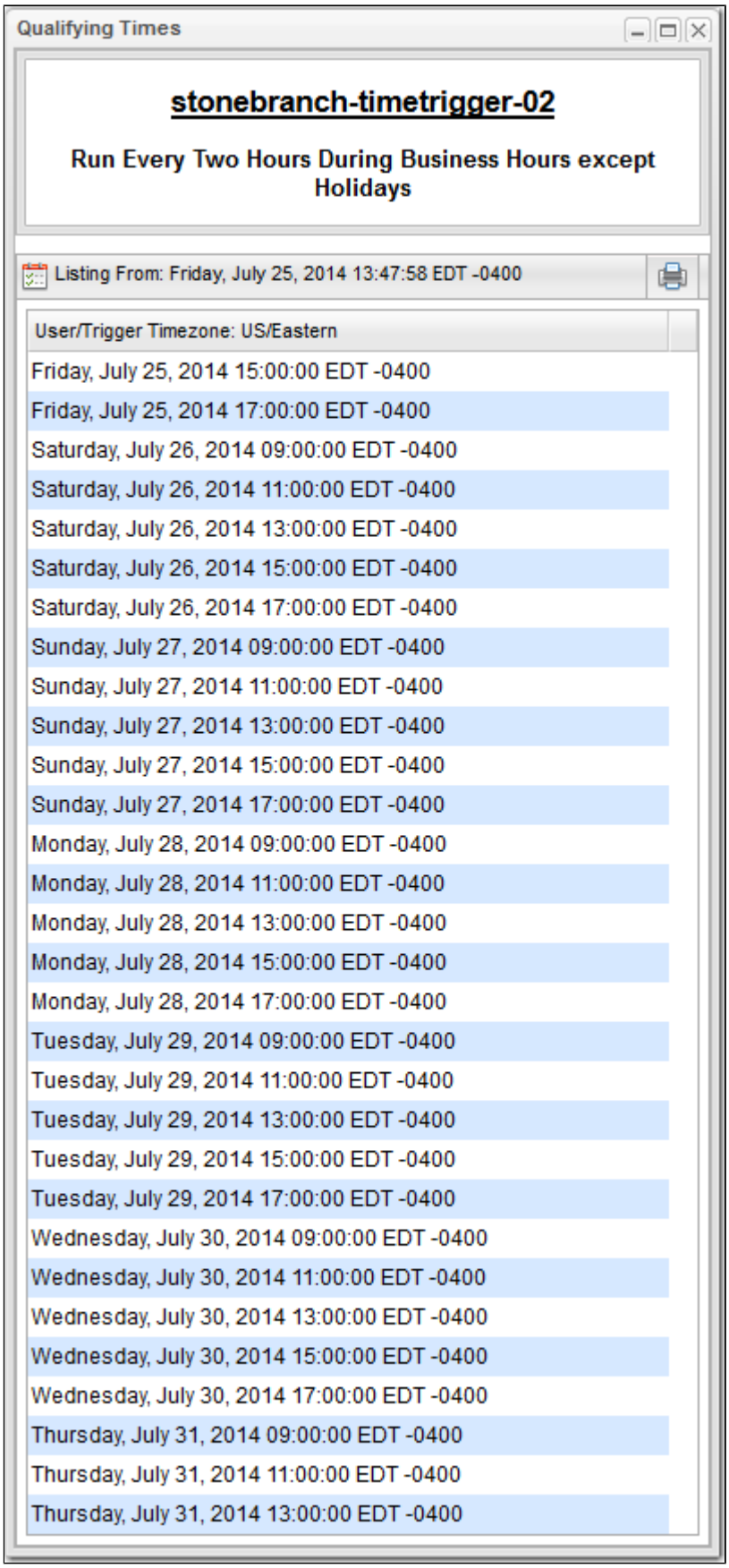
Also, if the task is still running two hours later when it is time to run the next task instance, the Controller will be instructed not to run the next instance.

Step 1	From the Automation Center navigation pane, select **Triggers > Time Triggers**. The Triggers list displays.
Step 2	Click the **New** button to display an empty Time Trigger Details.
Step 3	In the **Name** field, enter **stonebranch-timetrigger-02**.
Step 4	In the **Description** field, enter **Run Every Two Hours During Business Hours Except Holidays**.
Step 5	In the **Calendar** field, select **stonebranch-calendar-01** (created in the Launching a Task Every Monday Except Holidays tutorial).
Step 6	In the **Task(s)** field, select **stonebranch-timertask-01** (created in the Creating and Manually Launching a Simple Task tutorial).
Step 7	Enable the **Skip Trigger if Active** field. This tells the Controller not to trigger the task if the previous instance of the task is still active.
Step 8	In the **Time Style** field, select **Time Interval** and specify the following: • **Time Interval** = 2 • **Time Interval Units** = Hours
Step 9	Enable the **Restrict Times** field and specify the following: • **Enabled Start** = 09:00 • **Enabled End** = 17:00
Step 10	Specify that this trigger should run on weekdays only by selecting either: • Day Style = Simple • Business Days = Enabled OR • Day Style = Complex • Date Adjective = Every • Date Noun = Business Day • Date Qualifier = Year Both methods use the Business Days specified in **stonebranch-calendar-01**.
Step 11	Enable the **Special Restriction** field and specify the following: • **Situation** = On Holiday • **Action** = Do Not Trigger

Step 12 Click the **Save** button.



Step 13 Re-open **stonebranch-timetrigger-02** and click the **List Qualifying Times** button. The **Qualifying Times** list displays the next scheduled 30 (by default) times when **stonebranch-timetrigger-02** will launch **stonebranch-timertask-01**.



For additional information, see:

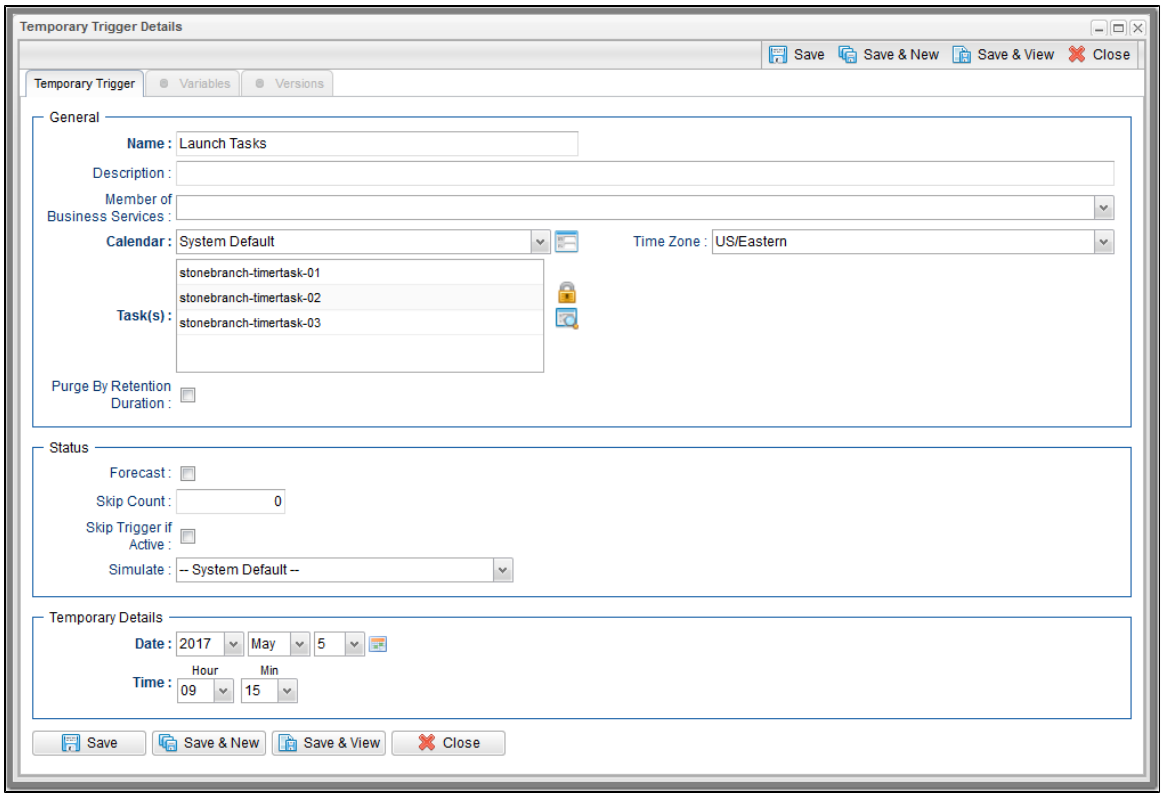
- Triggers
- Time Trigger
- Calendars

Tutorial - Launching Tasks at a Future Time

In this exercise, we will create a trigger that will launch multiple tasks at the same time in the future (in two minutes). For this exercise, we will use the SQL tasks created in the Running a Workflow with a Conditional Path tutorial.

(A future date also can be selected, but to see now that the tasks have been launched, keep the current date.)

Create a Temporary Trigger

Step 1	From the Automation Center navigation pane, select **Triggers > Temporary Triggers**. The Temporary Triggers list displays.
Step 2	Click the **New** button to display Temporary Trigger Details for a new trigger and enter/select the following values: • **Name** = Launch Tasks • **Tasks** = stonebranch-timertask-01, stonebranch-timertask-02, and stonebranch-timertask-03 Timer tasks • **Date** = current date • **Time** = 5 minutes from the current time • **Time Zone** = your time zone
Step 3	Click the **Save** button. 
Step 4	Right-click **Launch Tasks** on the Temporary Triggers list and click **Enable**.
Step 5	Open the Activity Monitor to see that the three Timer tasks are run at the selected time.

For additional information, see:

- Temporary Trigger

Tutorial - Launching an Email Task Based on a File Monitor

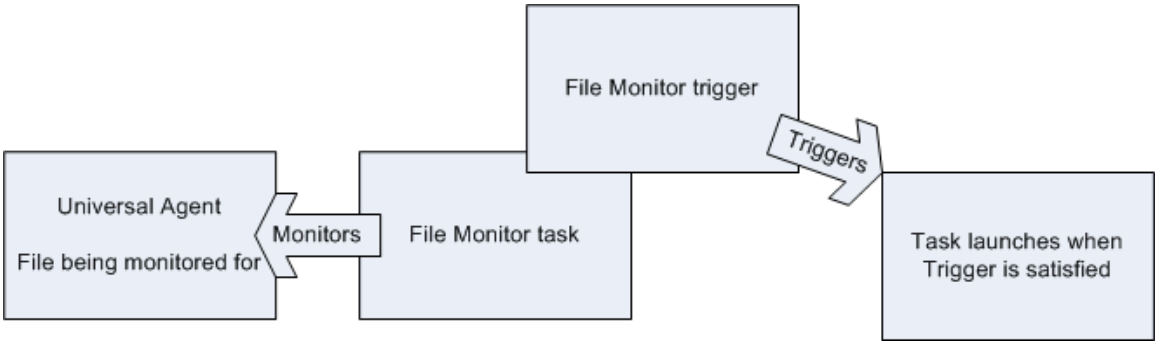
- Introduction
- Create File Monitor Task
- Create an Email Task
- Create File Monitor Trigger
- Test Your File Monitor Set-up

Introduction

In this exercise, we will monitor a machine for a specific file. When the file appears, we will send an email that uses variables to supply information about when and how the email was sent.

In order to set this up, we need the following:

- Universal Agent
- File Monitor task
- File Monitor trigger
- Email task being launched by the trigger.



Prerequisites:

- In order to perform this exercise, you need a Windows, Linux/Unix, or z/OS Agent running on the machine that is being monitored for the file. Create a directory on the machine called **`controller tutorial`**. Later on, you will copy a text file to this directory to satisfy the File Monitor trigger.

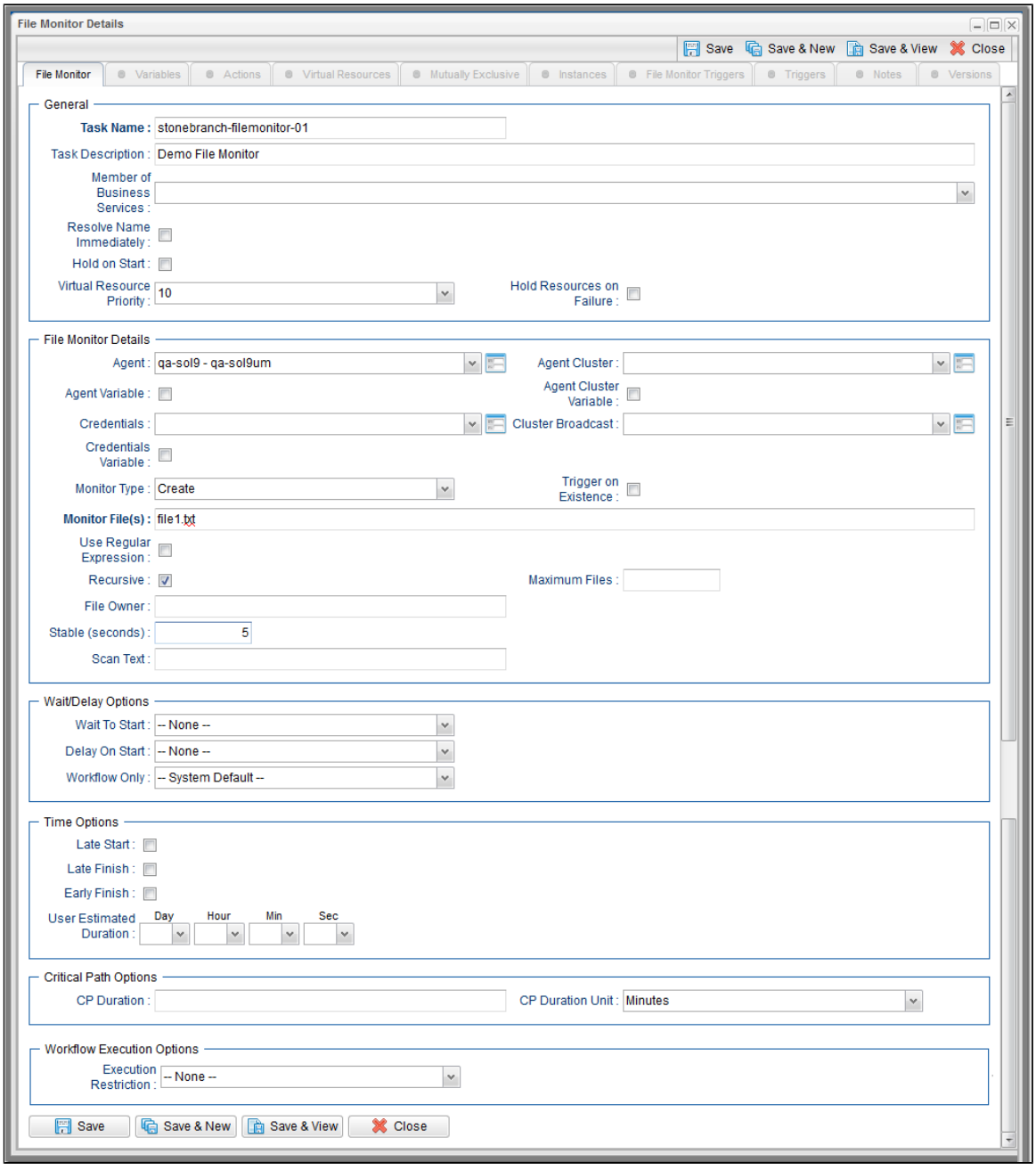
> **Note**
> If you do not have a running Agent, you can bypass this step by manually satisfying the trigger, as per instructions in the tutorial. However, you do need an Agent defined in the database.

- Since we are going to generate an email when the trigger is satisfied, you will need an Email Connection defined.

Create File Monitor Task

The File Monitor task monitors the agent machine for the specified file.

Step 1	From the Automation Center navigation pane, select **Tasks > File Monitors**. The File Monitors list displays.
Step 2	Click the **New** button to display an empty File Monitor Details.
Step 3	In the **Task Name** field, enter **stonebranch-filemonitor-01**.
Step 4	In the **Task Description** field, enter **Demo File Monitor**.
Step 5	In the **Agent** field, select an Agent.

Step 6	In the **Monitor Type** field, keep the default value, **Create**. (See File Monitor Task Details Field Descriptions for details about the other file monitor options.)
Step 7	In the **Monitor File** field, type **file1.txt**. Since we have not specified any directory, the Controller will search the root directory.
Step 8	Enable the **Recursive** field. Since we are going to write our file to the **`controller tutorial`** directory, we want the Controller to search all sub-directories for the file.
Step 9	In the **Stable (seconds)** field, enter **5**. This tells the Controller to satisfy the trigger only when the file has not changed in 5 seconds. 
Step 10	Click **Save**.

Create an Email Task

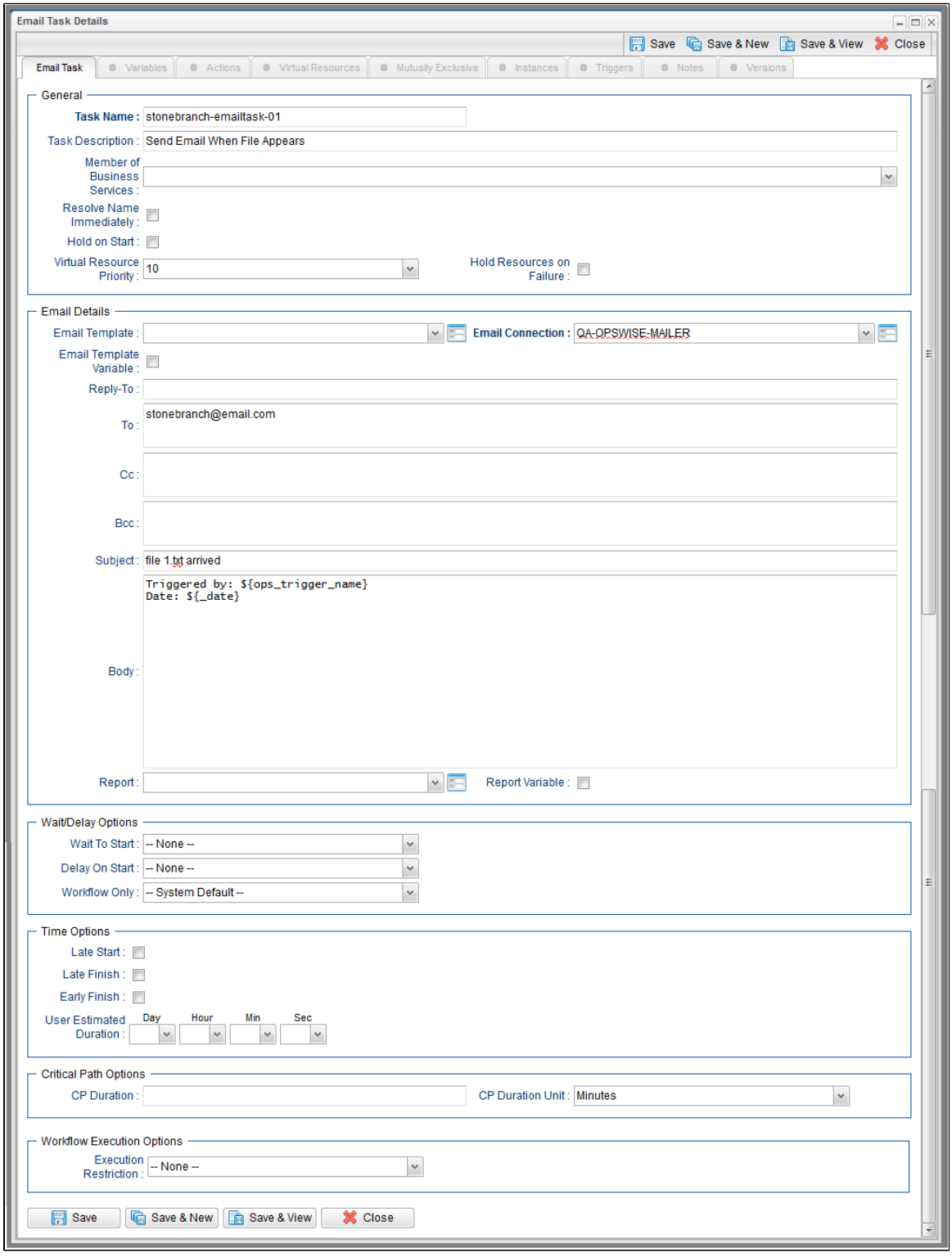
Create the task that will run when the File Monitor is satisfied. In this case, we will **generate an email, using the Email task:**

Step 1	From the **Automation Center** navigation pane, select **Tasks > Email Tasks**. The Email Tasks list displays.

Step 2	Click the **New** button to display an empty Email task Details.
Step 3	In the **Task Name** field, enter **stonebranch-emailtask-01**.
Step 4	In the **Description** field, enter **Send Email When File Appears**.
Step 5	In the **Email Connection** field, select your Email Connection.
Step 6	In the **To** field, enter your email address. This is where the email will be sent.
Step 7	In the **Subject** field, enter **file1.txt arrived**.

Step 8	In the body field, enter the following Universal Controller variable and Universal Controller function:
	
	`Triggered by: ${ops_trigger_name}`
	`Date: ${_date}`

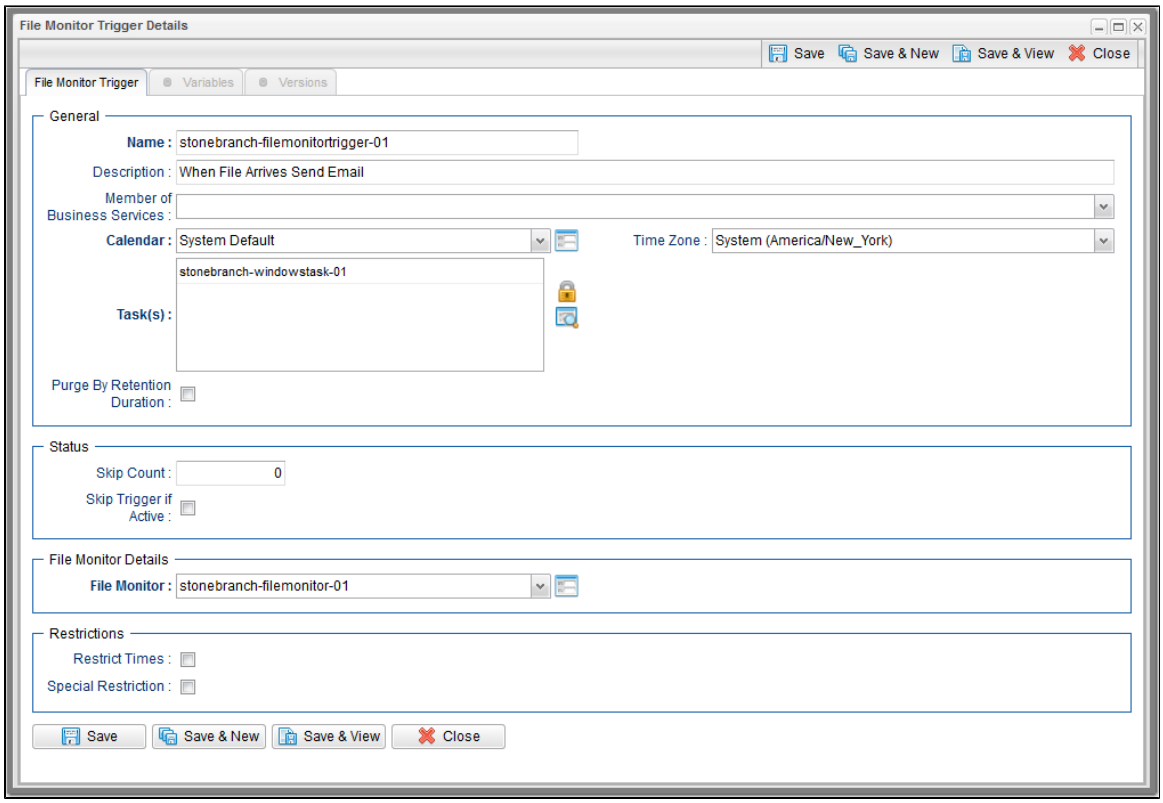
Step 9 When the email is generated, the variables will be substituted with the name of the trigger that launched the Email task, along with the date and time the task was launched.



Step 9 Click the **Save** button.

Create File Monitor Trigger

Create the File Monitor trigger:

Step 1	From the Automation Center* navigation pane, select **Triggers > File Triggers**. The File Monitor Triggers list displays.
Step 2	Click the **New** button to display an empty File Monitor Trigger Details.
Step 3	In the Trigger Name field, enter **stonebranch-filemonitortrigger-01**.
Step 4	In the **Description** field, enter **When File Arrives Send Email**.
Step 5	In the **Task(s)** field, select **stonebranch-emailtask-01**.)
Step 6	In the **File Monitor** field, select **stonebranch-filemonitor-01**.
Step 7	Click the **Save** button. 

Test Your File Monitor Set-up

Test your File Monitor set-up.

Step 1	Enable the File Monitor trigger. This launches the File Monitor task. It will appear in the Activity Monitor with a status of Running.
Step 2	Do one of the following: • If you have a running agent, place a text file called **`file1.txt`** in the **`controller tutorial`** directory on the machine that is being monitored by the File Monitor task. When the file appears, the File Monitor task waits five seconds, as specified, and then satisfies the trigger. • If you do not have a running agent but do have an agent connected to your instance, you can manually satisfy the trigger as follows: 1. From the **Automation Center** navigation pane, select **Triggers > File Triggers** to display the File Monitor Triggers list. 2. Right-click the **stonebranch-file monitortrigger-01** trigger and select **Trigger Now**.
Step 3	When the trigger is satisfied, the Email is sent. Go to the Activity Monitor and note that the **astonebranch-emailtask-01** Email task has been launched.
Step 4	Go to your email account where the email was sent and open the email. Note that the variables were resolved.

For additional information, see:

- Email Task
- File Monitor Task
- File Trigger
- Variables

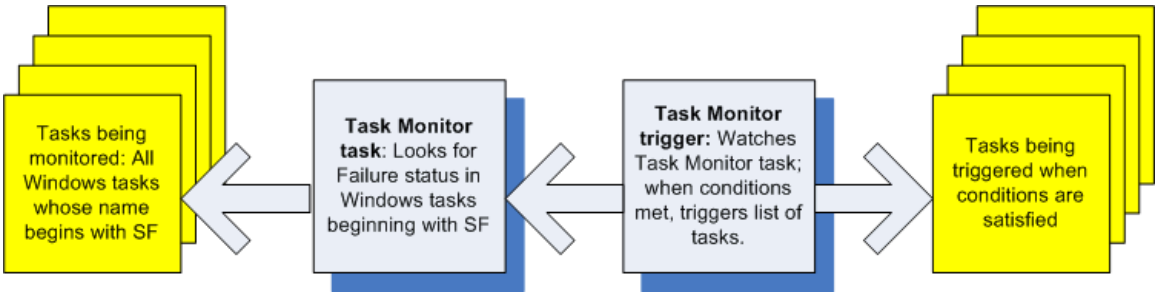
Tutorial - Launching an Email Task Based on a Task Monitor

- Introduction
- Select the Activity Monitor Problem Filter
- Create an Email Template
- Creating an Email Task Using the Email Template
- Creating a Task Monitor Task
- Creating a Task Monitor Trigger
- Running the Task Monitor

Introduction

In this exercise, we will set up a Task Monitor and Task Monitor trigger. The Task Monitor will monitor all tasks for a status that indicates some sort of problem. When the trigger is satisfied, Universal Controller will launch an Email task that notifies a user that there is a problem. We will also create an Email template for use in our Email task and create an Activity Monitor filter that displays only problem tasks.

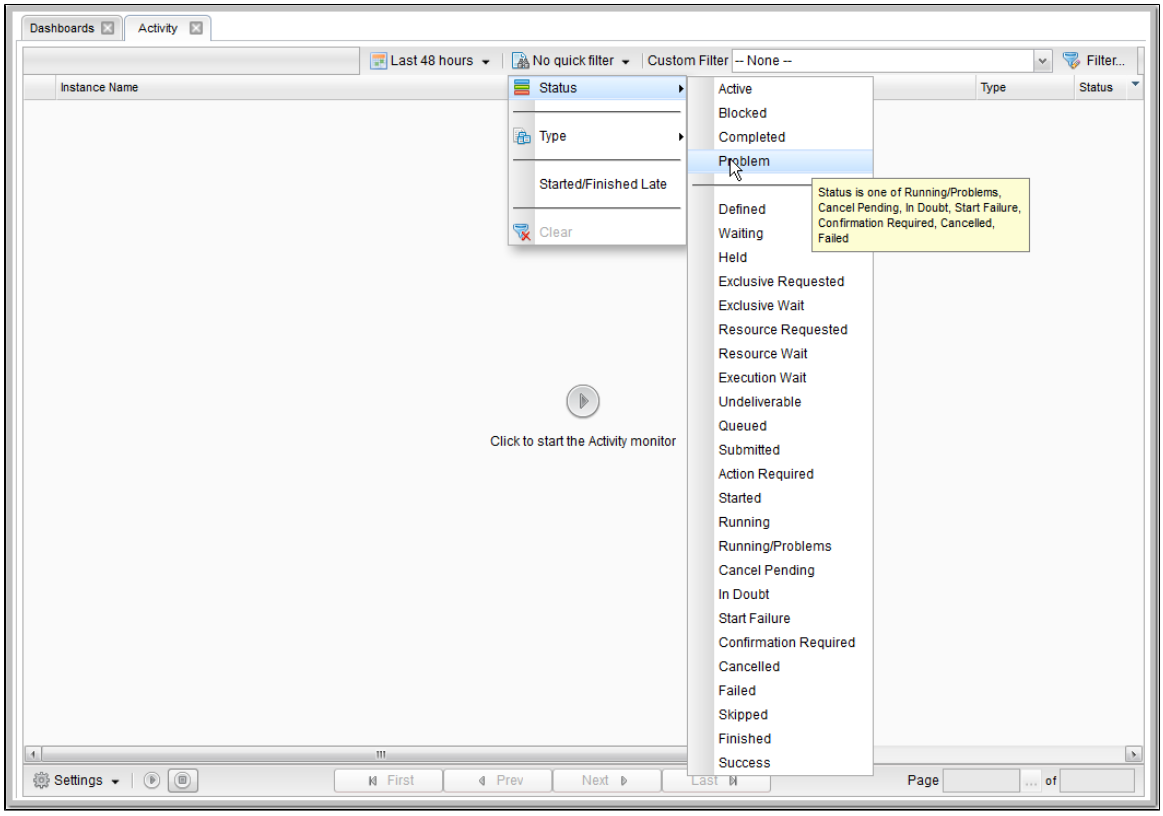
The following illustration shows the various components used to trigger tasks based on the status of other tasks.



Select the Activity Monitor Problem Filter

The first task we will do is select the Problem filter for the Activity Monitor. When the user receives an email indicating there is a problem, the user can view this Activity Monitor to find out more information.

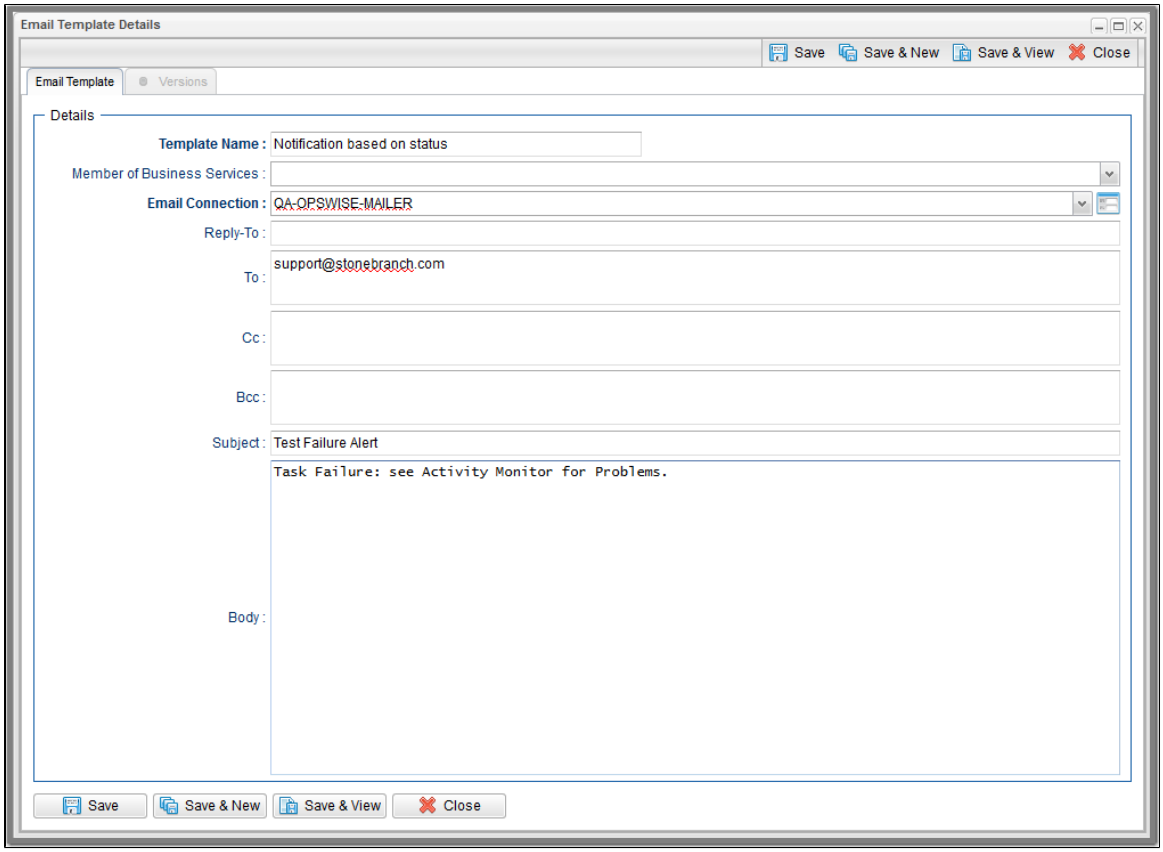
Step 1	From the Automation Center navigation pane, select **Task Instances > Activity**. The Activity Monitor displays.
Step 2	Click the **Click to start the Activity monitor** button. All active task instances display.

Step 3	From the Quick Filter drop-down list, select **Status > Problem**.  The Activity Monitor now will display only Problem task instances.

Create an Email Template

Email Templates allow you to create pre-defined Email task information that you refer to when creating an Email task. This is useful if you have a large number of common parameters on Email tasks but still require separate tasks.

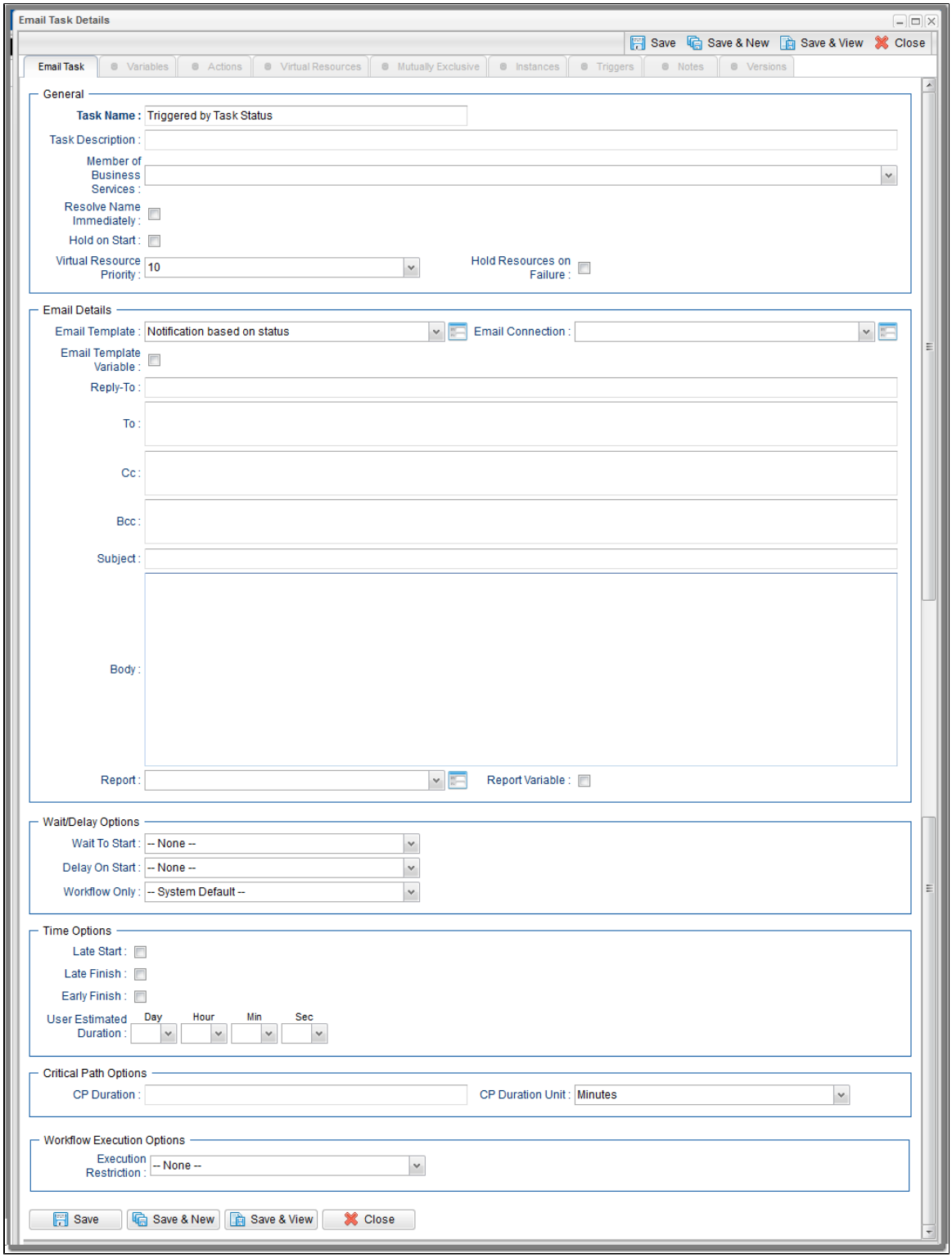
Step 1	From the Agents & Connections navigation pane, select **System > Email Templates**. The Email Templates list displays.
Step 2	Click the **New** button to display an empty Email Template Details and enter the following values: • **Template Name** = Notification based on status • **Email Connection** = (a valid email connection) • **To** = (a valid email account) • **Subject** = Task Failure Alert • **Body** = Task failure, see Activity Monitor for Problems

Step 3	Click the **Save** button. 

Creating an Email Task Using the Email Template

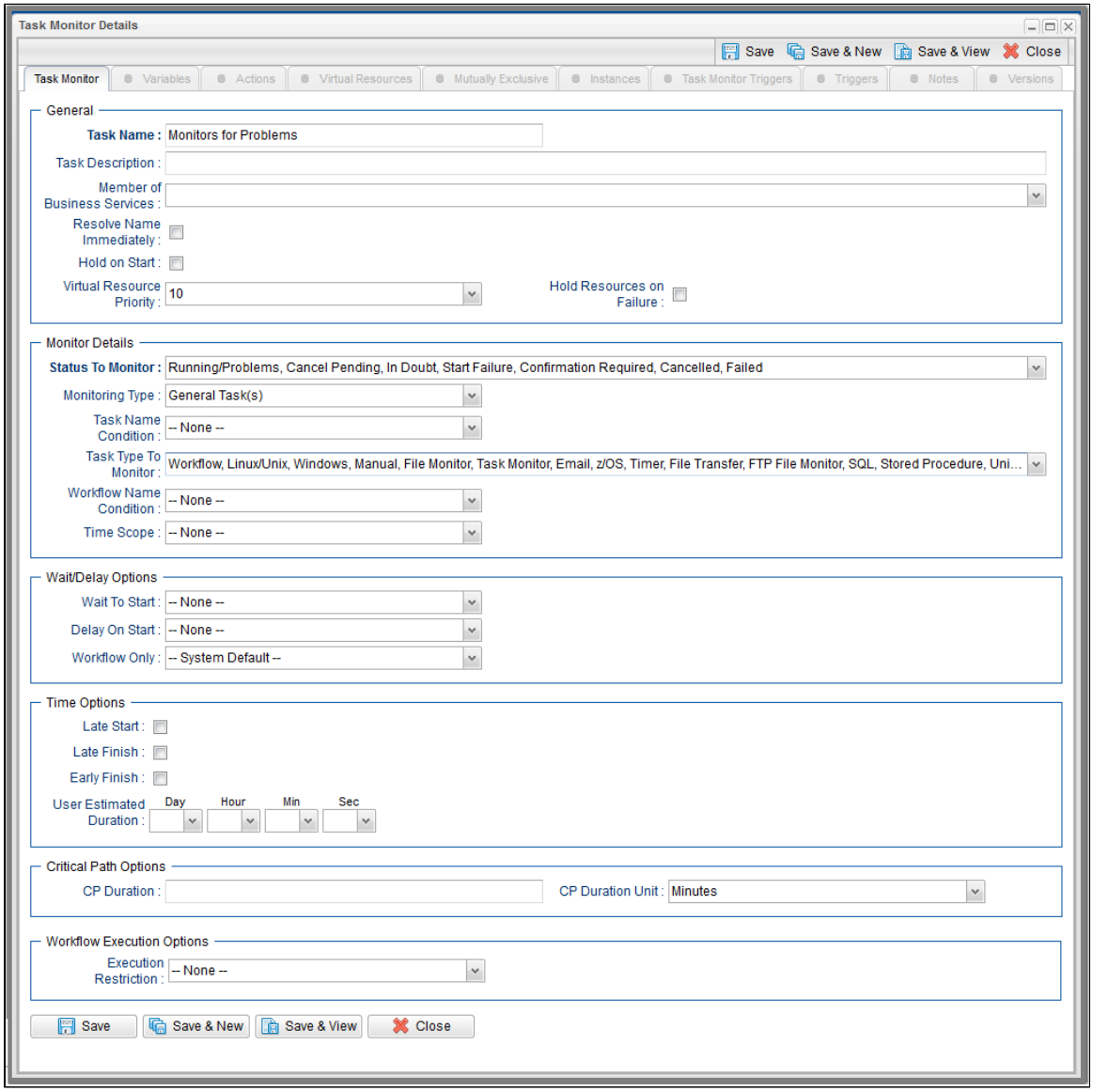
Step 1	From the Automation Center navigation pane, select **Tasks > Email Tasks**. The Email Tasks list displays.
Step 2	Click the **New** button to display an empty Email Task Details and enter the following values: • **Task Name** = Triggered by Task Status • **Email Template** = Notification based on status Leave the remaining fields blank, since we want to use the information from the template. (If you fill in any of the duplicate fields, the information from the task overrides the information from the template.)

Step 3 Click the **Save** button.



Creating a Task Monitor Task

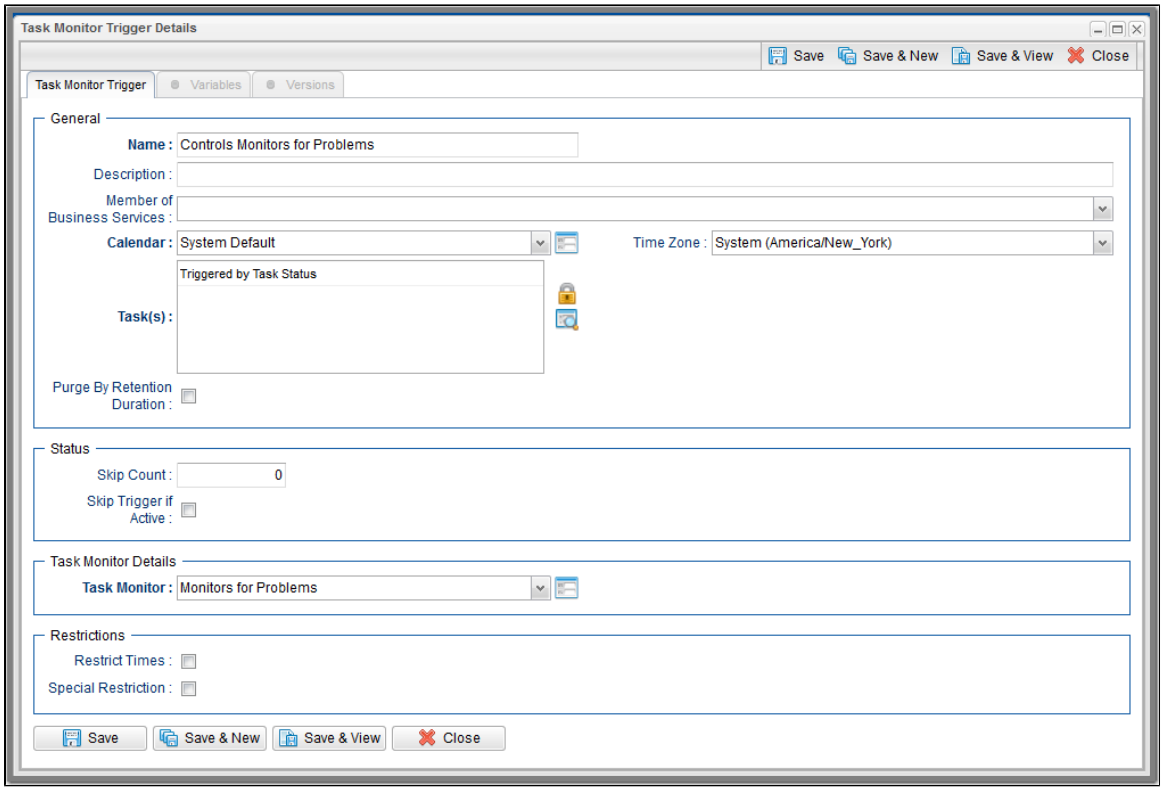
For this Task Monitor, we will monitor the status of all other tasks. If any task has a status that indicates there is some sort of problem, we will generate an email. Once the Task Monitor is launched by the Task Monitor trigger, it remains active, launching an Email every time any of its conditions are met. The Task Monitor task remains active until the Task Monitor trigger is disabled or until a user manually stops it.

Step 1	From the Automation Center navigation pane, select **Tasks > Task Monitors**. The Task Monitors list displays.
Step 2	Click the **New** button to display an empty Task Monitor Details and enter the following values: • **Task Name** = Monitors for Problems • **Status To Monitor** = enable Running/Problems, Cancel Pending, In Doubt, Start Failure, Confirmation Required, Cancelled, and Failed. (For a description of each status, see Displaying Task Instance Status). • **Monitoring Type** = General Task(s) • **Task Type To Monitor** = enable all task types
Step 3	Click the **Save** button. 

Creating a Task Monitor Trigger

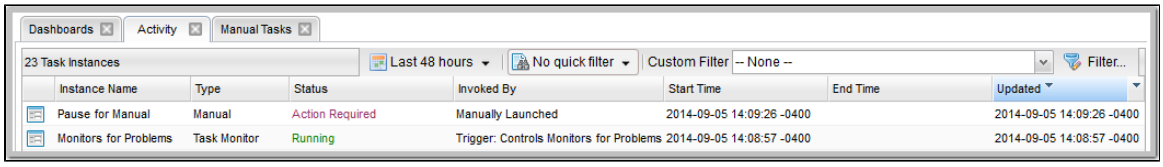
As the last step in our set-up process, we will create the Task Monitor trigger, which controls when the Task Monitor task is started and stopped.

Step 1	From the Automation Center navigation pane, select **Triggers > Task Monitor Triggers**. The Task Monitor Triggers list displays.

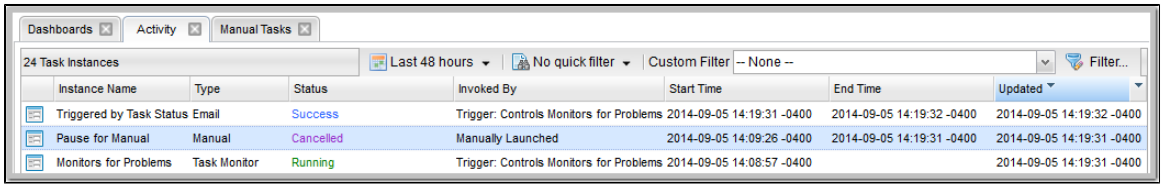
Step 2	Click the **New** button to display an empty Task Monitor Trigger Details and enter the following values: • **Name** = Controls Monitors for Problems • **Task(s)** = Triggered by Task Status • **Task Monitor** = Monitors for Problems
Step 3	Click the **Save** button. 
Step 4	On the Task Monitor Triggers list, right-click **Controls Monitors for Problems** and then click **Enable**.

Running the Task Monitor

To test our set-up, we need to run a task to one of the failure statuses that will trigger the email. To do so, we will launch the **Pause for Manual** Manual task created in the Running a Workflow with a Conditional Path tutorial and force it into Failed status.

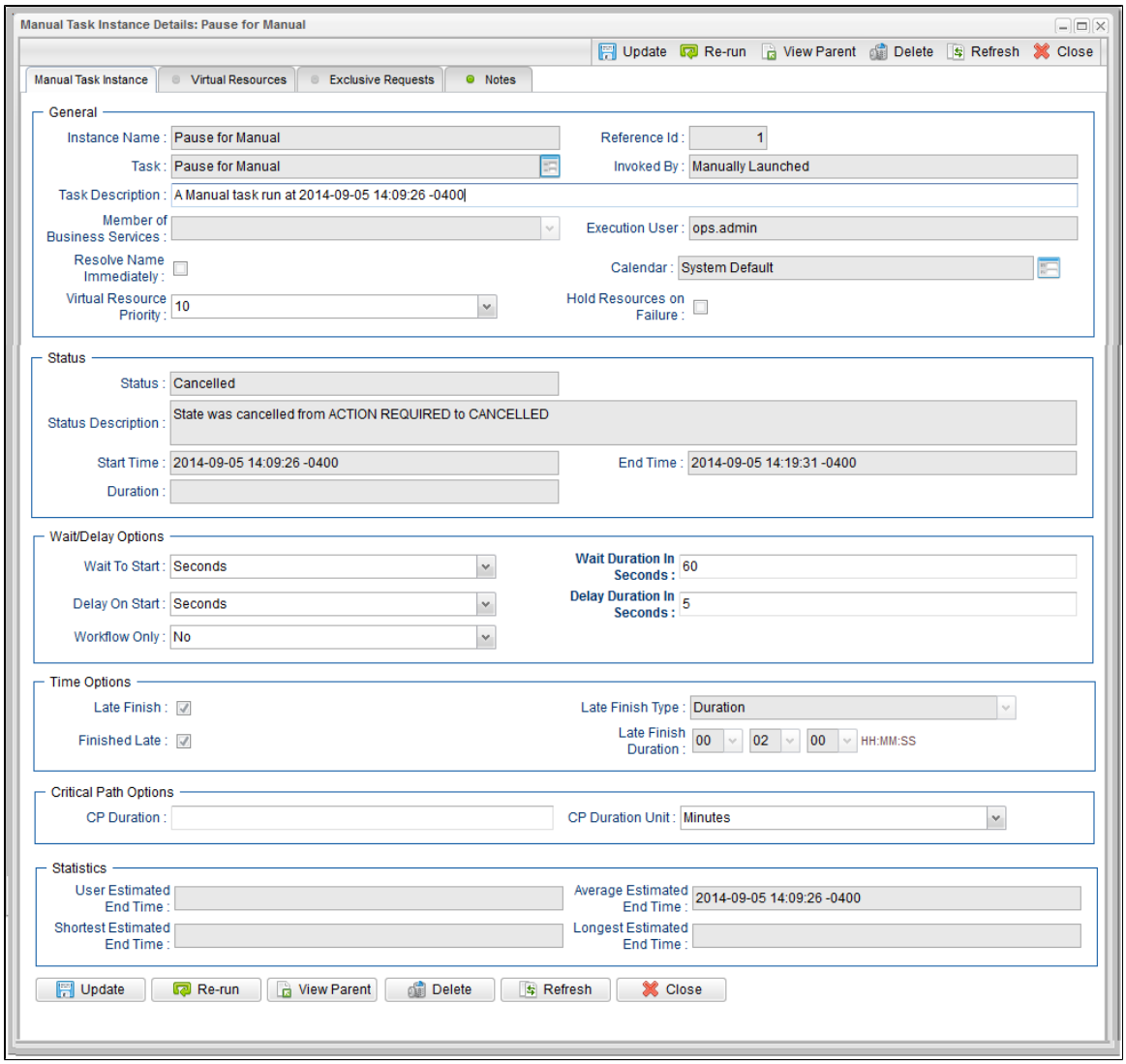
Step 1	From the Automation Center navigation pane, select **Tasks > Manual Tasks**. The Manual Tasks list displays.
Step 2	Right-click the **Pause for Manual** task and, on the Action menu, select **Launch Task**.
Step 3	Display the Activity Monitor. It will list the **Pause for Manual** task instance, in **Action Required** status, and the **Monitors for Problems** Task Monitor task, which was launched when enabled the **Controls Monitor for Problems** triggered 
Step 4	Right-click on **Pause for Manual** to display an Action menu of actions currently available for this task instance.

Step 5 Click **Cancel**. **Pause for Manual** goes to **Cancelled** status and an Email task is launched by the Task Monitor trigger.



Step 8 Check the Email account where you sent the notification.

Step 9 Once the user receives the email, the user can quickly check for more information by looking at the Activity Monitor using the Cancelled Task Instances filter and clicking on the Instance Name of the problem task. As shown in the illustration below, additional information about the issue is displayed in the Status Description field.



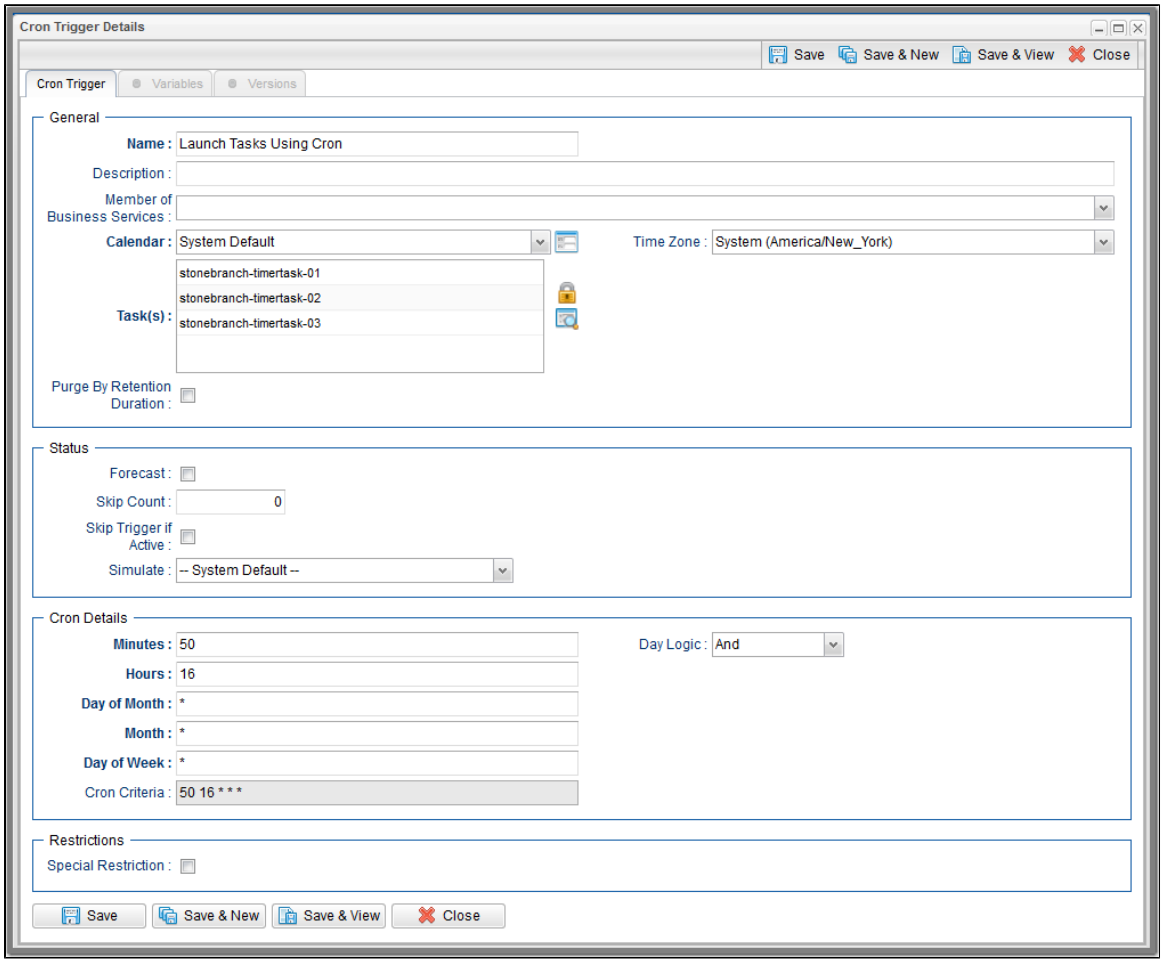
For additional information, see:

- Email Task
- Email Connections
- Email Templates
- Reports
- All Task Instances Table (ops_exec)

- Activity Monitor
- Task Monitor Task
- Task Monitor Trigger
- Command Quick Reference
- Cancelling a Task Run

Tutorial - Launching Tasks Using a Cron Trigger

Create a Cron Trigger

Step 1	From the Automation Center navigation pane, select **Triggers > Cron Triggers**. The Cron Triggers list displays.
Step 2	Click the **New** button to display Cron Trigger Details for a new trigger and enter/select the following values: • **Name** = Launch Tasks Using Cron • **Task(s)** = stonebranch-timertask-01, stonebranch-timertask-01=2, and stonebranch-timertask-03 Timer tasks • **Minutes** = Number of minutes past the hour you want the tasks to run. For example, if you want the tasks to run at for 3:16, enter 16. • **Hours** = Hour (in 24-hour time) that you want the tasks to run. For example, if you want the tasks to run at for 3:16, enter 15. Universal Controller uses the time zone of the Controller server.
Step 3	Keep the asterisks (*) in the remaining fields and click the **Save** button. 
Step 4	In the Cron Triggers list, right-click Launch Tasks Using Cron and click **Enable**.
Step 5	Open the Activity Monitor to see that three Timer tasks are run at the selected time.

For additional information, see:

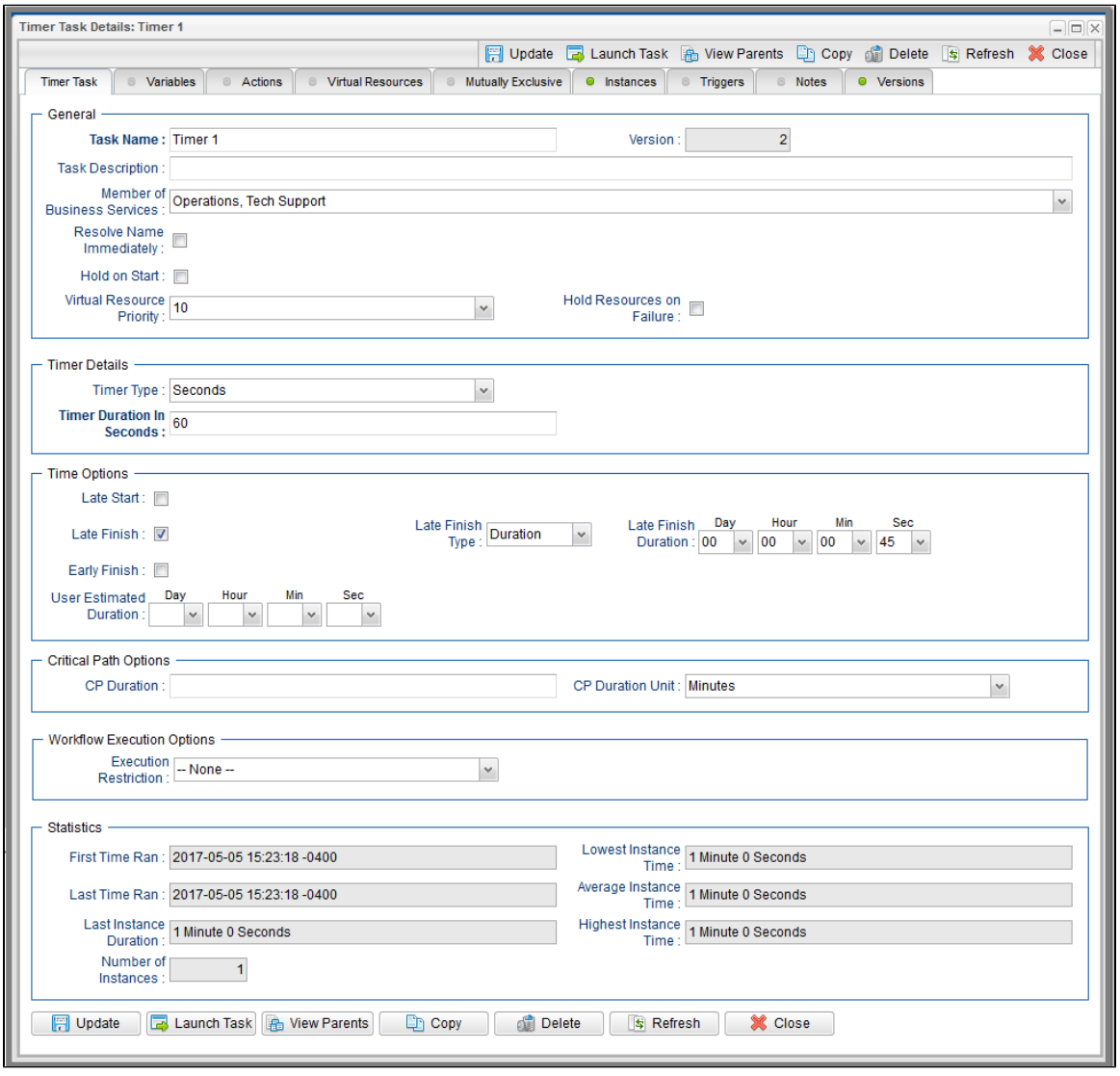
- Cron Trigger

Tutorial - Aborting a Process Launched by a Task

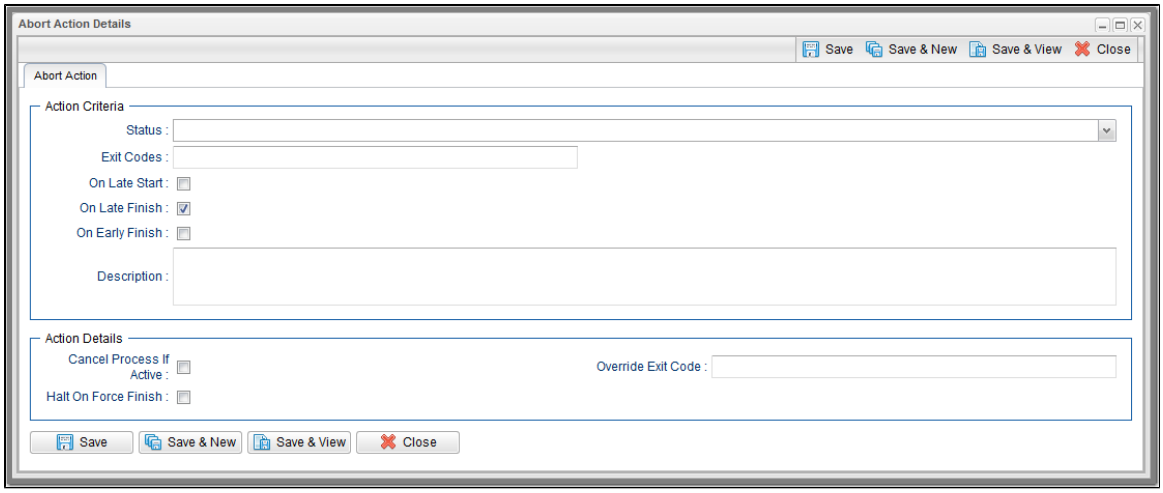
You can use an Abort Actions to instruct Universal Controller to abort a process under certain conditions. For example, you may want to abort a task if it is running too long.

In this tutorial, we will set a Timer task to run for 60 seconds and specify an Abort Action when the task runs 45 seconds.

Step 1	Open the Timer1 task created in the Creating a Simple Workflow tutorial and enter / select the following values: • **Time in Seconds** = 60 • **Late Finish** = enabled • **Late Finish Type** = Duration • **Late Finish Duration** = 00:00:45

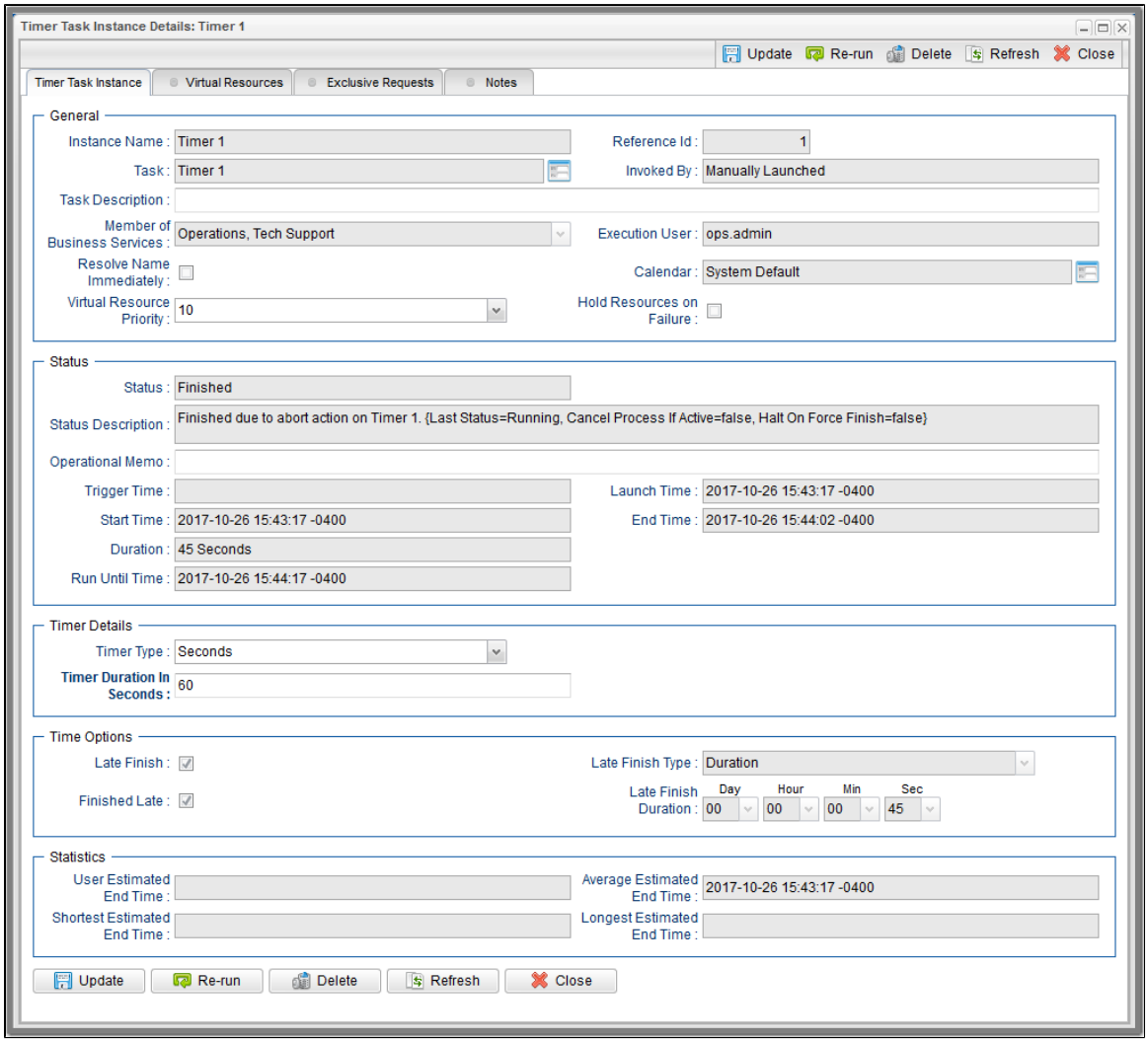


Step 2	Click the **Actions** tab to display the Actions list.
Step 3	Click **Abort Action** to display the Abort Action list.

Step 4	Click **New** to display Abort Action Details for a new Abort action, enable **On Late Finish**, and click **Save**. 
Step 5	Click the **Timer Task** tab and then click the **Update** button.
Step 6	Right-click **Timer1** in the Timers Tasks list and click **Launch Task**.
Step 7	Navigate to the Activity Monitor and verify that after running for 45 seconds, the task instance status changes from Running to Finished.

Step 8 Open the task instance Details and note the status description indicates:

```
Finished due to abort action on Timer 1
```



For additional information, see:

- Setting Up Abort Actions

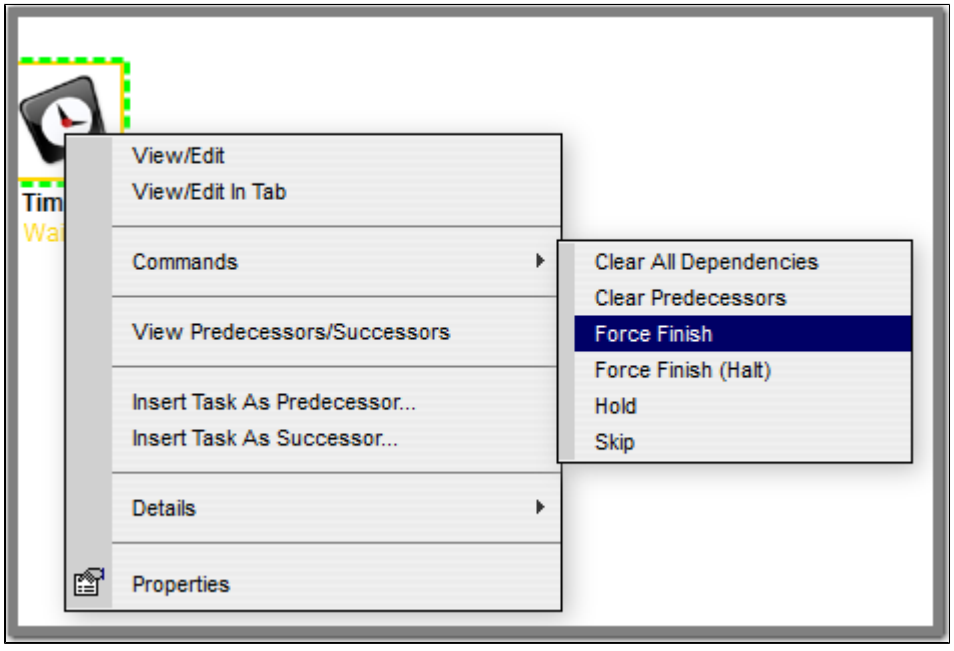
Tutorial - Force Finishing, Force Finish-Cancelling, and Cancelling a Task

In this exercise, we will force finish, force finish/cancel, and cancel tasks within a workflow from three areas:

- Workflow Monitor
- Activity Monitor
- Task Instances list

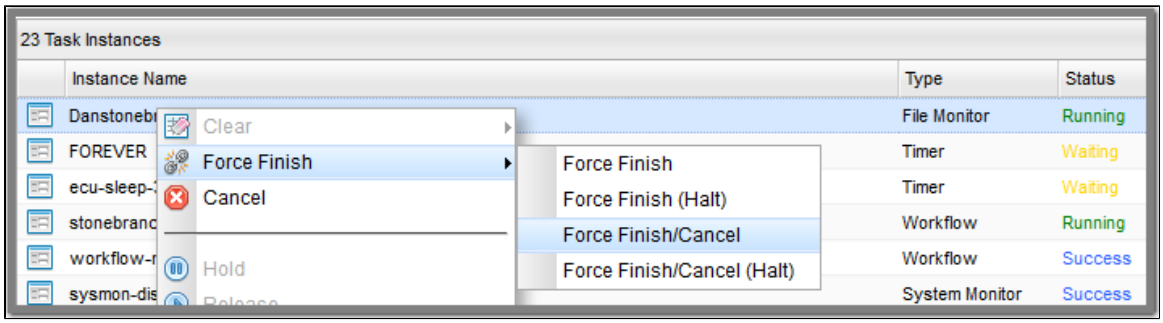
You can run any of these three commands from any of these three areas. For stand-alone tasks, you can run these commands only from the Activity Monitor and Task Instances list.

> **Note**
> You can force finish, force finish/cancel, and cancel any task in Running status, but you only can force finish a task in Waiting status. See Manually Running and Controlling Tasks for a complete list of task statuses for each command.

Step 1	From the Automation Center navigation pane, select **Tasks > Workflow Tasks**. The Workflow Tasks list displays.
Step 2	Create a Workflow with an Agent-based task, such as a File Monitor task, that can be Force Finished, Force Finish / Cancelled, and Cancelled. (See the Creating a Simple Workflow tutorial for help on creating the Workflow.)
Step 3	Click **Launch Task** to run the Workflow.
Step 4	**Force Finish a task from the Workflow Monitor:** 1. From the **Automation Center** navigation pane, select **Task Instances > Task Instances** to display the Activity Monitor which displays, by default, a list of Active Task Instances. 2. Open the running Workflow and click the View Workflow button to display its Workflow Monitor. 3. Right-click a Waiting task and, from the list of Commands on the pop-up menu, click Force Finish. The status of the task changes from Waiting to Finished, and all successor task instances waiting for successful completion of this task instance will start. 

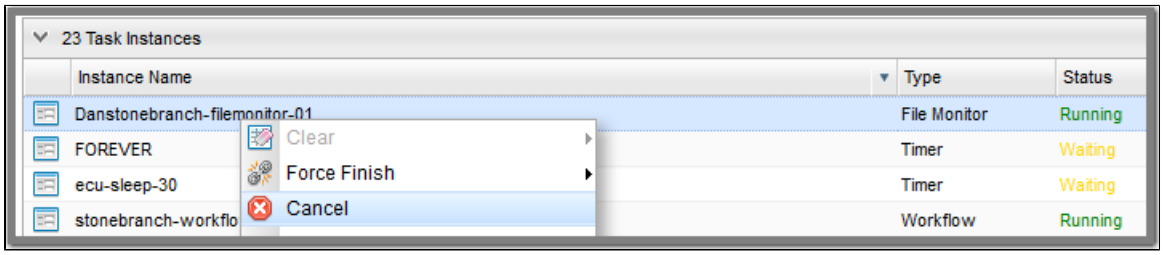
Step 5 **Force Finish/Cancel a task from the Activity Monitor:**

1. Return to the Activity Monitor.
2. Right-click a Simple Workflow task that is in the Running status and, on the Action menu, click Force Finish/Cancel. The status of the task changes from Running to Finished, and all successor task instances waiting for successful completion of this task instance will start.



Step 6 **Cancel a task from the Task Instances list:**

1. From the **Automation Center** navigation pane, select **Task Instances > Task Instances** to display the Task Instances list.
2. Right-click a Simple Workflow task in the Running status and, on the Action menu, click Cancel. The status of the task changes from Running to Cancelled, and all successor task instances waiting for successful completion of this task will remain in the Waiting status. The status of Simple Workflow changes from Running to Running/Problems.



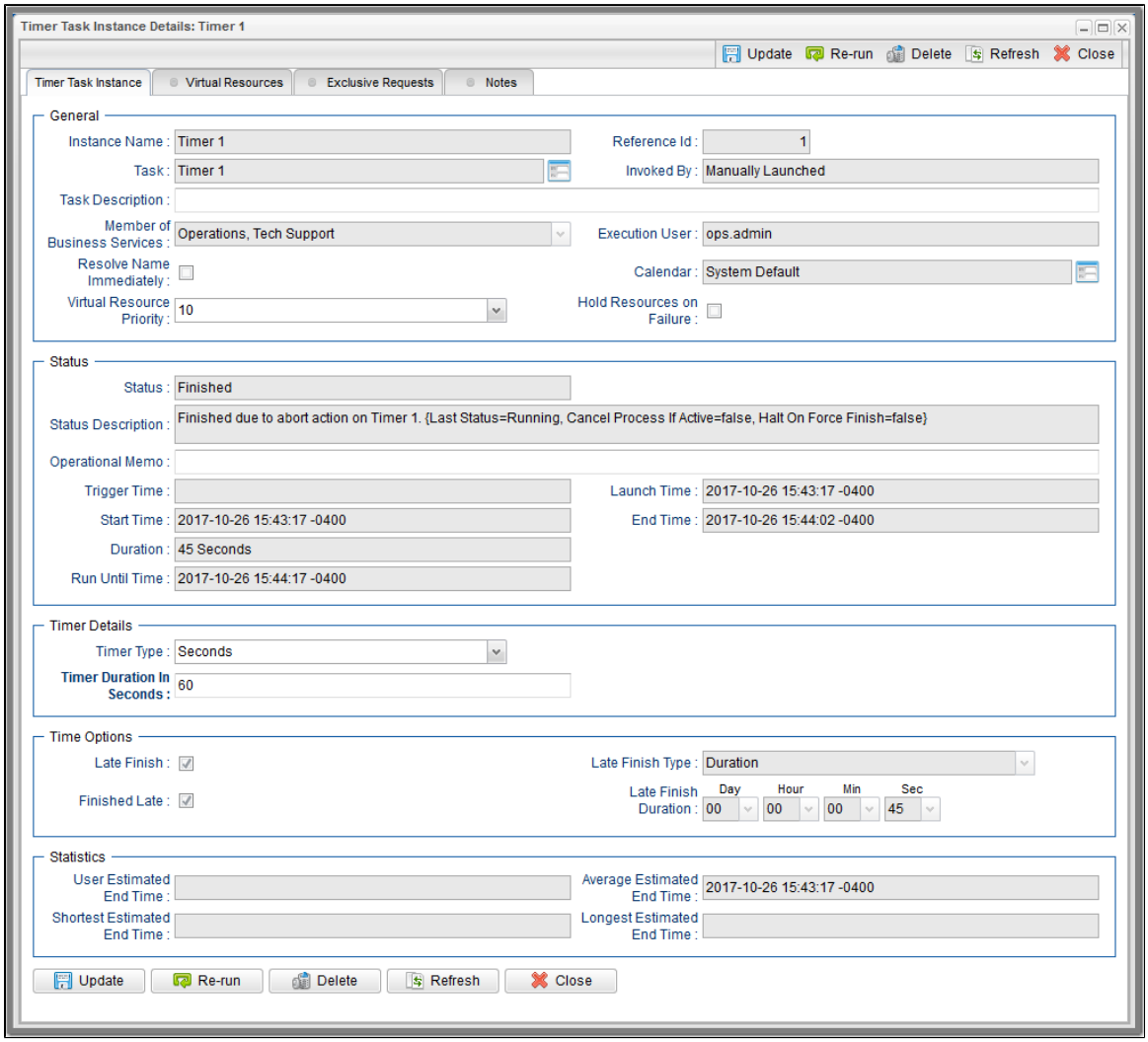
For additional information, see:

- Force Finishing a Task
- Cancelling a Task
- Force Finish / Cancelling a Task
- Monitoring Activity from the Activity Monitor
- Monitoring Activity from the Task Instances List
- Monitoring Workflows

Tutorial - Accessing Task Instance Details

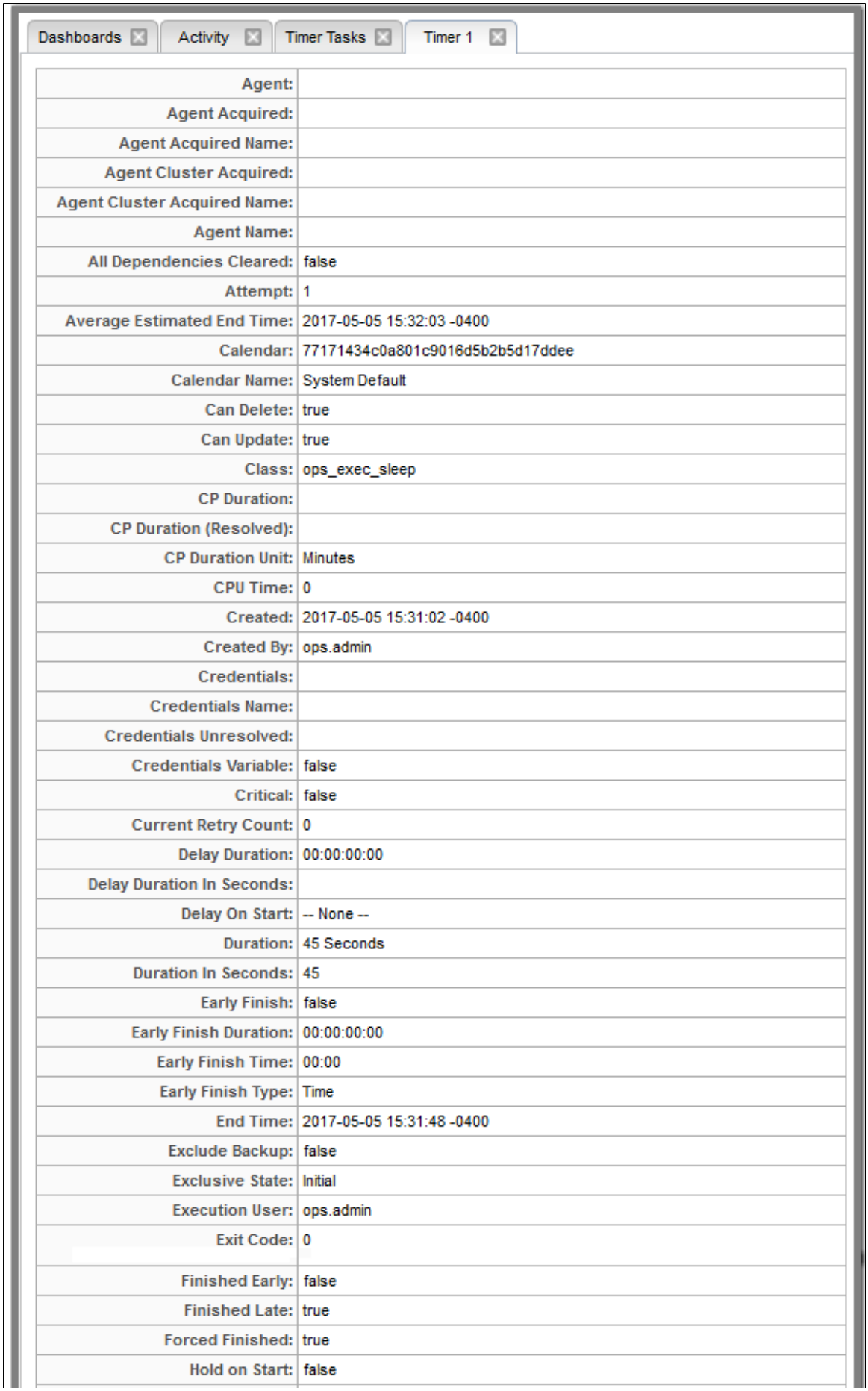
Step 1 From the Automation Center navigation pane, select **Task Instances > Activity** to display the Activity Monitor.

Step 2 Click the **Instance Name** of any task instance to display its Task Instance Details. For example:



Task Instance Details contain many fields not displayed in the Task Details for this task that provide information about this run (instance) of the task and all runs of the task, including the first time it was run; the last time it was run; the number of times it has run; and the least, average, and most amount of time it has ever taken to run.

Step 3 To view all details stored in the All Task Instances table (`ops_exec`) for this task instance, right-click anywhere in the Task Instance Details to display an Action menu and then click **Details > Show Details**.



Dashboards | Activity | Timer Tasks | Timer 1

Agent:	
Agent Acquired:	
Agent Acquired Name:	
Agent Cluster Acquired:	
Agent Cluster Acquired Name:	
Agent Name:	
All Dependencies Cleared:	false
Attempt:	1
Average Estimated End Time:	2017-05-05 15:32:03 -0400
Calendar:	77171434c0a801c9016d5b2b5d17ddee
Calendar Name:	System Default
Can Delete:	true
Can Update:	true
Class:	ops_exec_sleep
CP Duration:	
CP Duration (Resolved):	
CP Duration Unit:	Minutes
CPU Time:	0
Created:	2017-05-05 15:31:02 -0400
Created By:	ops.admin
Credentials:	
Credentials Name:	
Credentials Unresolved:	
Credentials Variable:	false
Critical:	false
Current Retry Count:	0
Delay Duration:	00:00:00:00
Delay Duration In Seconds:	
Delay On Start:	-- None --
Duration:	45 Seconds
Duration In Seconds:	45
Early Finish:	false
Early Finish Duration:	00:00:00:00
Early Finish Time:	00:00
Early Finish Type:	Time
End Time:	2017-05-05 15:31:48 -0400
Exclude Backup:	false
Exclusive State:	Initial
Execution User:	ops.admin
Exit Code:	0
Finished Early:	false
Finished Late:	true
Forced Finished:	true
Hold on Start:	false

Hold Reason:	
Hold Resources on Failure:	false
Instance Name:	Timer 1
Invoked By:	Manually Launched
IO Other:	0
IO Reads:	0
IO Writes:	0
Is Version:	false
Late Finish:	true
Late Finish Duration:	00:00:00:45
Late Finish Time:	00:00
Late Finish Type:	Duration
Late Start:	false
Late Start Duration:	00:00:00:00
Late Start Time:	00:00
Late Start Type:	Time
Launch Time:	2017-05-05 15:31:02 -0400
Longest Estimated End Time:	2017-05-05 15:32:03 -0400
Maximum Retries:	0
Member of Business Services:	Operations, Tech Support
Member of Business Services:	3fa01c7d335a44f0a93c85955c833aac,209686ac5a4f4eaebe488e9a3749ceb7
Memory Peak:	0
Memory Used:	0
Next Retry Time:	
Operational Memo:	
Predecessors Satisfied Time:	
Progress:	
Projected End Time:	
Queued Time:	
Reference Id:	2
Resources Consumed:	false
Resources State:	Initial
Retention Time:	
Retry Indefinitely:	false
Retry Interval (Seconds):	60
Run Called:	true
Run Criteria Run Time:	false
Run Criteria Trigger Time:	false
Run Until Time:	2017-05-05 15:32:03 -0400
Security Name:	Timer 1
Shortest Estimated End Time:	2017-05-05 15:32:03 -0400
Start Time:	2017-05-05 15:31:03 -0400
Started Late:	false
State Changed Time:	2017-05-05 15:31:48 -0400
Status:	Finished
Status Description:	Finished due to abort action on Timer 1. {Last Status=Running, Cancel Process If Active=false}

Status History:	2017-05-05 15:31:02 -0400: Defined 2017-05-05 15:31:03 -0400: Running 2017-05-05 15:31:48 -0400: Finished
Suppress Intermediate Failures:	false
Tab Names Containing Data:	
Table Name:	ops_exec_sleep
Task:	6fb0f68152a748f6ab7346647c5f53e5
Task Description:	
Task Name:	Timer 1
Task Priority:	MEDIUM
Time Wait State:	Initial
Timer Day Constraint:	-- None --
Timer Duration:	00:00:00:00
Timer Duration In Seconds:	60
Timer Time (HH:MM):	00:00
Timer Type:	Seconds
Trigger:	
Trigger Name:	
Trigger Time:	
Type:	Timer
Universal Template:	
Universal Template Name:	
Updated:	2017-05-05 15:31:48 -0400
Updated By:	ops.system
User Defined Field 1:	
User Defined Field 2:	
User Estimated End Time:	
UUID:	1493749096262784946 8SYUYMII970CZ
Vertex Id:	
Virtual Resource Priority:	10
Wait Day Constraint:	-- None --
Wait Duration:	00:00:00:00
Wait Duration In Seconds:	
Wait Time (HH:MM):	00:00
Wait To Start:	-- None --
Wait Until Time:	
Waited for Exclusive:	false
Waited for Resources:	false
Workflow:	
Workflow Definition:	
Workflow Definition Name:	
Workflow Name:	
Workflow Only:	Yes
Workflow Start Time:	

Print

- Left column shows each field in the All Task Instances table for this task instance.
- Right column shows the current value for each field for this task instance.

Tutorial - Monitoring Task Activity

- Starting and Stopping the Activity Monitor
- Apply Time Constraints and Filters to the Activity Monitor
- Apply Display Settings to the Activity Monitor

In this tutorial, we will monitor task activity from the Activity Monitor.

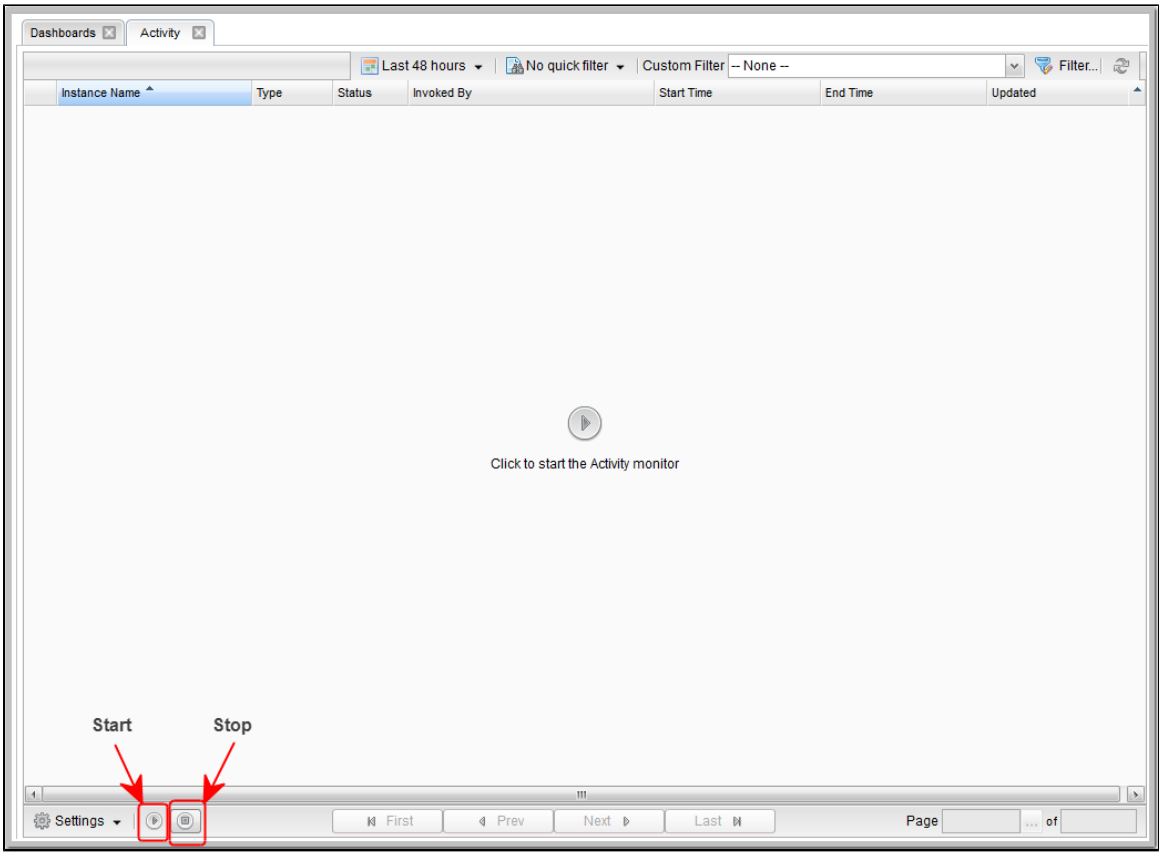
We will start and stop the Activity Monitor, apply time constraints and filters to the task instances displayed on the Activity Monitor, and apply display settings to the Activity Monitor.

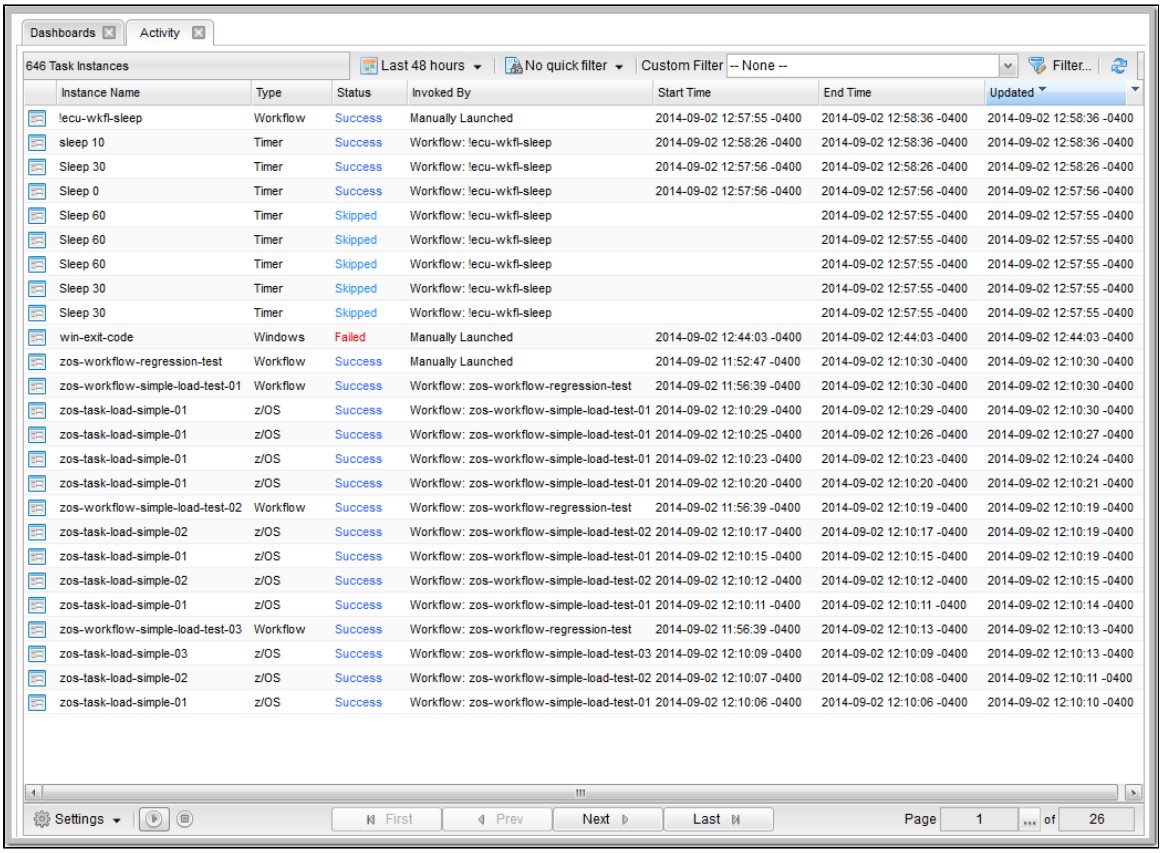
Starting and Stopping the Activity Monitor

By default, the Activity Monitor does not automatically monitor Universal Controller activity when you log in. You must start the Activity Monitor to see task activity.

(You can allow the Activity Monitor to automatically monitor Controller activity when you log in by changing the Activity Monitor Automatically user preference.)

Step 1 From the Automation Center navigation pane, select **Task Instances > Activity**. The Activity Monitor displays.



Step 2	To start the Activity Monitor, either: 1. Click the **Click to start the Activity monitor** button in the middle of the Activity Monitor. 2. Click the **Start Activity monitor** button at the bottom of the Activity Monitor. The Activity Monitor then displays a list of all current activity. 
Step 3	To stop the Activity Monitor, click the **Stop Activity monitor** button at the bottom of the Activity Monitor.

Apply Time Constraints and Filters to the Activity Monitor

(You can select apply any time constraint for any combination of multiple filters for the list of task instances on the the Activity Monitor.)

Step 1	Select a time frame of task activity to display on the Activity Monitor by clicking the Time Constraint button and selecting a time fame from the menu. The default is **Last 48 hours**, which means that the Activity Monitor will display only task activity that occurred in the last 48 hours.
Step 2	Click the Quick Filter button to display a menu of simple, pre-defined filters that you can apply to the list, such as a specific task type of specific task instance status. You can apply as many Quick Filters as you like to the list.
Step 3	Click the **Clear** button at the bottom of the Quick Filter menu to remove all Quick Filters from the list.
Step 4	Click the **Filter** button to select a Custom Filter of complex, user-defined filter that you can apply to the list, such as only task instances that belong to a specific Business Service.

Apply Display Settings to the Activity Monitor

Step 1	Click the Settings button at the bottom of the Activity Monitor to select: • Number of task instances to display on each page of the Activity Monitor. • Refresh rate for the dynamic data displayed on the Activity Monitor.

Step 2	Click the **First**, **Prev**, **Next**, and **Last** buttons at the bottom of the Activity Monitor to navigate through multiple pages of activity.
Step 3	Click the ellipse (**...**) button at the bottom of the Activity Monitor to select a specific page of activity to display.

Tutorial - Creating a Simple Workflow

- Introduction
- Create and Copy Tasks
- Creating a Simple Workflow
- Running the Workflow

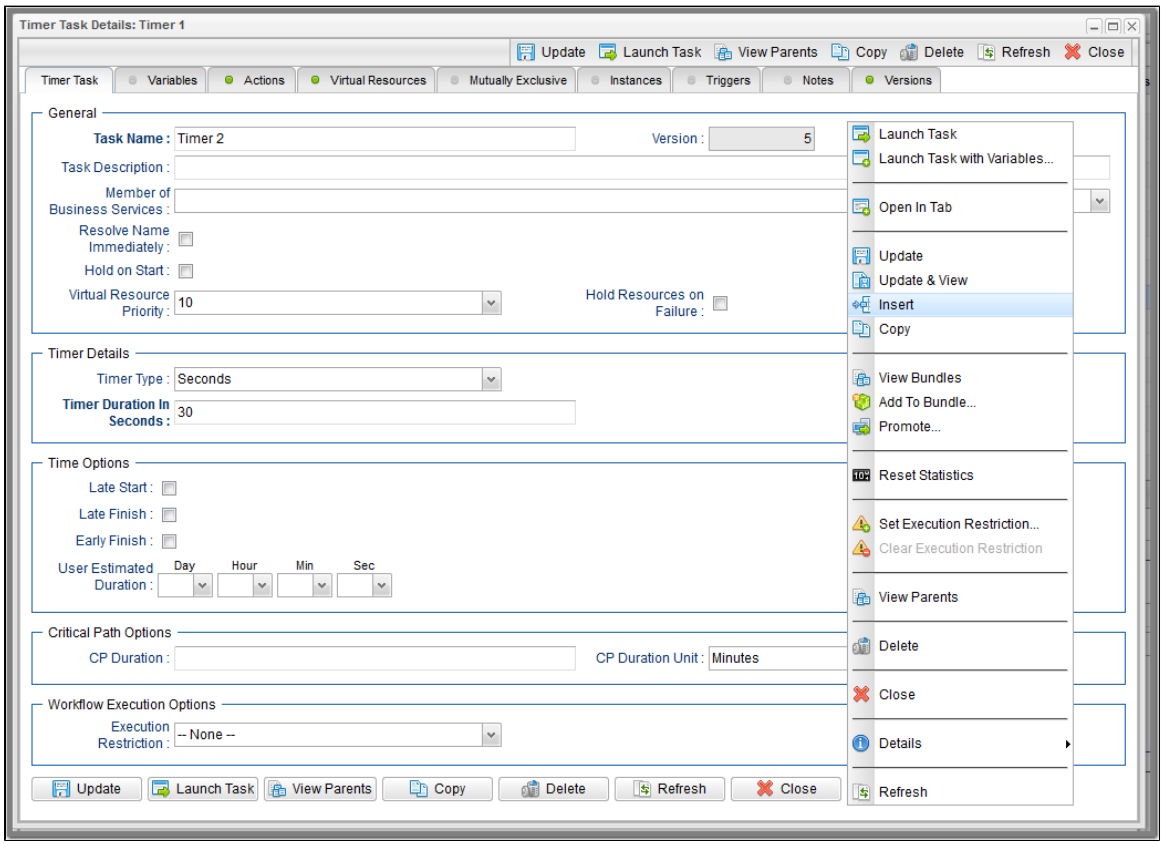
Introduction

In this tutorial, we will learn how to copy tasks, create a simple Workflow of Timer tasks, and use the tools available in the Workflow Editor.

Create and Copy Tasks

Create a Timer task and make five copies for use in the Workflow. Use the names Timer1 through Timer6, and assign each task a time of 10 seconds.

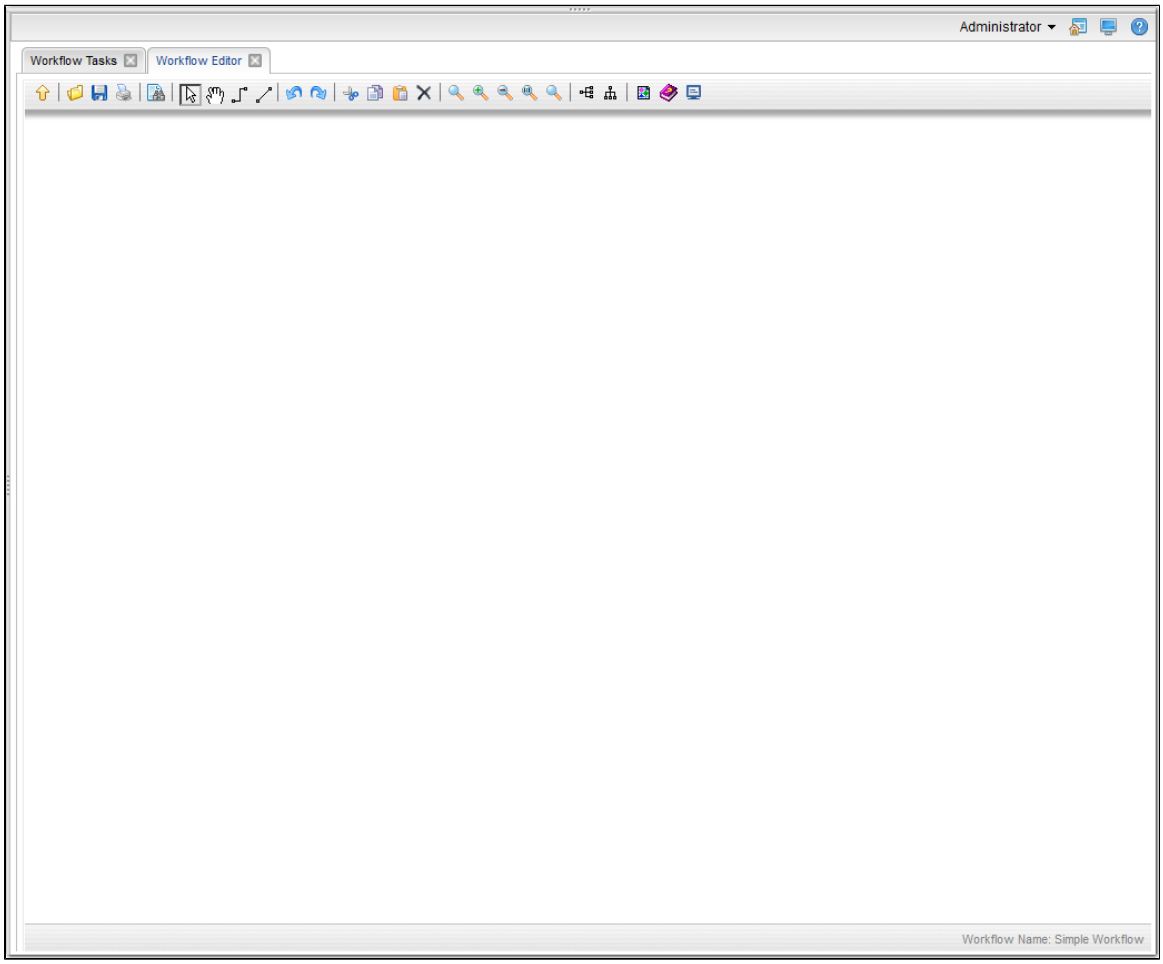
> **Note**
> You can copy tasks using different methods. One method is provided below.

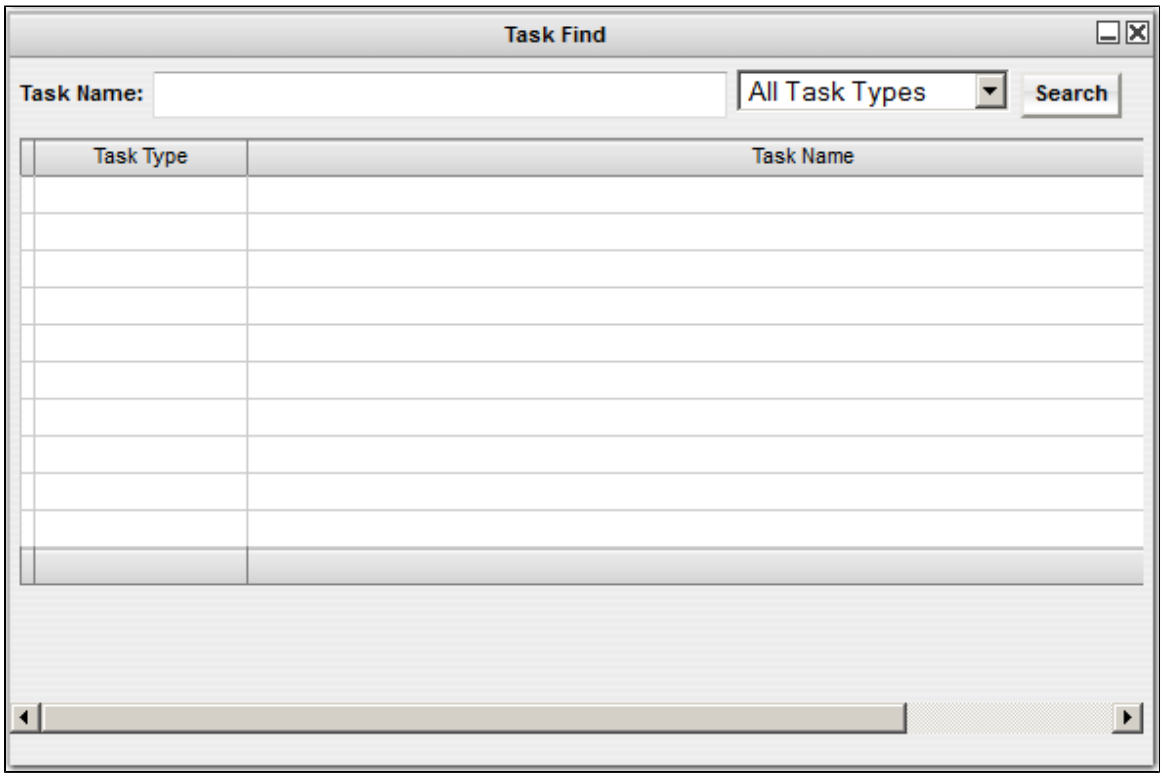
Step 1	From the Automation Center navigation pane, select **Tasks > Timer Tasks**. The Timer Tasks list displays.
Step 2	In the Timer Task Details below the list, create Timer1 and click **Save**.
Step 3	Change the **Task Name** from Timer1 to Timer2, right-click the Details to display an Action menu, and click **Insert** to make a copy of the Timer1 task named Timer2. 
Step 4	Repeat Step 3 for Timer3 through Timer6.

Creating a Simple Workflow

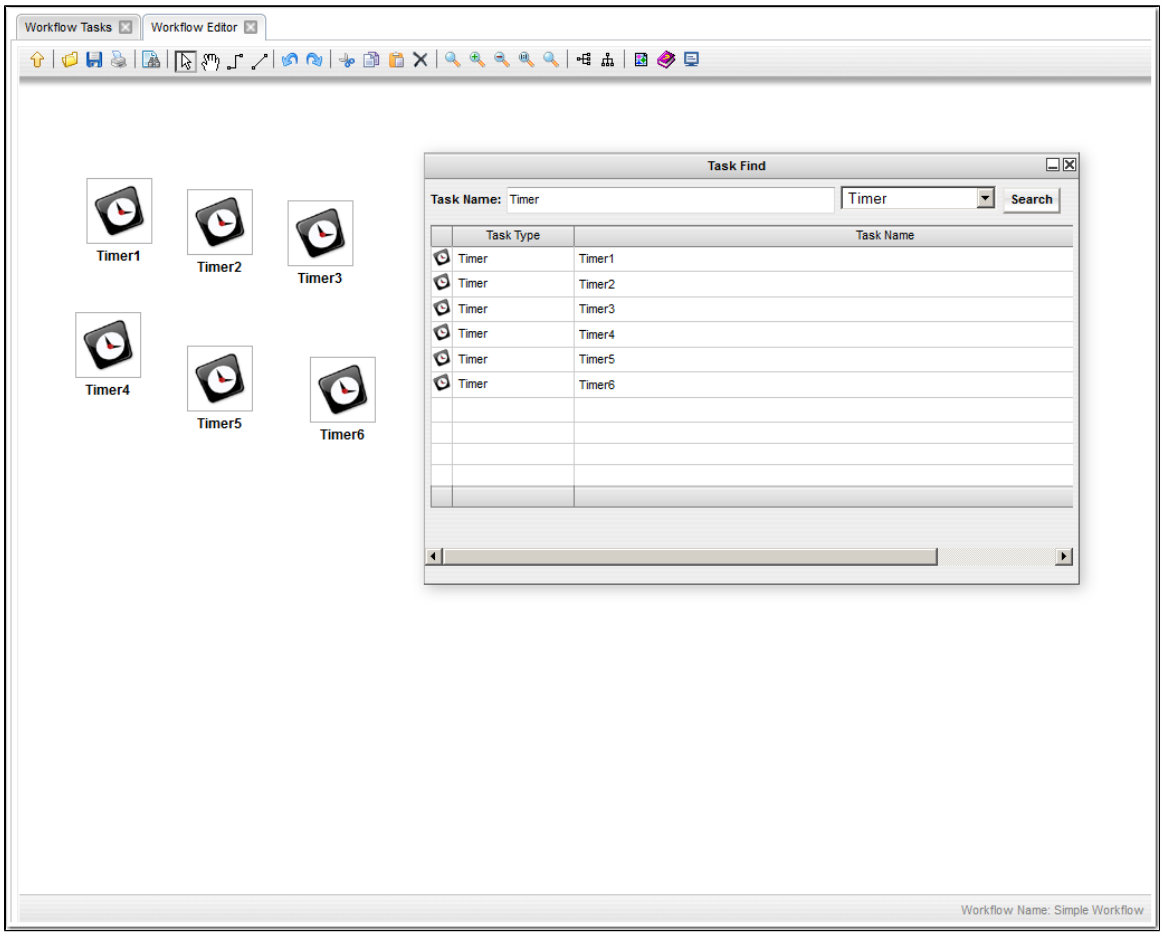
Now that we have six Timer tasks, we are ready to create a simple workflow.

(See Creating and Maintaining Workflows for detailed information on the tools and icons used in this procedure.)

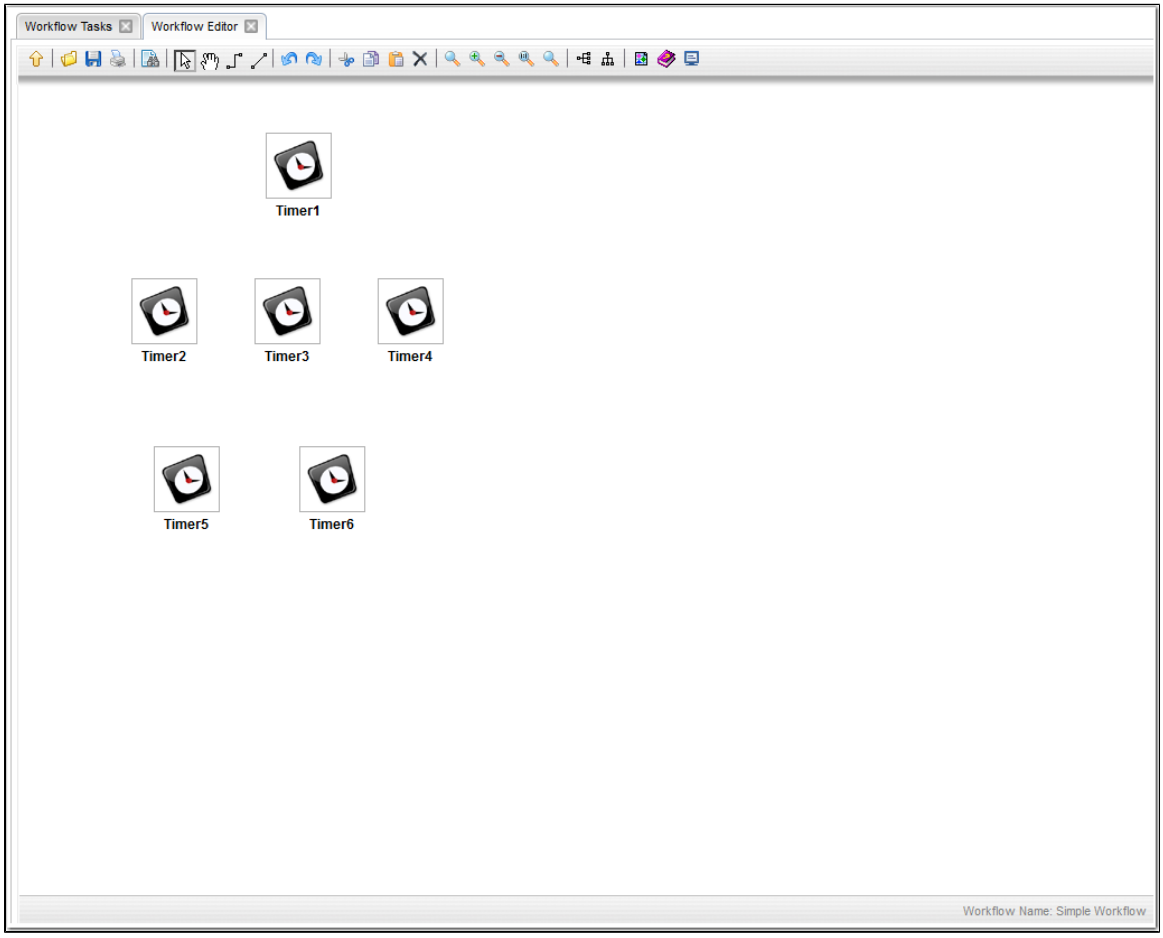
Step 1	From the **Automation Center** navigation pane, select **Tasks > Workflow Tasks**. The Workflow Tasks List displays.
Step 2	In the Workflow Task Details below the list, enter **Simple Workflow** in the **Task Name** field and then click the **Save** button.
Step 3	Right-click **Simple Workflow** in the Workflow Tasks list and click **Edit Workflow** in the Actions menu to display the Workflow Editor.  By default, the Workflow Editor displays in **Select** mode, which lets you select tasks for the Workflow. (See Workflow Editor Icons for a description of each icon on the toolbar.)

Step 4	Click the **Add Task** icon. The **Task Find** pop-up dialog displays. 
Step 5	Select **Timer** in the drop-down list and click the **Search** button. A list of all Timer tasks displays.

Step 6	Drag and drop the icon for Timer1 onto the Workflow Editor canvas, and repeat for Timer2 through Timer6.
Step 7	Close the **Task Find** pop-up dialog.

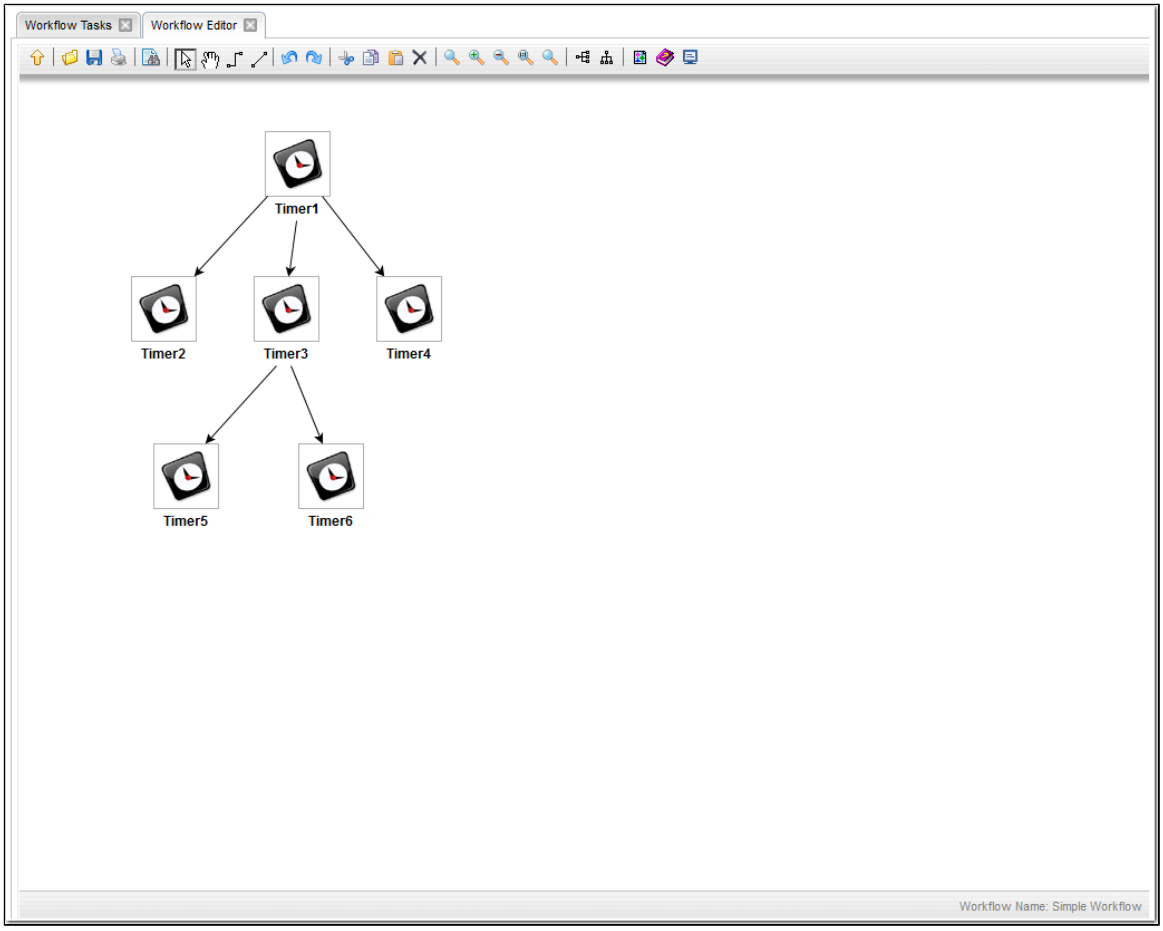


Step 8 Click and drag the tasks so that Timer1 is at the top of the canvas; Timer2, Timer3, and Timer4 are below Timer1; and Timer 5 and Timer6 are below Timer2, Timer3, and Timer4.

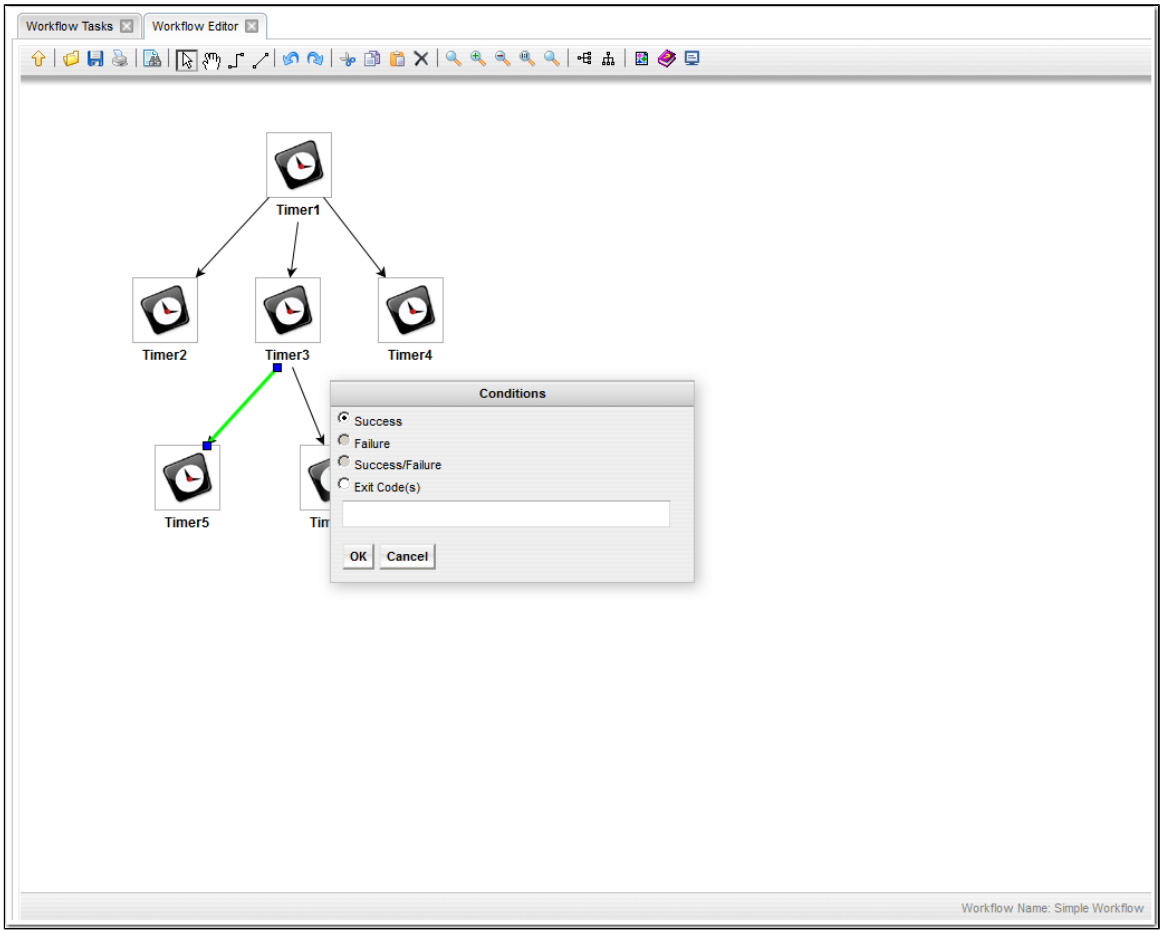


Step 9

Click a Connect Icon to connect the tasks so that Timer1 is the top-level predecessor task; Timer2, Timer3, and Timer4 are successor tasks to Timer1; and Timer5 and Timer6 are successor tasks to Timer3.



Step 10	The default condition (or dependency) for connectors is Success. That is, a successor task runs if its predecessor task goes to Success. To view the conditions for a successor task, right-click the connector between it and its predecessor task, and then click **Conditions**. The Conditions pop-up displays.



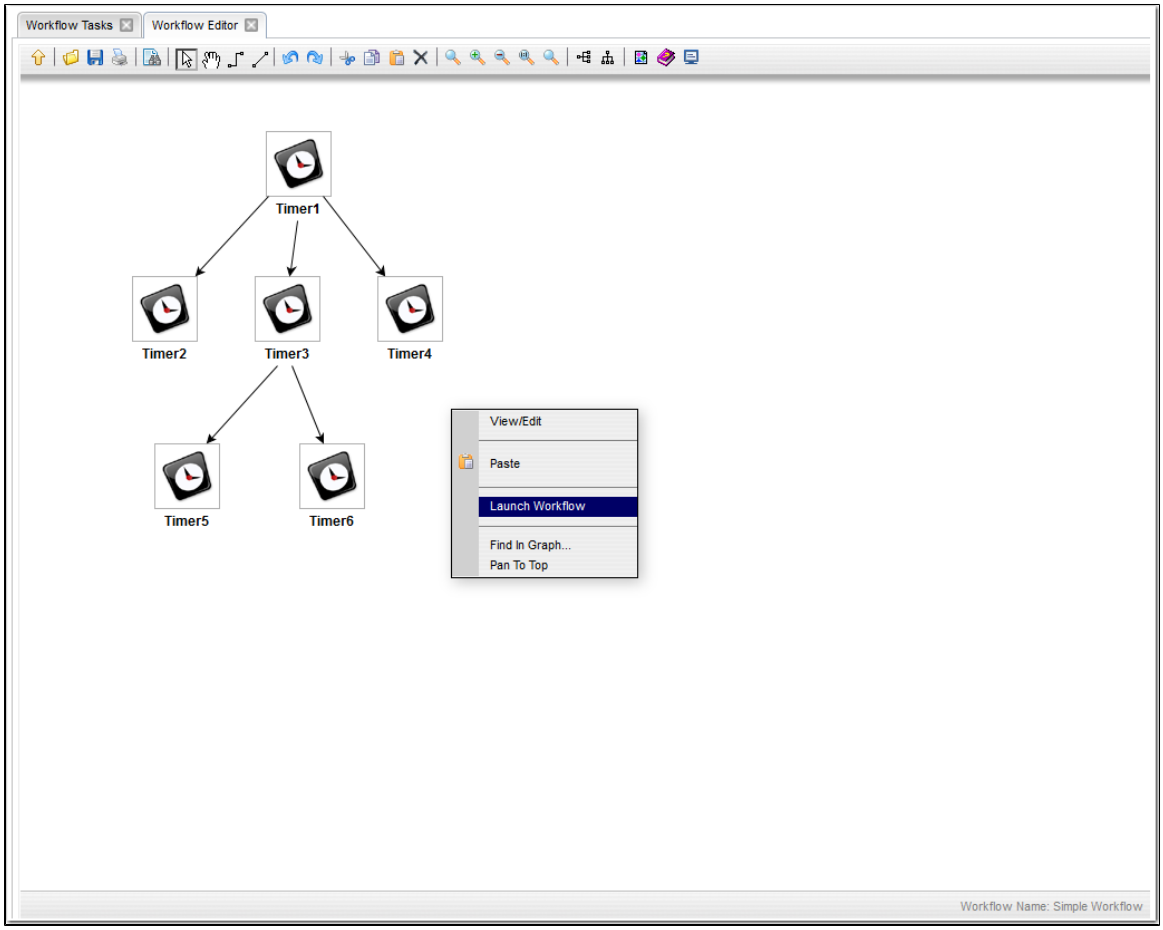
> **Note**
> Since a Timer task cannot go to Failure, the Failure and Success/Failure conditions are grayed out.

Step 11	Click the **Cancel** button on the Conditions pop-up, and then click the Save icon in the Workflow Editor task bar to save the Workflow.

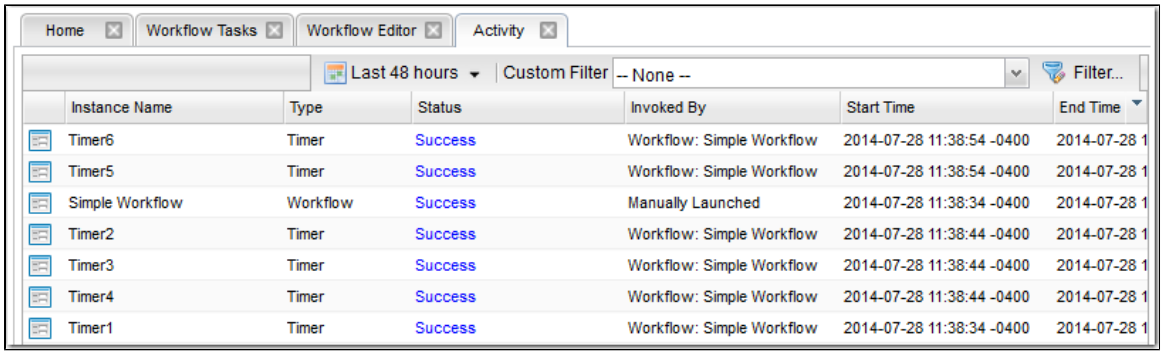
Running the Workflow

Now we will manually launch the Workflow and view it from the Activity Monitor.

Step 1 Right-click anywhere in the Workflow Editor canvas and, on the pop-up menu that displays, click **Launch Workflow**.



Step 2 From the **Automation Center** navigation pane, select **Task Instances > Activity**. You will see six task instances: the Simple Workflow task, and the six Timer tasks.



For additional information, see:

- Saving, Updating, Deleting, and Copying Records
- Creating Workflows

Tutorial - Running a Workflow with a Conditional Path

- Introduction
  - Prerequisite
- Create a Timer Task
- Create SQL Tasks
- Create a Manual Task
- Create a Workflow
- Run the Workflow to Success
- Run the Workflow Down the Conditional Path

Introduction

In this exercise, we will create a short workflow of SQL tasks. We will begin with a two-minute Timer task so that we will have enough time to see what the Workflow looks like on the Activity Monitor when we launch it. We will also create a conditional path, as follows:

- The Workflow runs seven days a week and creates a new database table. If that is successful, additional SQL tasks run that insert a value, select a count, and delete a value. Each subsequent task runs if the previous is successful.
- If the first (table creation) task fails, the Workflow goes to a Manual task instead of the regular flow. This is the conditional path. The Manual task creates a pause in the Workflow and sends an Email Notification. A user is expected to check the database and fix the problem that caused the first task to fail. If the Manual task is set to a Complete status, it goes to Success and the Workflow then returns to the remaining SQL tasks. While the Manual task remains in the Action Required status, the successor tasks have a Waiting status.

We will also add an Email Notification and a Note to this Workflow.

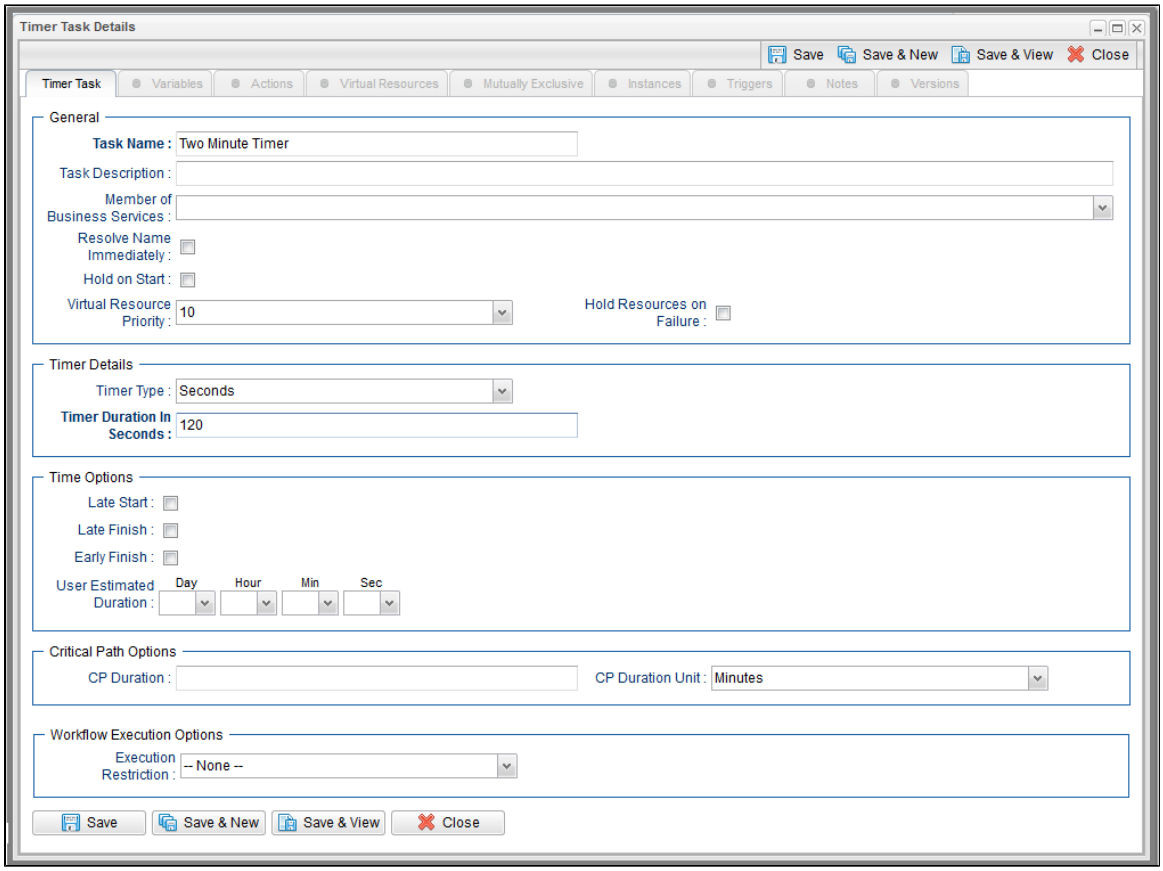
Prerequisite

Since we are using SQL tasks in this exercise, you will first need to create a SQL Database Connection.

Create a Timer Task

We will add a Timer task at the beginning of our Workflow so that we will have a chance to view it when Universal Controller loads it into the Activity Monitor.

Step 1	From the Automation Center navigation pane, select **Tasks > Timer Tasks**. The Timer Tasks list displays.

Step 2	Click **New** to display an empty Timer Task Details and enter / select the following values: • **Task Name** = Two Minute Timer • **Timer Type** = Secocnds • **Time Duration in Seconds** = 120 
Step 3	Click the **Save** button.

Create SQL Tasks

In this exercise, we will create SQL tasks that execute the following SQL commands:

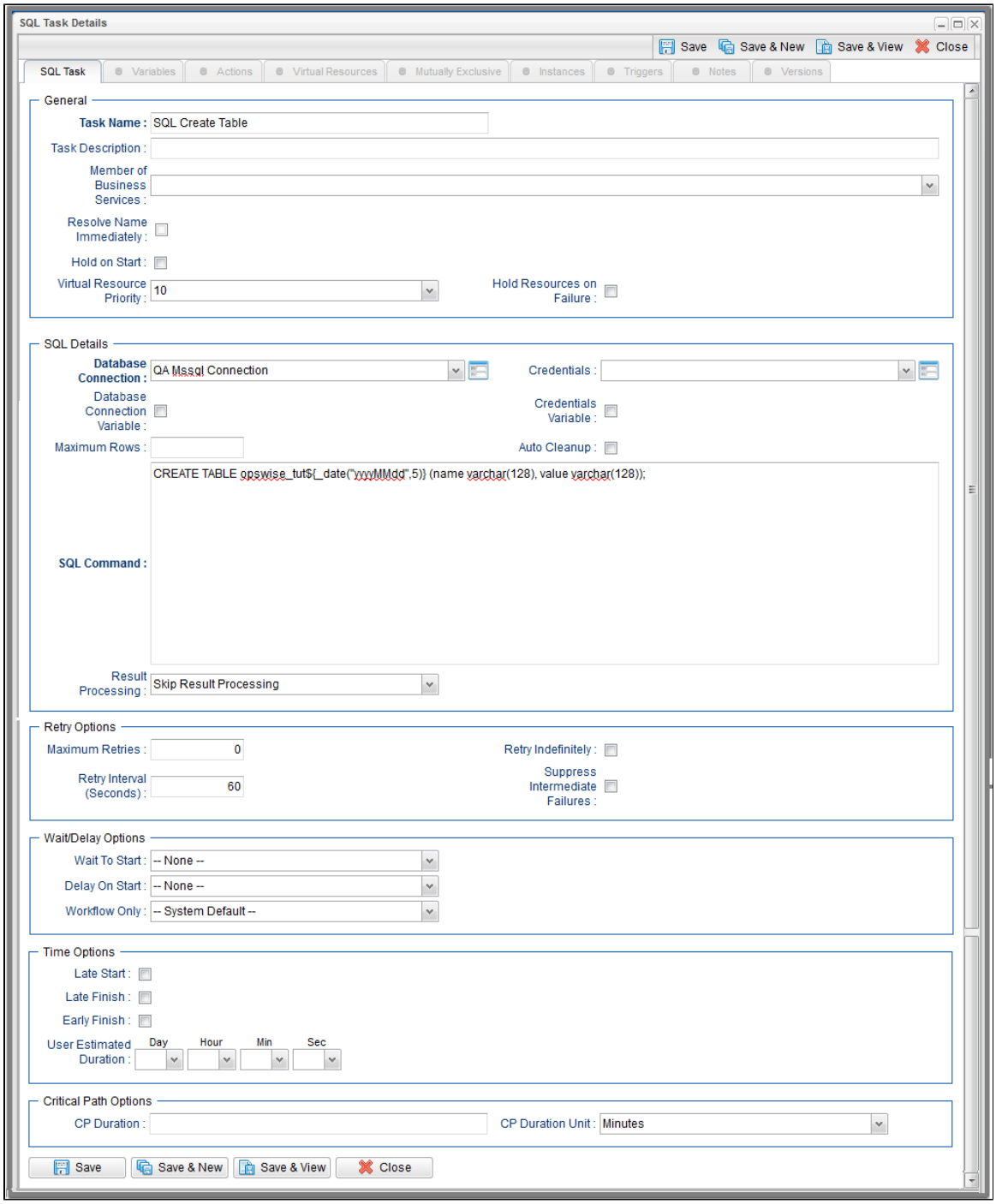
- Create a new table in the database.
- Insert a value into the table.
- Select a count value from the table.
- Delete the value from the table.

Perform the following steps to create the SQL tasks:

Step 1	From the **Automation Center** navigation pane, select **Tasks > SQL Tasks**. The SQL Tasks list displays.

Step 2 Click **New** to display an empty SQL Task Details and enter / select the following values:

- **Task Name** = SQL Create Table
- **Database Connection** = (the database connection you created as a [prerequisite|#Prerequisite])
- **SQL Command** = CREATE TABLE opswise_tut${_date("yyyyMMdd",5)} (name varchar(128), value varchar(128));



Step 3 Click the **Save** button.

Step 4 Create a SQL task called **SQL Insert Value** with this value:

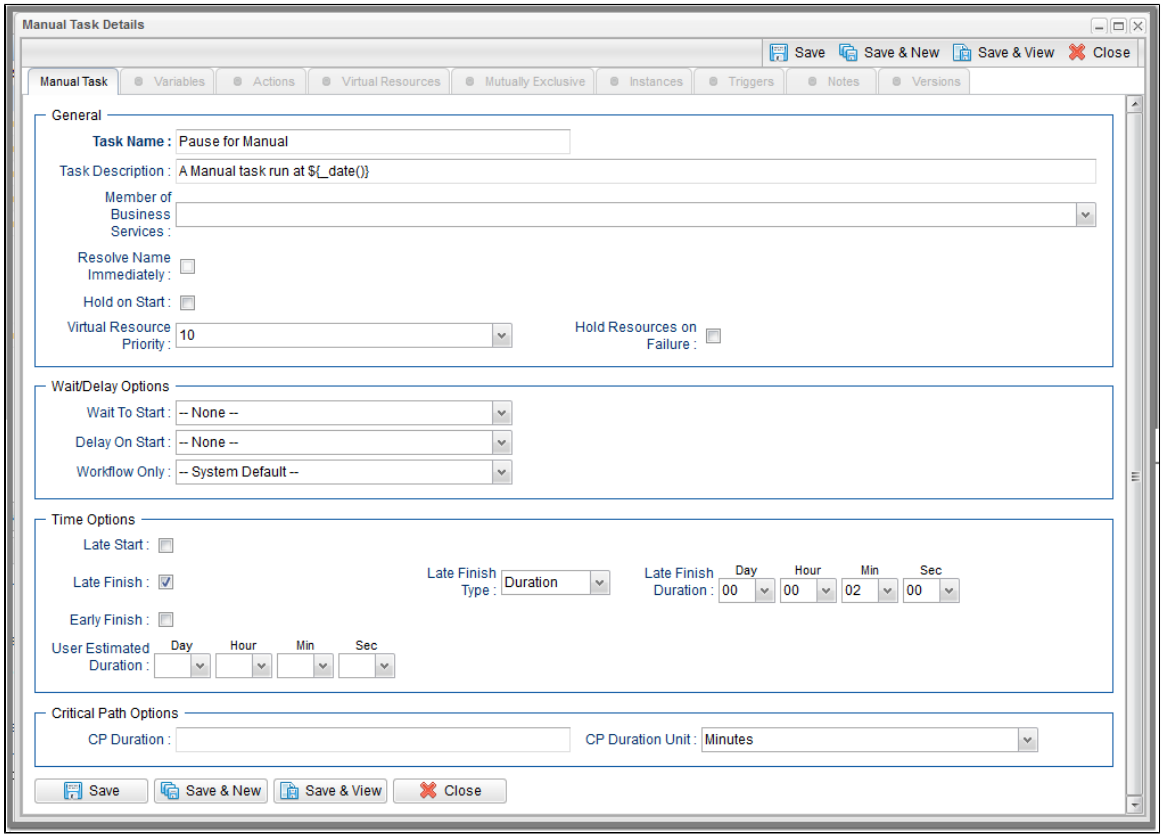
- **SQL Command** = INSERT INTO opswise_tut${_date("yyyyMMdd",5)} (name, value) values ('A', 'F'), ('B', 'S'), ('C', 'F');

Step 5	Create a SQL task called **SQL Select Count** with this value: • **SQL Command** = SELECT count (*) as count FROM opswise_tut${_date("yyyyMMdd",5)} WHERE value = 'F';
Step 6	Create a SQL task called **SQL Delete** with this value: • **SQL Command** = DELETE FROM opswise_tut${_date("yyyyMMdd",5)};

Create a Manual Task

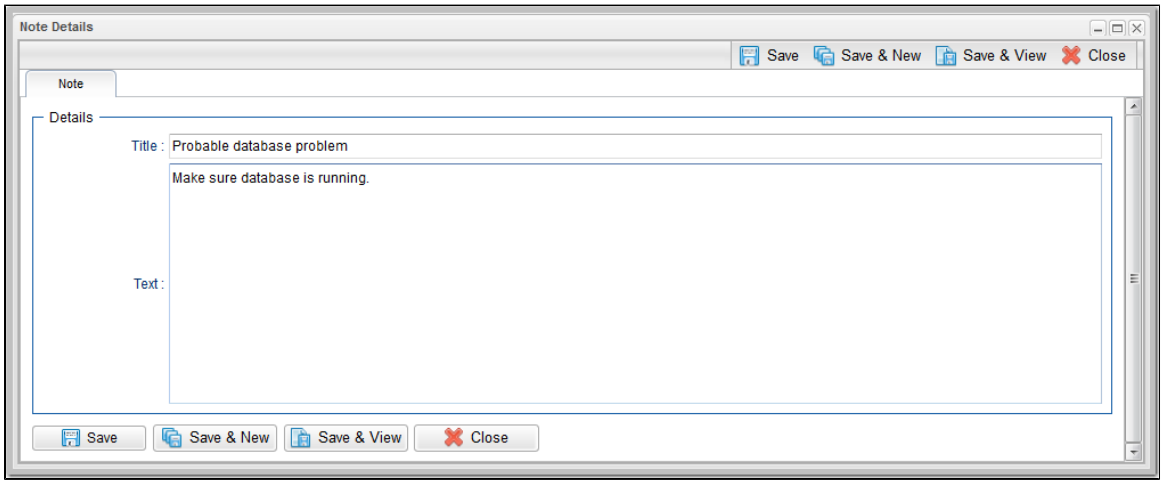
A Manual task is used within a Workflow to create a pause in processing, during which the user must perform some task. When the user task is complete, the user sets the Manual task to a completed state and processing continues.

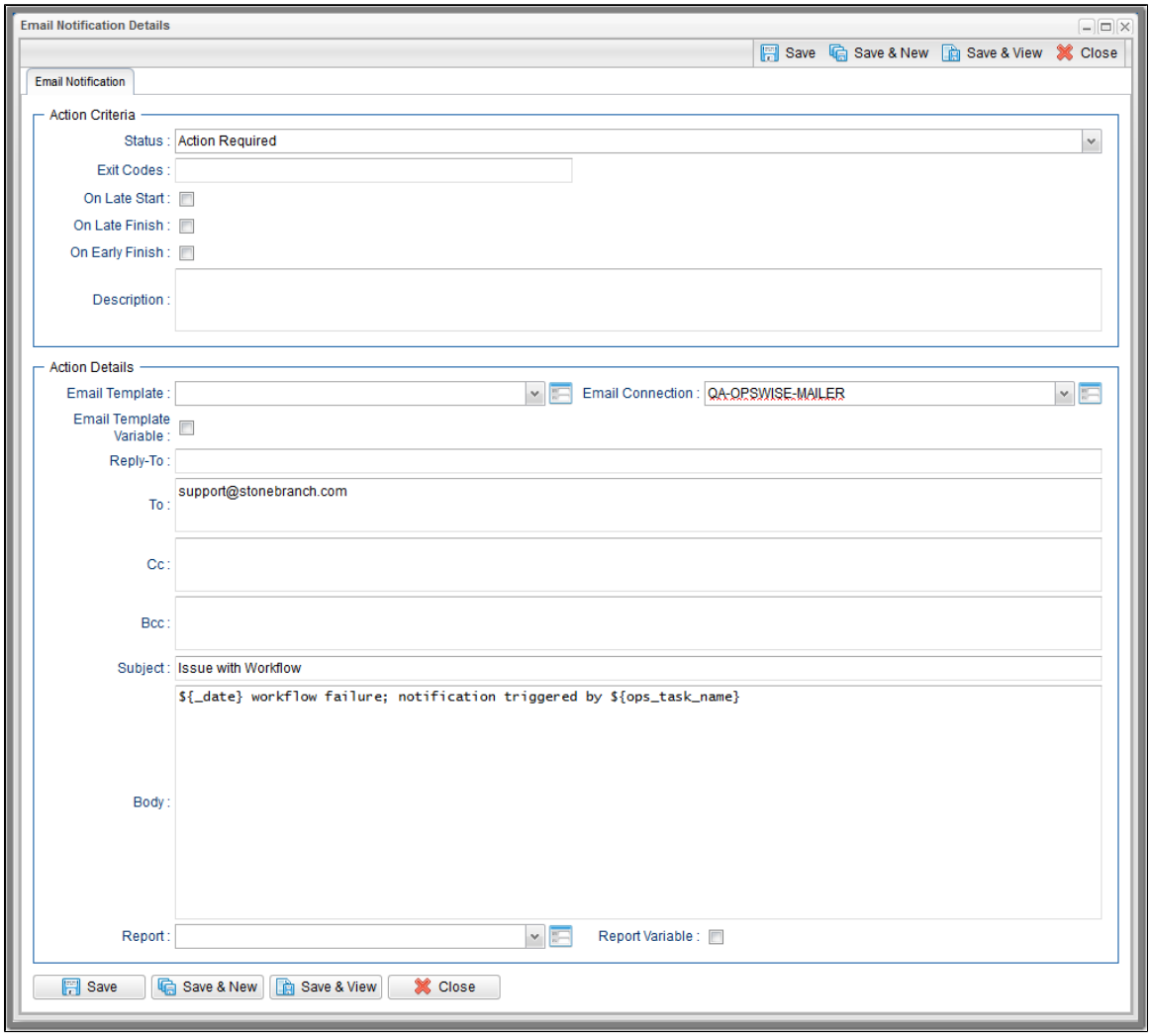
For our Manual task, we are also going to request a warning if the user takes too long to complete it.

Step 1	From the **Automation Center** navigation pane, select **Tasks > Manual Tasks** and click **New**.
Step 2	In the Manual Task Details, enter / select the following values: • **Task Name** = Pause for Manual • **Task Description** = A Manual task run at `${_date()}` (a variable that indicates the date and time the Manual task launches) • **Late Finish** = enabled • **Late Finish Type** = Duration • **Late Finish Duration** = Hours 00 02 00 (2 minutes) 
Step 3	Click the **Save** button.

Step 4 Add a Note:

1. Click the **Notes** tab and then click **New** to display Note Details.
2. Enter the following values:
   - **Title** = Probable database problem
   - **Text** = Make sure database is running.
3. Click **Save**.

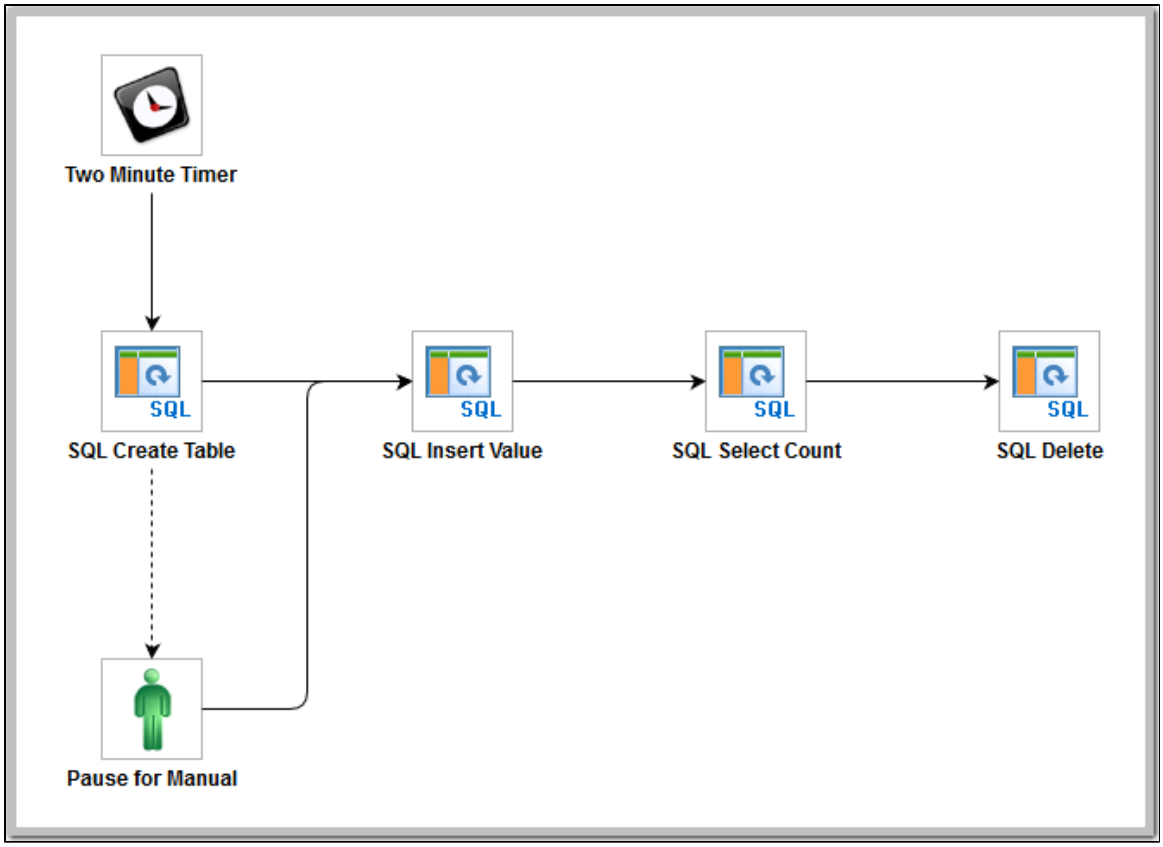


Step 5	Add an Email Notification: 1. Click the **Actions** tab. 2. Click **Email Notification** in the list of Action types. 3. Click the **New** button to display Email Notification Details for a new Email Notification. 4. Enter / select the following values: • **Status** = ACTION REQUIRED • **Email Connection** = (the Email Connection you created earlier) • **To** = (your Email address) • **Subject** = Issue with Workflow • **Body** = ${_date} workflow failure; notification triggered by $\{ops_task_name} | 
Step 6	Click the **Save** button.

Create a Workflow

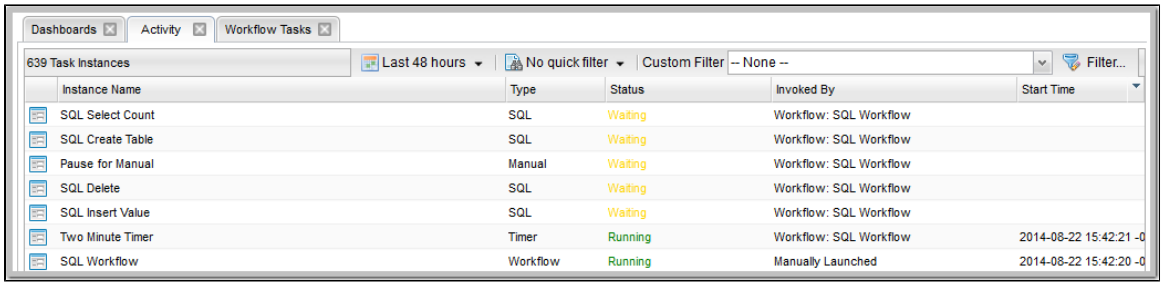
Create a Workflow containing the Timer, SQL, and Manual tasks that you just created.

Step 1	From the **Automation Center** navigation pane, select **Tasks > Workflow Tasks** and click **New**.
Step 2	In the Workflow Task Details, enter the following value: • **Task Name** = SQL Workflow
Step 3	Click the **Save** button, right-click SQL Workflow on the Workflow Tasks list, and then click **Edit Workflow** on the Action menu.

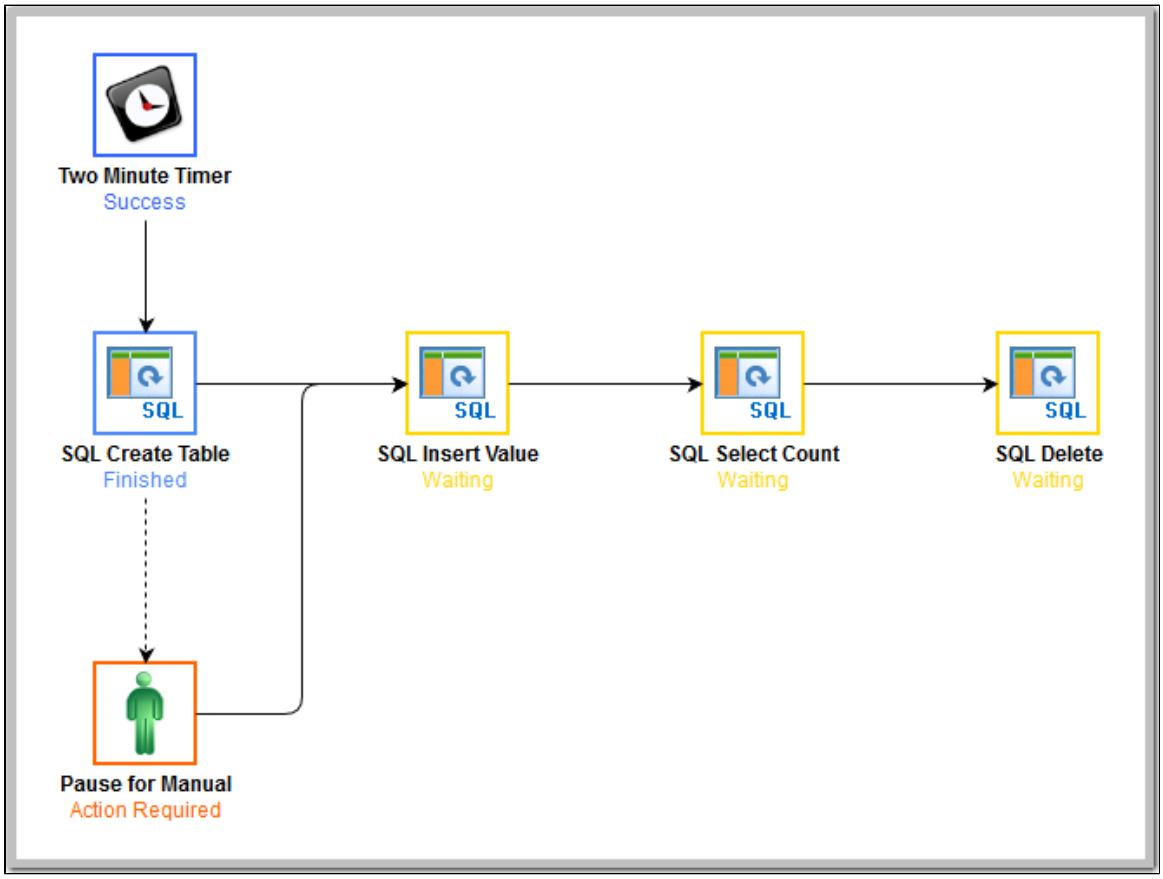
Step 4	In the Workflow Editor, use the Add Task tool to drag the tasks you just created onto the canvas.
Step 5	Organize the tasks and create connections as shown in the following illustration. The Success connectors tell the Controller that if SQL Create Table goes to Success, run Insert SQL Value and the other SQL tasks.
Step 6	Create a conditional path specifying that if SQL Create Table fails, the Controller should run the Pause for Manual task: 1. Right-click the connector between SQL Create Table and Pause for Manual. 2. On the pop-up menu, click **Conditions**. 3. On the **Conditions** pop-up dialog, enable **Failure** and click the **OK** button. Note that the connector is a dotted line, which indicates a Failure connection.
Step 7	On the Workflow Editor toolbar, click the **Save** icon. 

Run the Workflow to Success

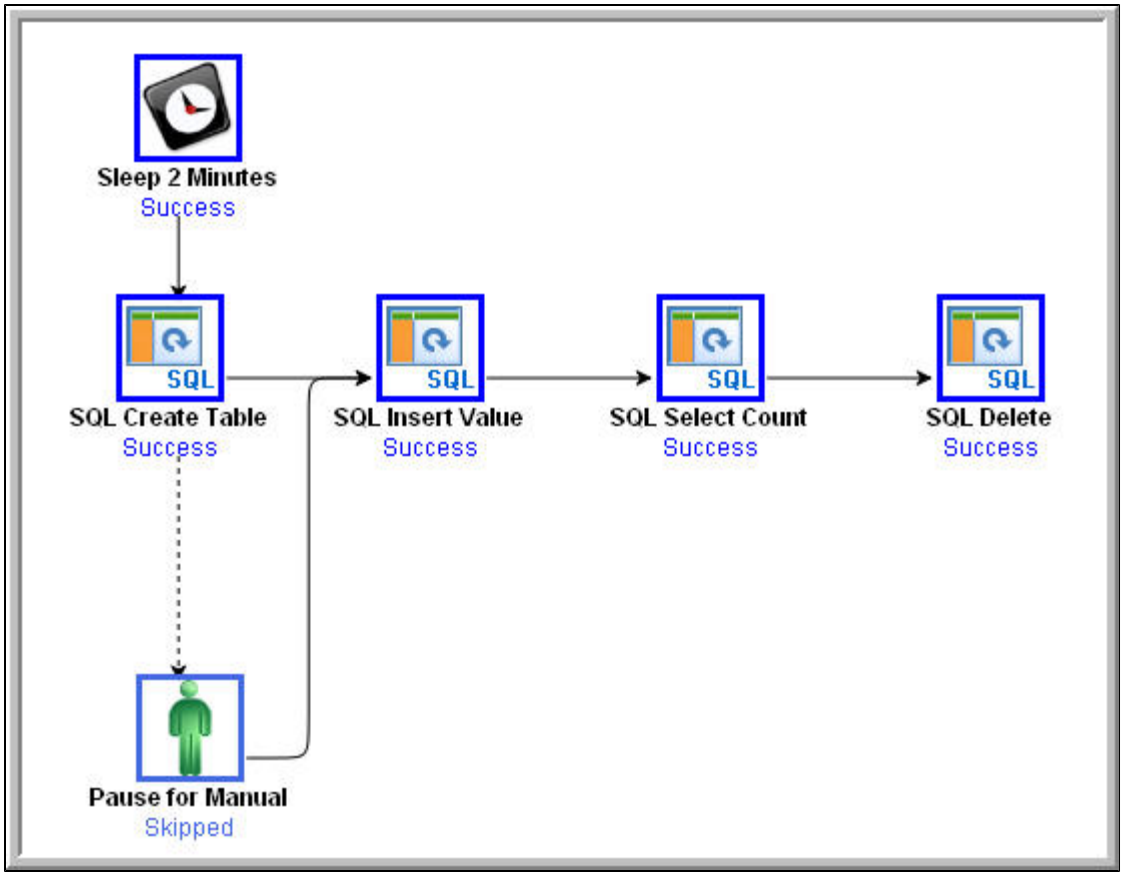
We are going to launch our Workflow and view it from two different perspectives: from the Activity Monitor and the Workflow Monitor.

Step 1	Launch the workflow manually.
Step 2	Display the Activity Monitor. Because the Two Minute Timer task is still running, your display should look similar to this: 
Step 3	Right-click SQL Workflow on the Activity Monitor list to display an Action menu and select **Workflow Task Commands > View**

Workflow. The Workflow Monitor opens and shows progress on the task. The Workflow Monitor updates automatically with each status change.



When the Timer task finishes, the SQL tasks execute. All tasks go to Success and the workflow goes to Success. The only task that did not run is the conditional task, Pause for Manual.

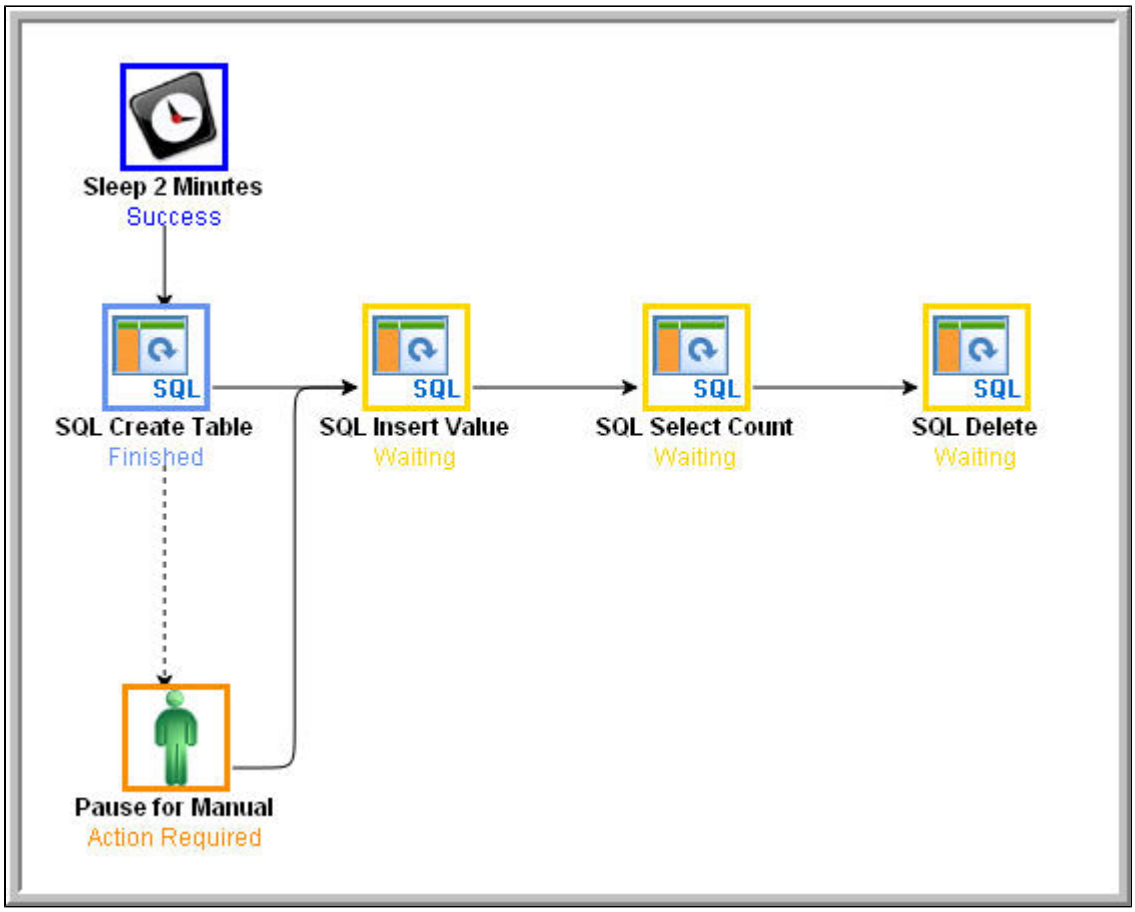


Run the Workflow Down the Conditional Path

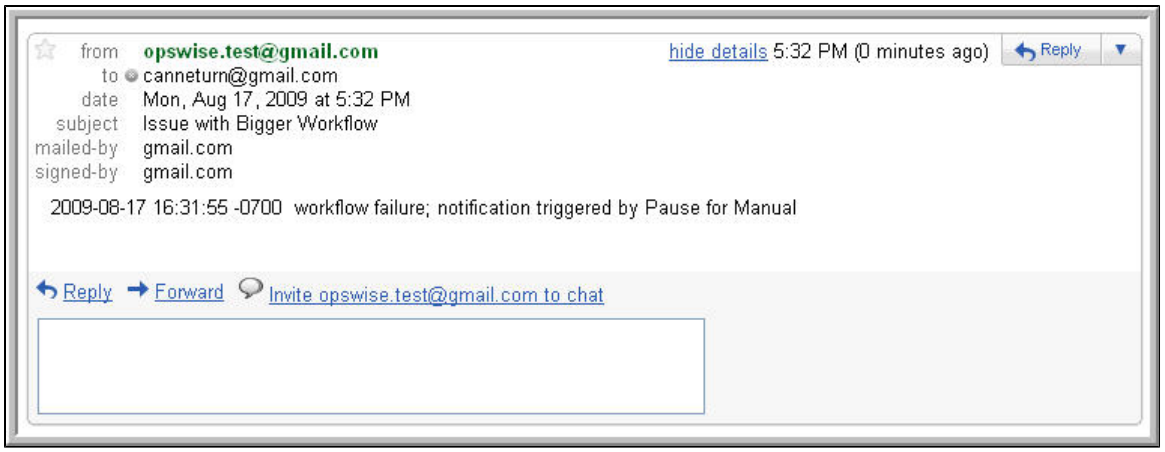
Recall that we inserted a date variable into the INSERT TABLE command. Thus, you can run this workflow every day and get a new table name each day, based on the date. For the purposes of our exercise, assuming you are performing it on the same day you did the previous exercise, the SQL Create Table task will fail this time because the table already exists.

Step 1	Return to the Bigger Workflow task and launch it again.

Step 2 From the Activity Monitor, click the Workflow name to view it from the canvas. This time, the workflow goes down the conditional path. N that when you set up a conditional path, what would normally be a Failure status for the SQL Create Table task becomes a status of Finished. If you ran this task as a standalone task or without the conditional path, its status would say Failed.



Step 3 When the Pause for Manual task launches, it generates the Email Notification we added earlier.

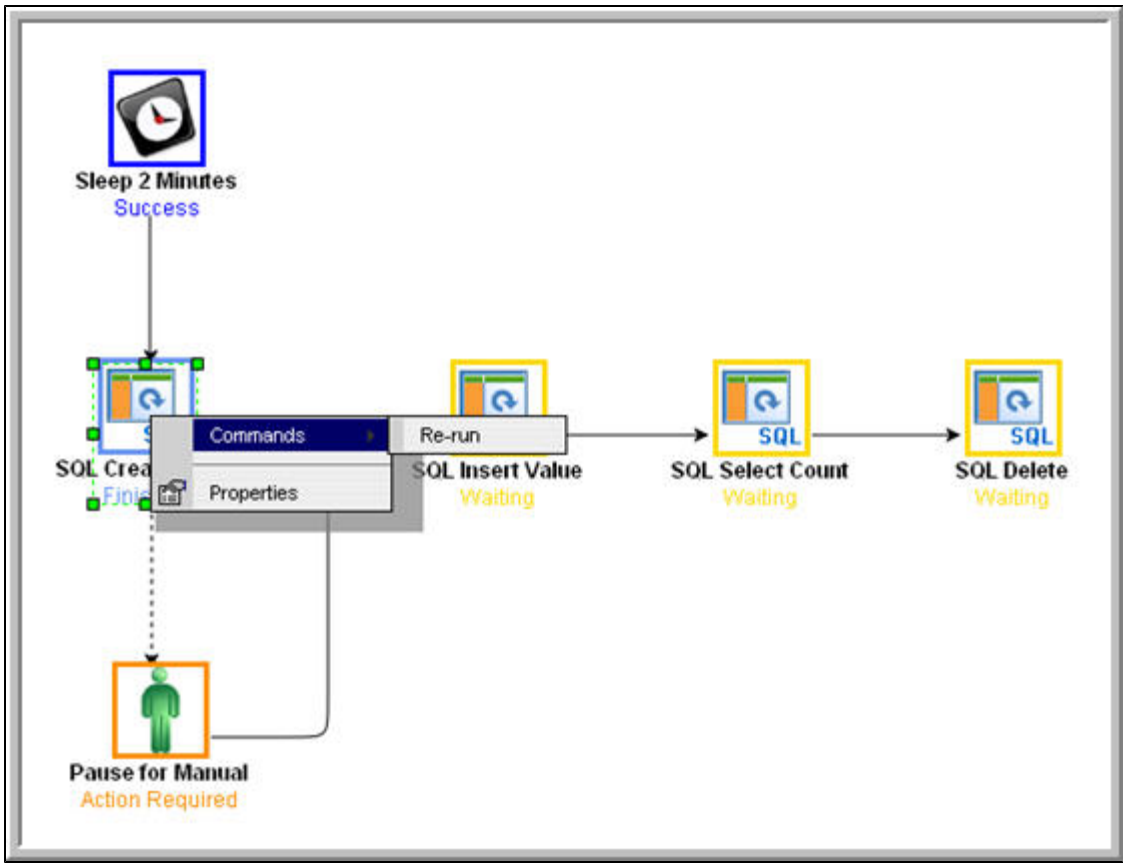


Step 4 The user receives the email, which provides the name of the task that generated it (Pause for Manual in our case). The user might also l running a special Activity Monitor that displays only Manual tasks in the Action Required status. According to our scenario, the user oper the Pause for Manual task and checks the Notes to find out what action he or she is supposed to take. In our case, the Notes say to che the database and bring it back up.

Step 5 At this point, the workflow processing could continue in either of two ways:

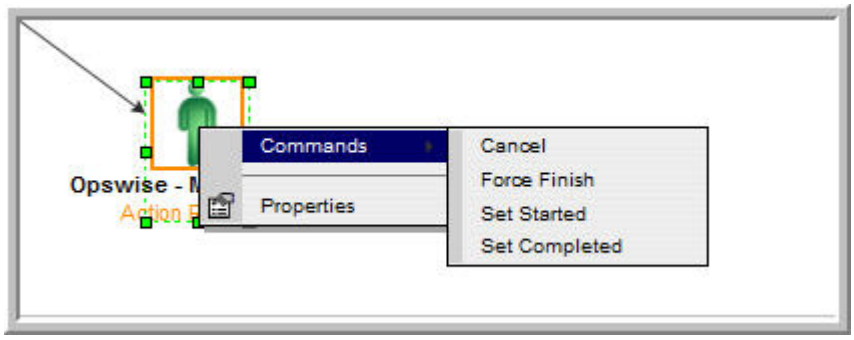
1. Re-run the task that failed and send the workflow down the success path.
2. Set the Pause for Manual task to Completed status and continue the workflow from there.
   We will try both methods.

3. Re-run the failed task; right-click the task and select **Commands > Re-run**. In a real processing situation, this is the method you would use because you need to create the table before you can continue.
    a. Right-click the task that failed and select **Commands > Re-run**.

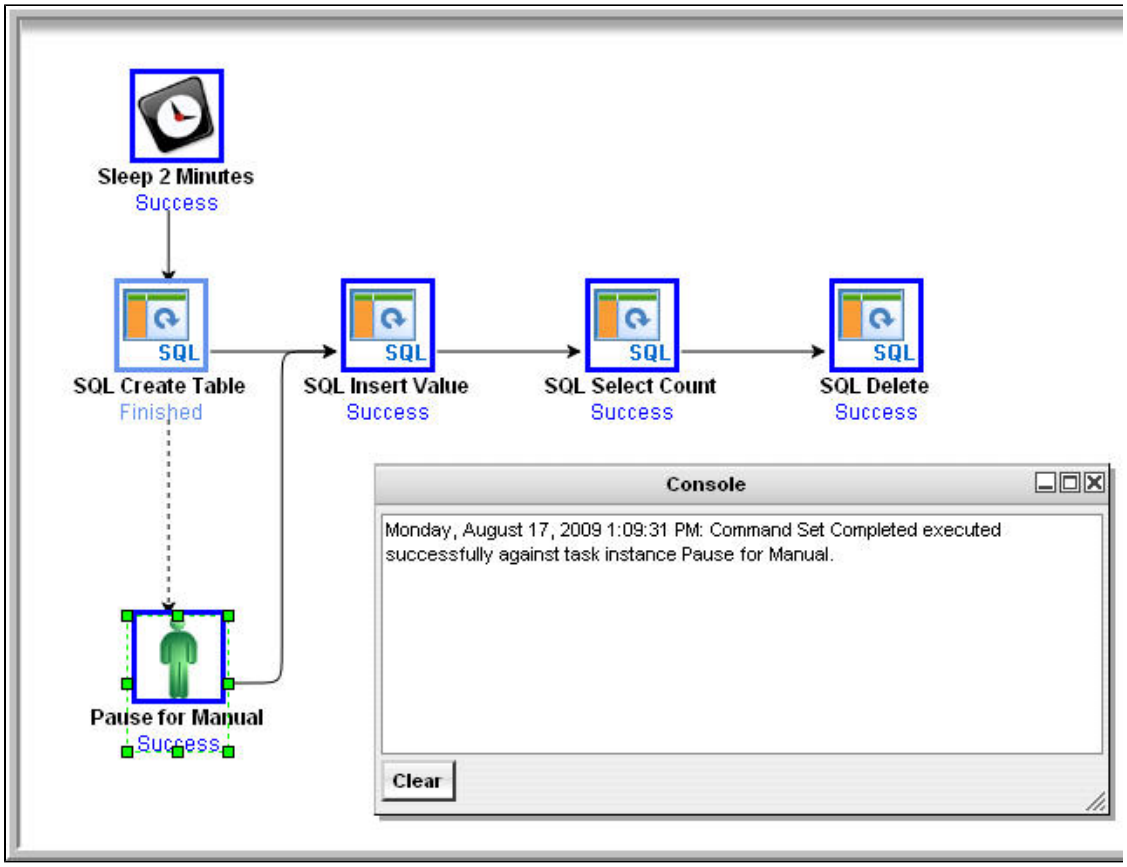


Note that we are still on the same day so the task fails again because the table already exists. In a real processing situation, the task would go to Success and the subsequent tasks would run as expected.

4. Set the Manual task to Complete status. For the purposes of our exercise, we will get the workflow going again by setting the Manual task to complete.
    a. Right-click the Manual task.
    b. Select **Commands > Set Completed**.



The Manual task goes to Success and the rest of the workflow runs.



    c. Because we set a two-minute Late Finish flag on our Manual task and we (presumably) took more than two minutes to complete it, the Manual task has been flagged as a late finish. To view the flag, go to the Activity Monitor and click the Manual task name. In the task instance Details, an enabled Finished Late field now displays, and the Duration field indicates the duration of the task.

For additional information, see:

- Activity Monitor
- Monitoring Workflows
- Database Connections
- SQL Task
- Manual Task
- Email Notification Actions

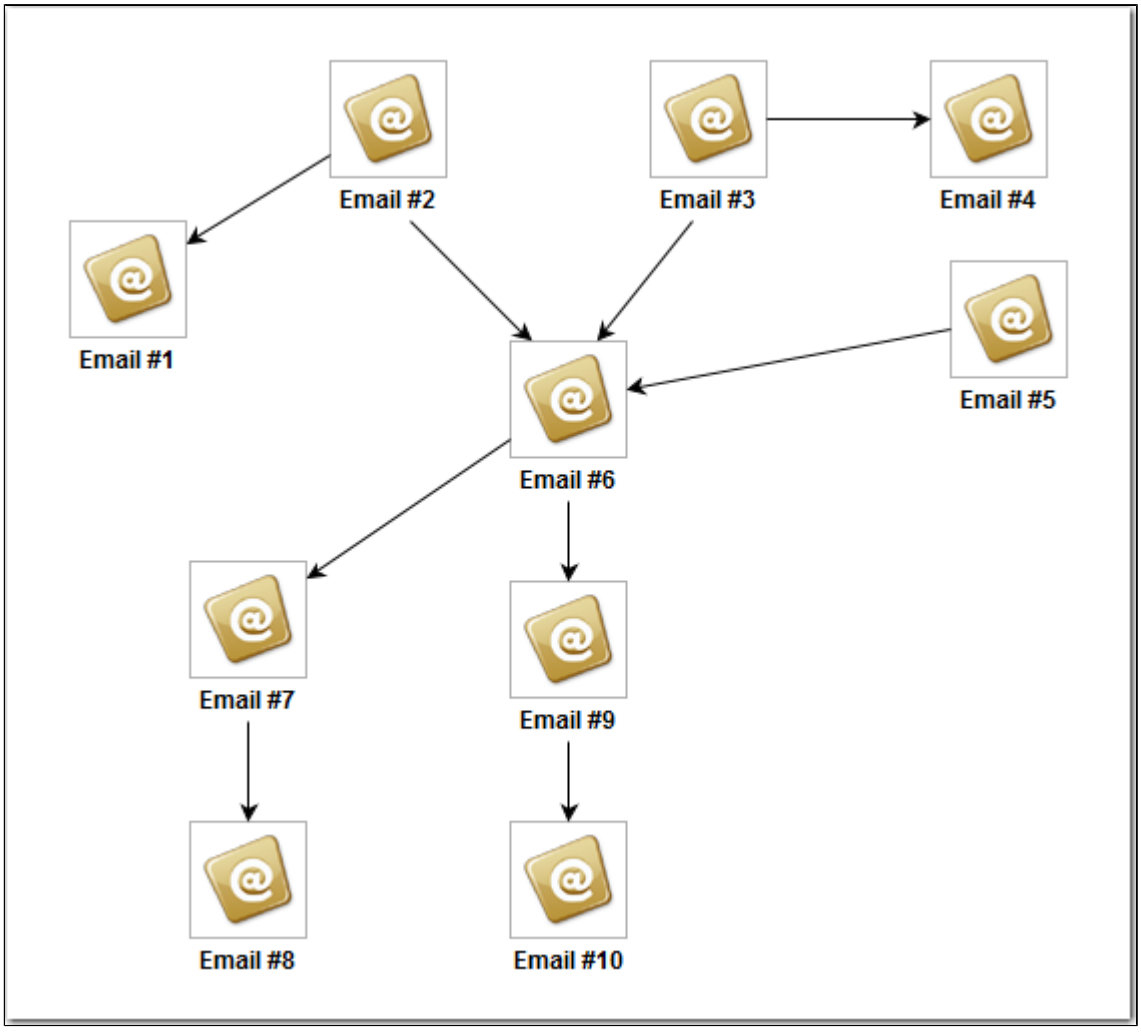
- Creating Notes
- Creating Conditional Paths
- Adding Skip/Run Criteria

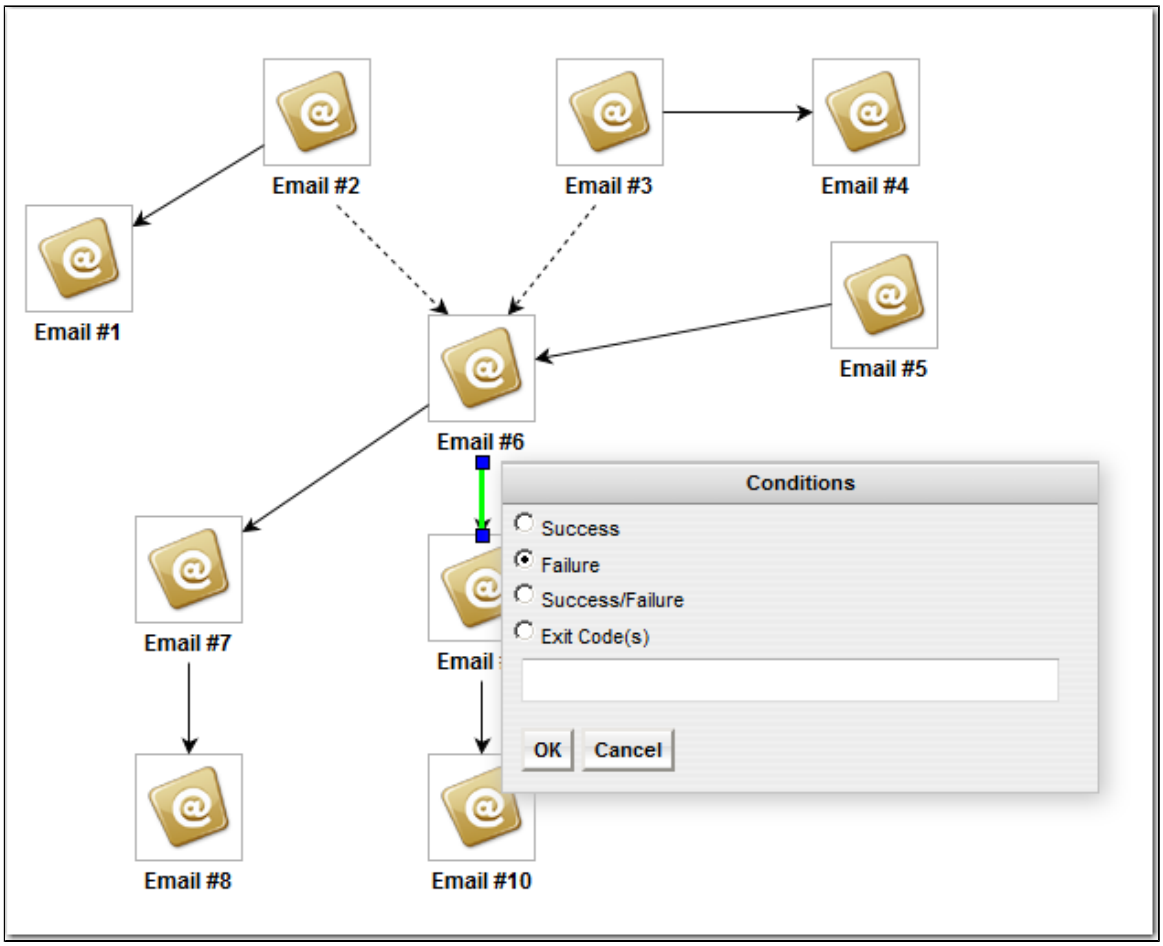
Tutorial - Running a Workflow with Multiple Conditional Paths

In this tutorial, we will create a Workflow containing tasks with multiple predecessors and multiple successors, and specify different conditional paths for those tasks.

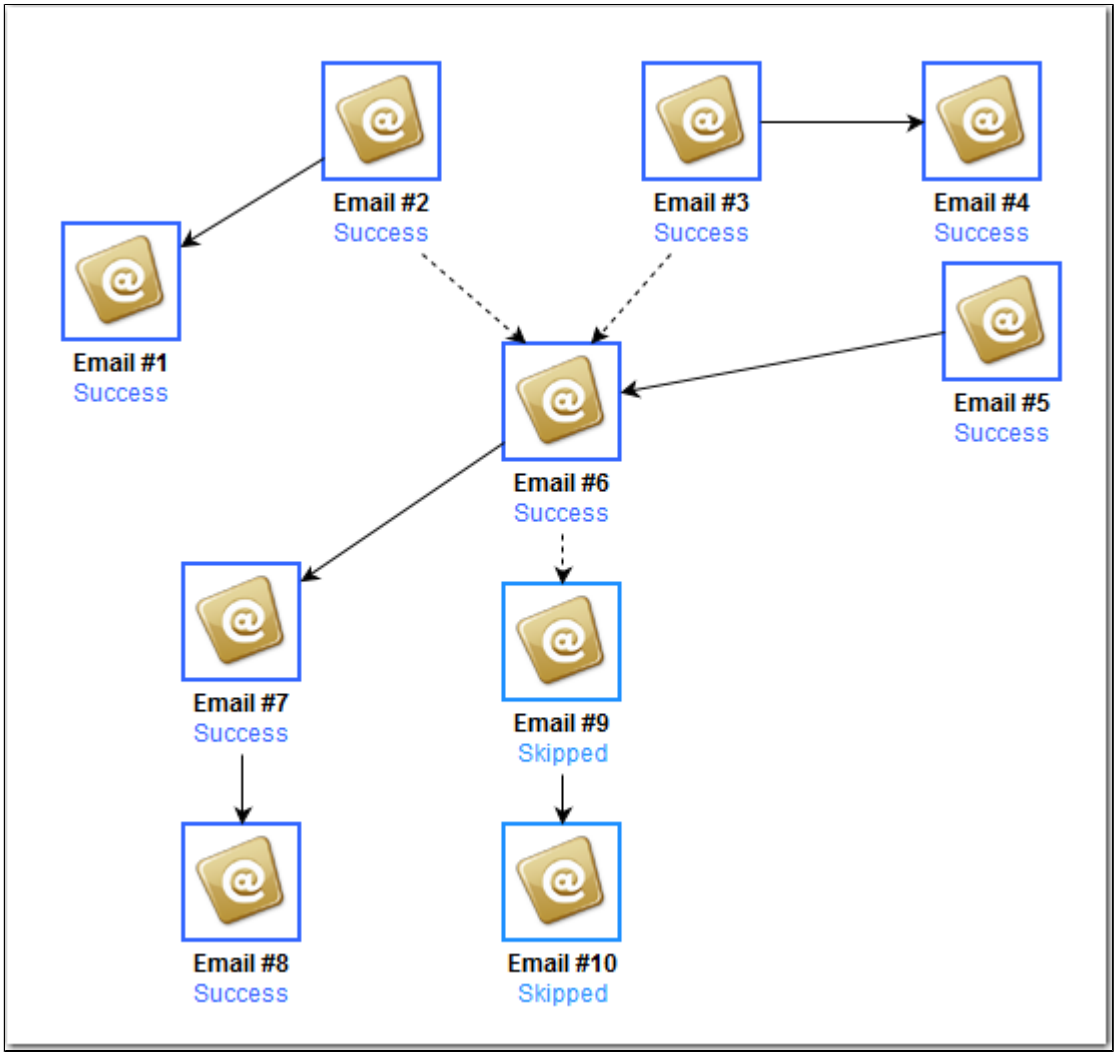
The Workflow will show that:

- A successor task with multiple predecessor tasks needs only one of the predecessor tasks to be Successful in order to run.
- Tasks are skipped if they are in conditional paths not taken.

Step 1	From the Automation Center navigation pane, select **Tasks > Workflow Tasks** and click **New**.
Step 2	In the Workflow Details, name the Workflow **Multiple Paths** and click the **Save** button.
Step 3	Click the **Edit Workflow** button to display the Workflow Editor.
Step 4	Add 10 tasks to the Workflow and specify connections for them, as shown below. (In this example, Email tasks have been added, but you can add any type of task except Timer tasks, since they have only one type of Conditional path: Success.) By default, all connections define a Successful condition; if the predecessor task runs to Success, the successor task will run. 

Step 5	Create a Failure condition between tasks #2 and #6, #3 and #6, and #6 and #9, as shown below: 1. Right-click the connection between them,. 2. Click **Conditions** on the pop-up menu. 3. Select **Failure**. 4. Click **OK**. Connections with Failure conditions displays as dotted lines. 
Step 8	Launch the workflow.
Step 9	In the **Automation Center** navigation pane, select **Task Instances > Activity** to display the Activity Monitor.

Step 10 From the list of Task Instances, click **Multiple Paths**. The Workflow Monitor for that Workflow displays.



The Workflow Monitor shows that task #6 ran even though the conditions for two of its predecessor tasks (#2 and #3) specified that it was to run only if those two tasks failed, because the condition for its #5 predecessor task specified that it was to run if #5 ran successfully, which it did.

It also shows that tasks #9 and #10 were Skipped because the Controller took the Success path for task #6 and ran Tasks #7 and #8.

Tutorial - Running a Workflow with Skipped Criteria

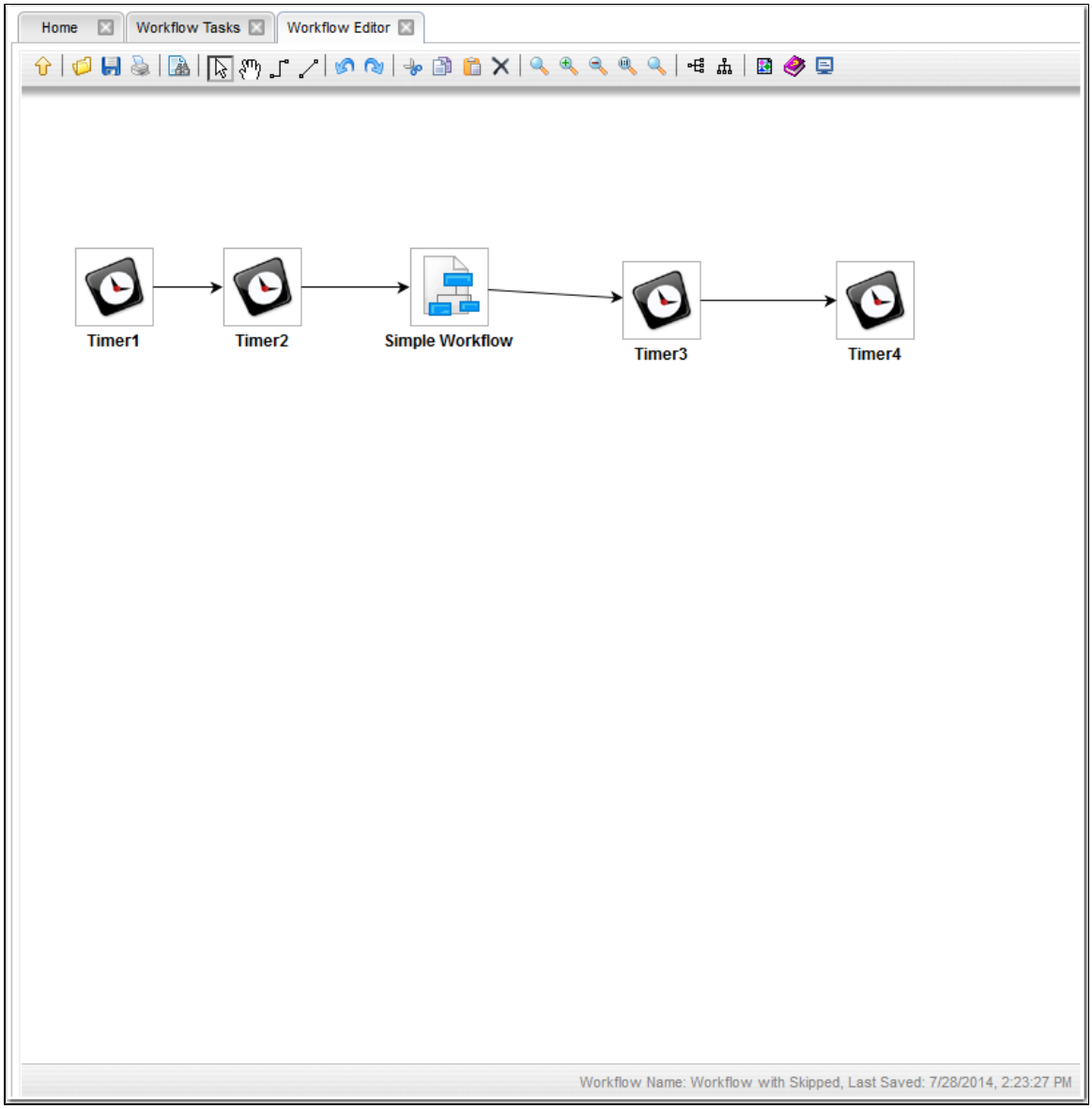
- Introduction
- Create the Daily Workflow
- Run the Daily Workflow
- Check the Skipped Workflow's History

Introduction

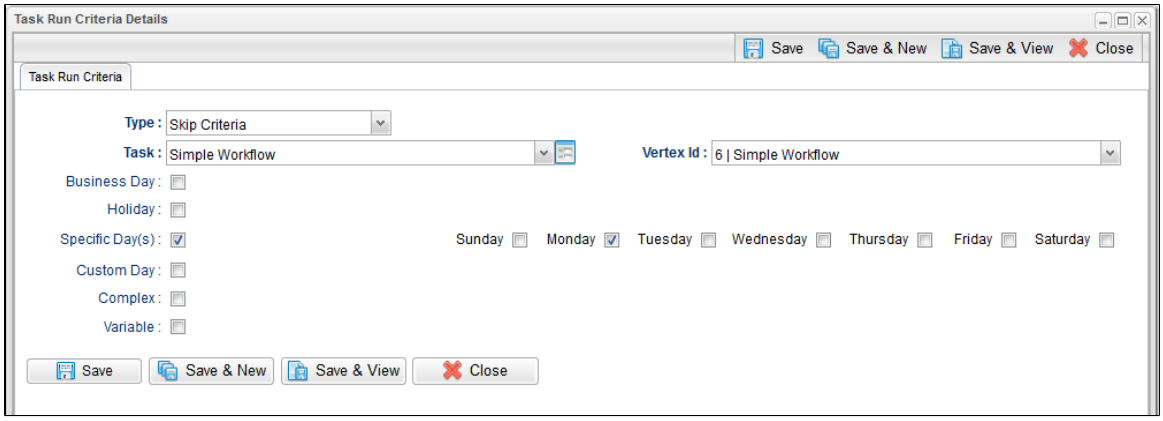
In this exercise, we will create a daily Workflow that includes a task that we want to skip on Fridays. We will also include a Workflow within a Workflow and later check the skipped status of the skipped Workflow.

Create the Daily Workflow

Step 1	Create a Workflow named **Workflow with Skipped** and add the following tasks created in the Creating a Simple Workflow tutorial: • Four Timer tasks: Timer1, Timer2, Timer3, Timer4. • One Workflow task: Simple Workflow.

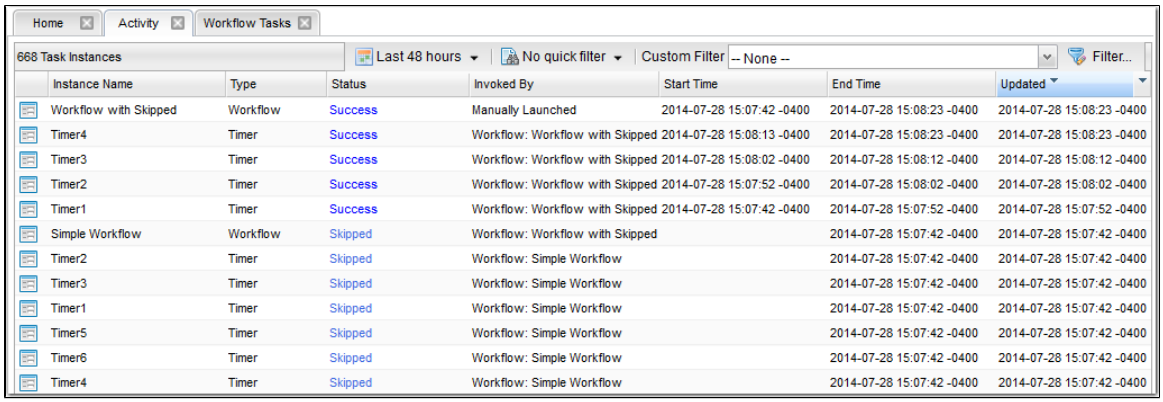
Step 2	Organize the tasks in the Workflow as shown in the following illustration (using all Success conditions) and click the **Save** icon. 
Step 4	Right-click **Simple Workflow** and, on the pop-up menu, click **View/Edit Run Criteria**. The Task Run Criteria list for **Simple Workflow** displays.
Step 5	Click **New** to display Task Run Criteria Details.
Step 6	In the **Type** field, select **Skip Criteria**.
Step 7	Select **Specific Day(s)**.
Step 8	Select the current day (that is, if today is Friday, select Fri). (See Task Run Criteria Field Descriptions for more details.)

Step 9	Click **Save**.



Run the Daily Workflow

Step 1	Right-click anywhere in the Workflow Editor and select **Launch Workflow**.
Step 2	Display the Activity Monitor and note that the sub-workflow (Simple Workflow) and all its sub-tasks have been skipped, as shown in the following example. Note also that this did not impact subsequent tasks, which ran as usual.

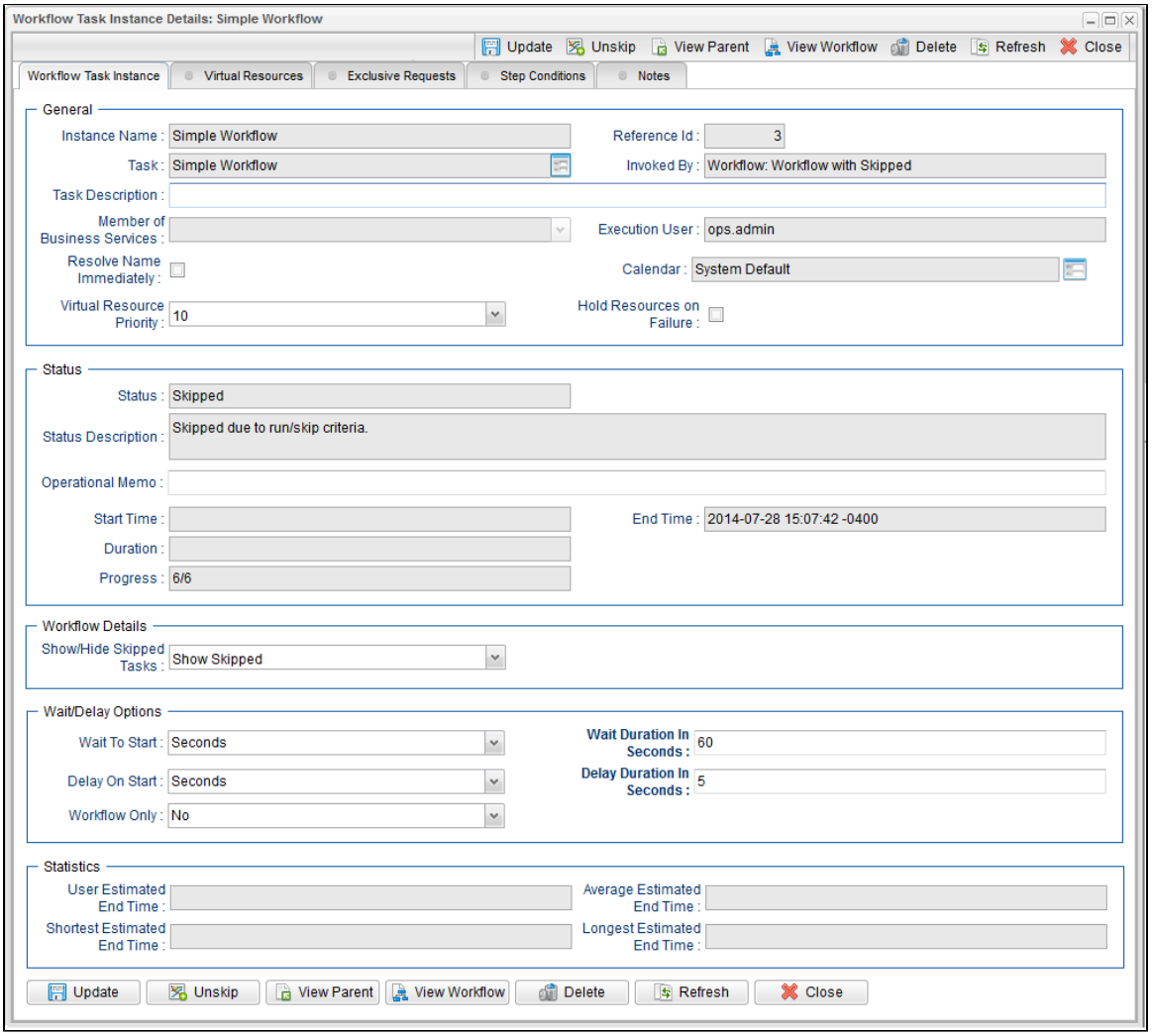


Instance Name	Type	Status	Invoked By	Start Time	End Time	Updated
Workflow with Skipped	Workflow	Success	Manually Launched	2014-07-28 15:07:42 -0400	2014-07-28 15:08:23 -0400	2014-07-28 15:08:23 -0400
Timer4	Timer	Success	Workflow: Workflow with Skipped	2014-07-28 15:08:13 -0400	2014-07-28 15:08:23 -0400	2014-07-28 15:08:23 -0400
Timer3	Timer	Success	Workflow: Workflow with Skipped	2014-07-28 15:08:02 -0400	2014-07-28 15:08:12 -0400	2014-07-28 15:08:12 -0400
Timer2	Timer	Success	Workflow: Workflow with Skipped	2014-07-28 15:07:52 -0400	2014-07-28 15:08:02 -0400	2014-07-28 15:08:02 -0400
Timer1	Timer	Success	Workflow: Workflow with Skipped	2014-07-28 15:07:42 -0400	2014-07-28 15:07:52 -0400	2014-07-28 15:07:52 -0400
Simple Workflow	Workflow	Skipped	Workflow: Workflow with Skipped		2014-07-28 15:07:42 -0400	2014-07-28 15:07:42 -0400
Timer2	Timer	Skipped	Workflow: Simple Workflow		2014-07-28 15:07:42 -0400	2014-07-28 15:07:42 -0400
Timer3	Timer	Skipped	Workflow: Simple Workflow		2014-07-28 15:07:42 -0400	2014-07-28 15:07:42 -0400
Timer1	Timer	Skipped	Workflow: Simple Workflow		2014-07-28 15:07:42 -0400	2014-07-28 15:07:42 -0400
Timer5	Timer	Skipped	Workflow: Simple Workflow		2014-07-28 15:07:42 -0400	2014-07-28 15:07:42 -0400
Timer6	Timer	Skipped	Workflow: Simple Workflow		2014-07-28 15:07:42 -0400	2014-07-28 15:07:42 -0400
Timer4	Timer	Skipped	Workflow: Simple Workflow		2014-07-28 15:07:42 -0400	2014-07-28 15:07:42 -0400

Check the Skipped Workflow's History

You can view a task instance Details to find out why it has a status of Skipped. On the Activity Monitor, click the Details icon in the first column for any task instance to display its task instance Details.

For example:



Note that the Status Description field indicates that **Simple Workflow** was skipped due to run/skip criteria.

For additional information, see:

- Activity display
- Monitoring Workflows
- Adding Skip/Run Criteria

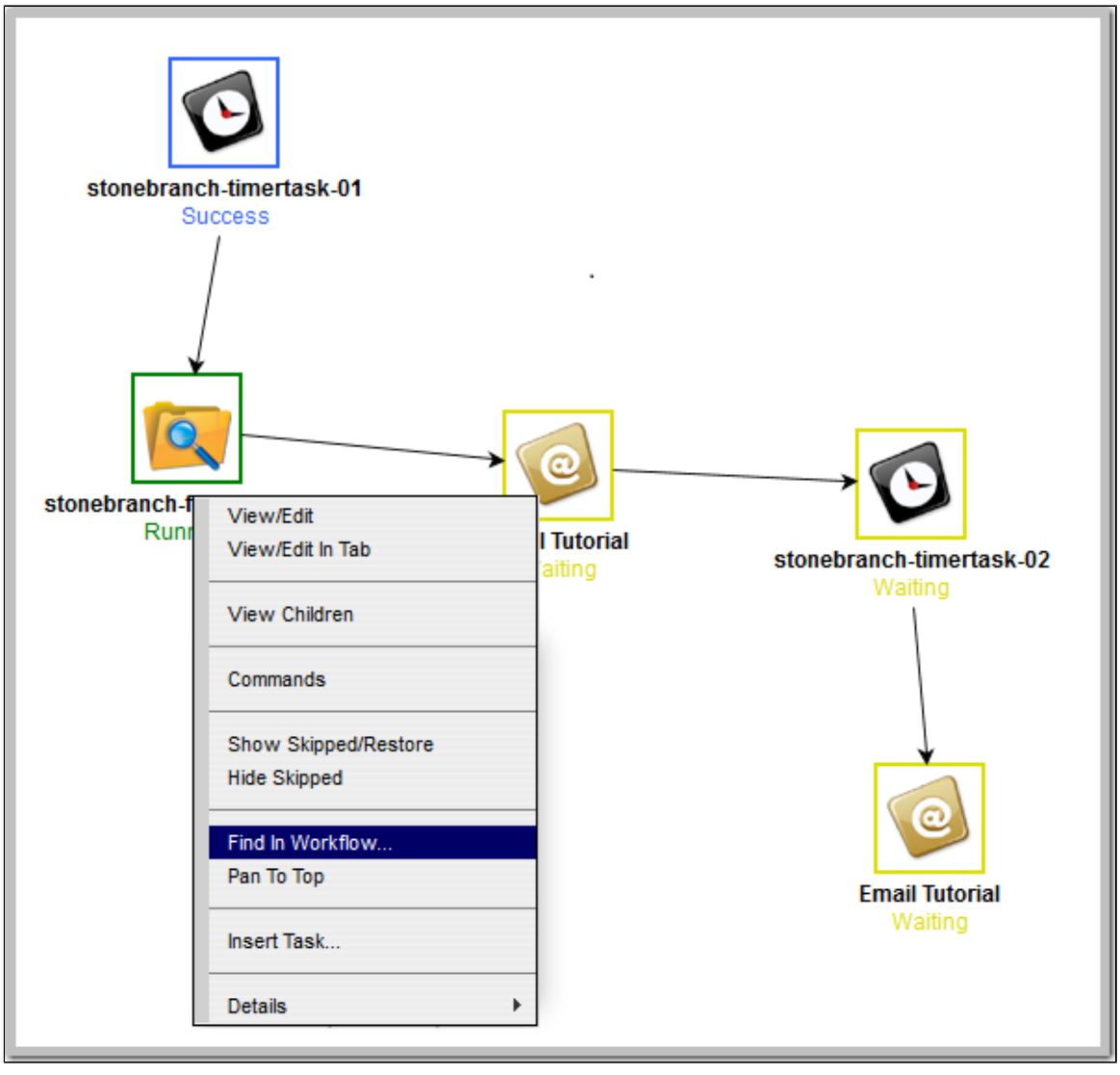
Tutorial - Finding and Inserting Tasks in an Active Workflow

In this exercise, we will run a workflow and:

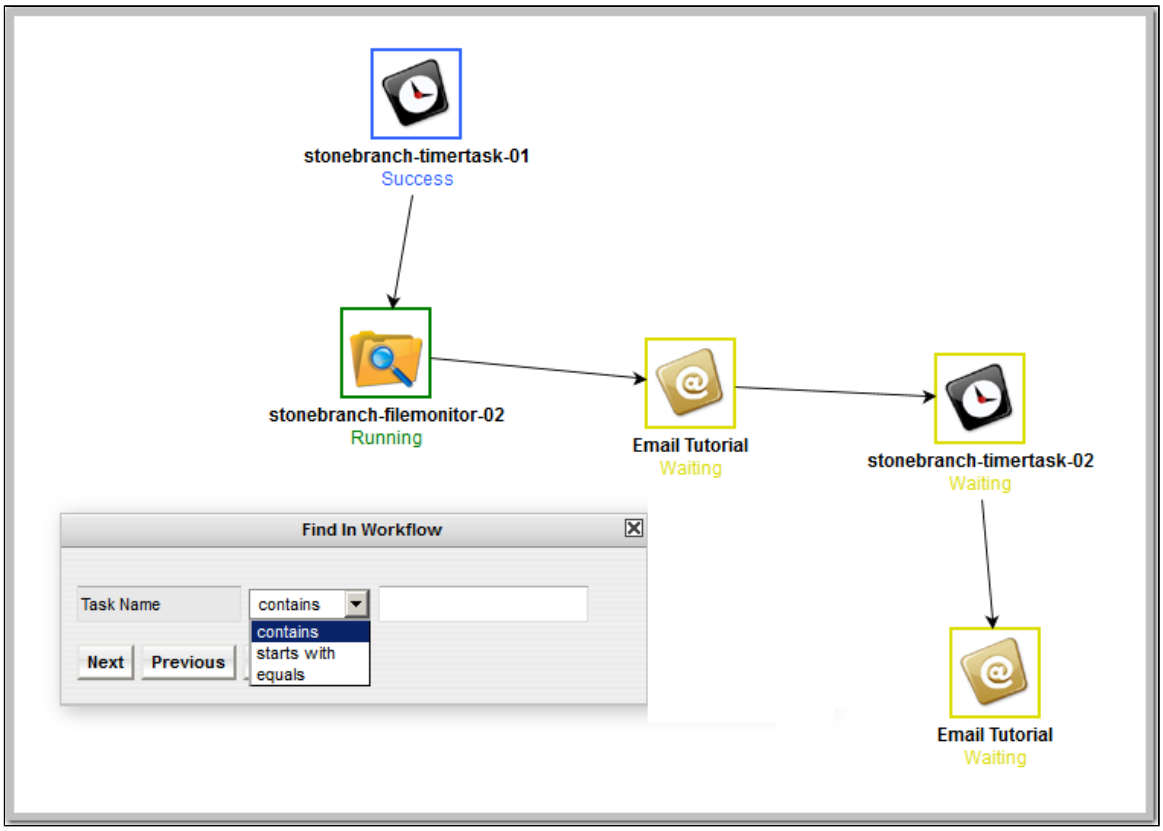
- Find a specific task within the workflow. (You also can find a task in a workflow that has not yet been launched or one that has run and completed.)
- Insert a task as a predecessor to another task in the workflow.
- Insert a task as a successor to another task in the workflow.

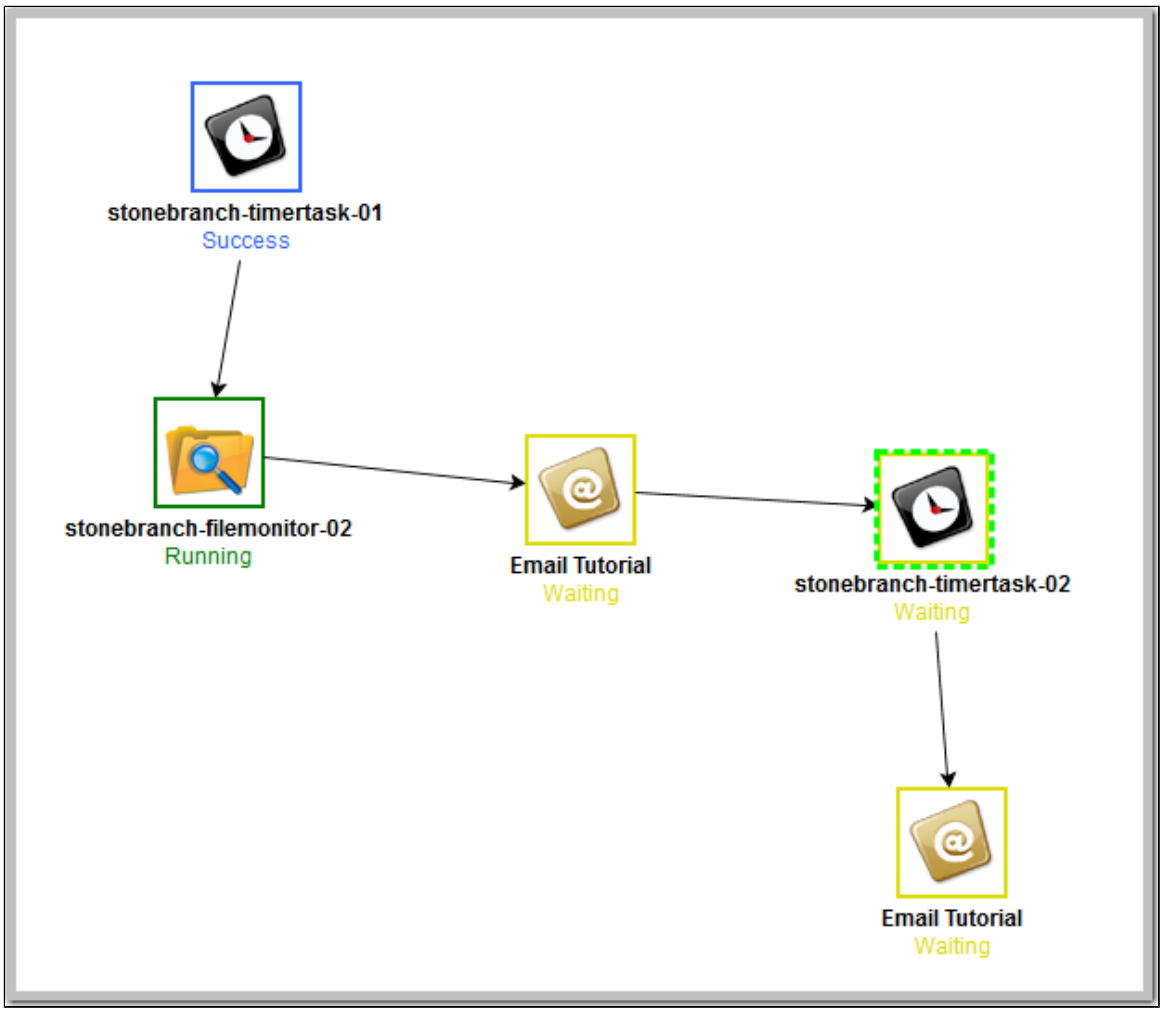
> **Note**
> You may find it easier to run a workflow three different times, once for each step in this exercise.

Step 1	Create a workflow so that all of its tasks cannot display on the Workflow Editor/Monitor at the same time.
Step 2	Launch the workflow and right-click in the Workflow Monitor canvas. A pop-up menu displays.

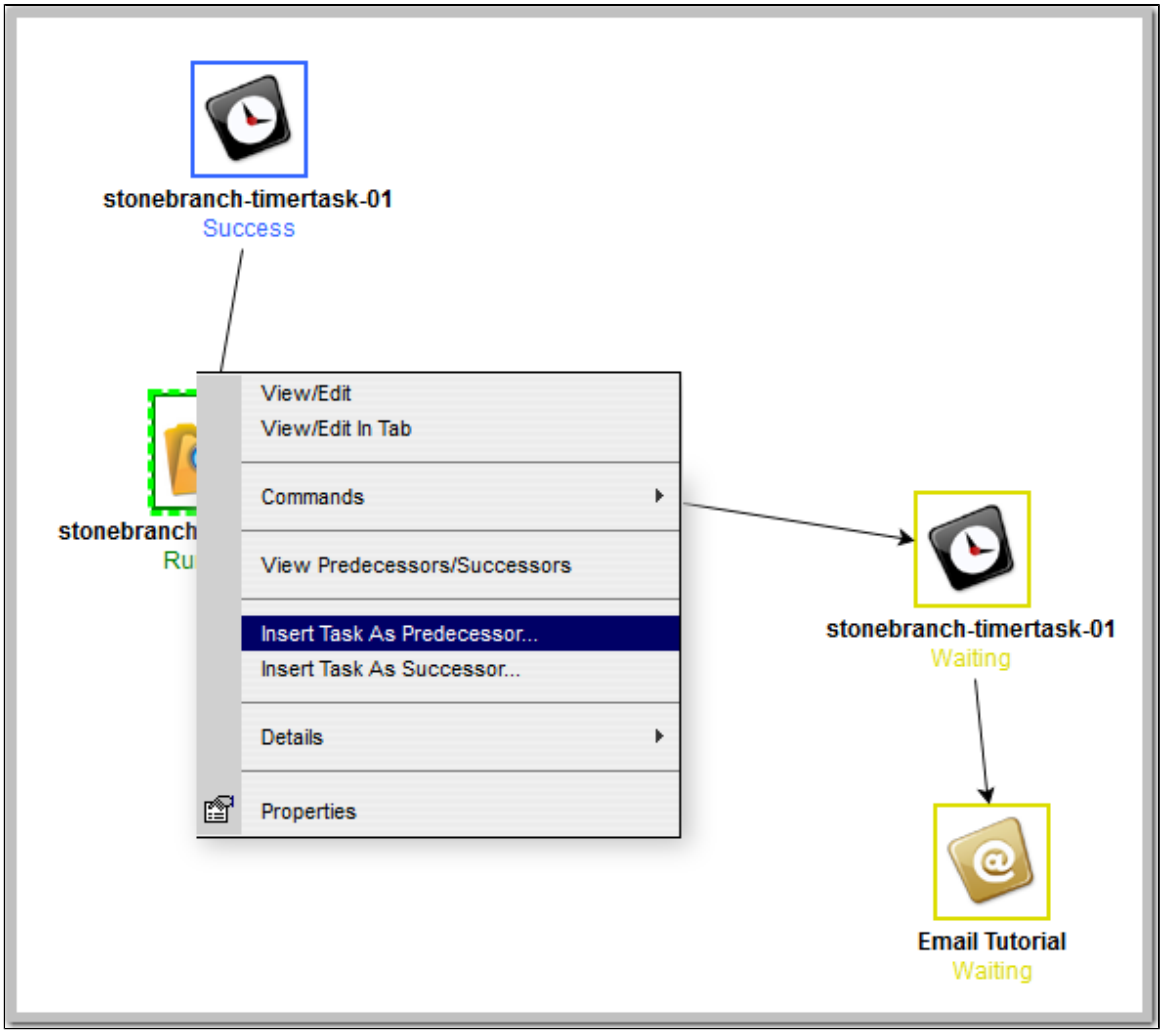


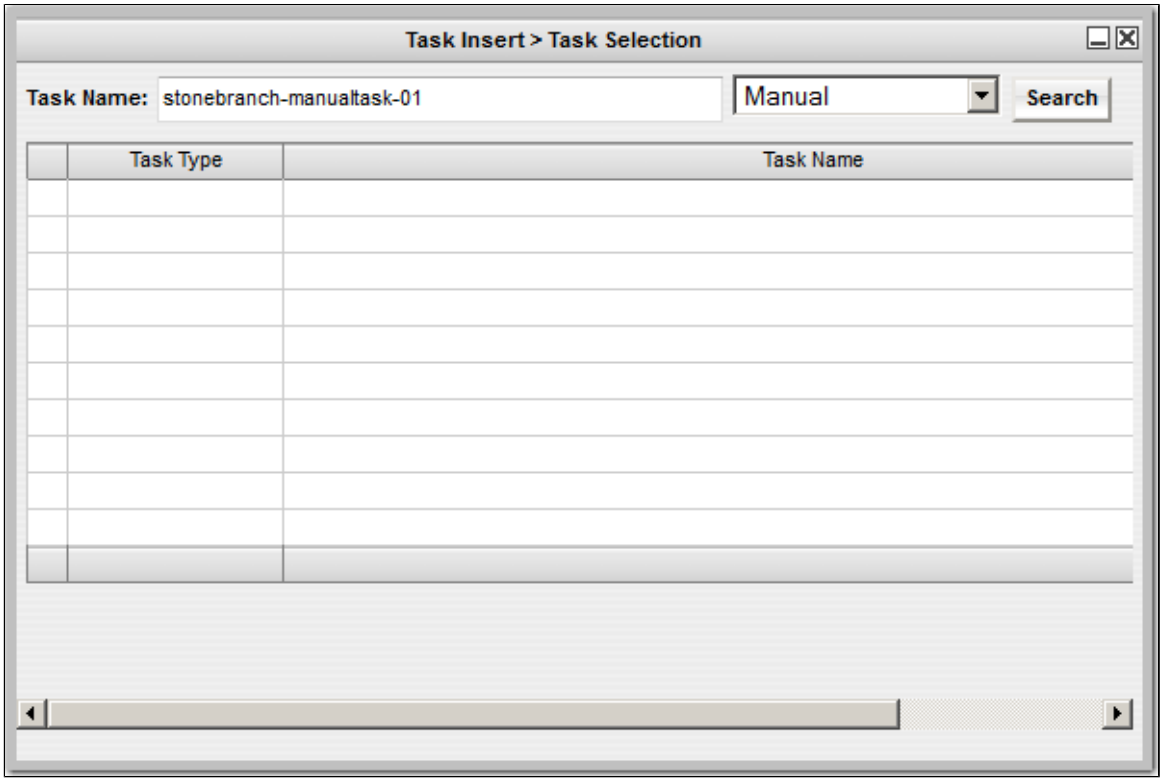
Step 3 Click Find in Workflow... to display the Find in Workflow dialog.

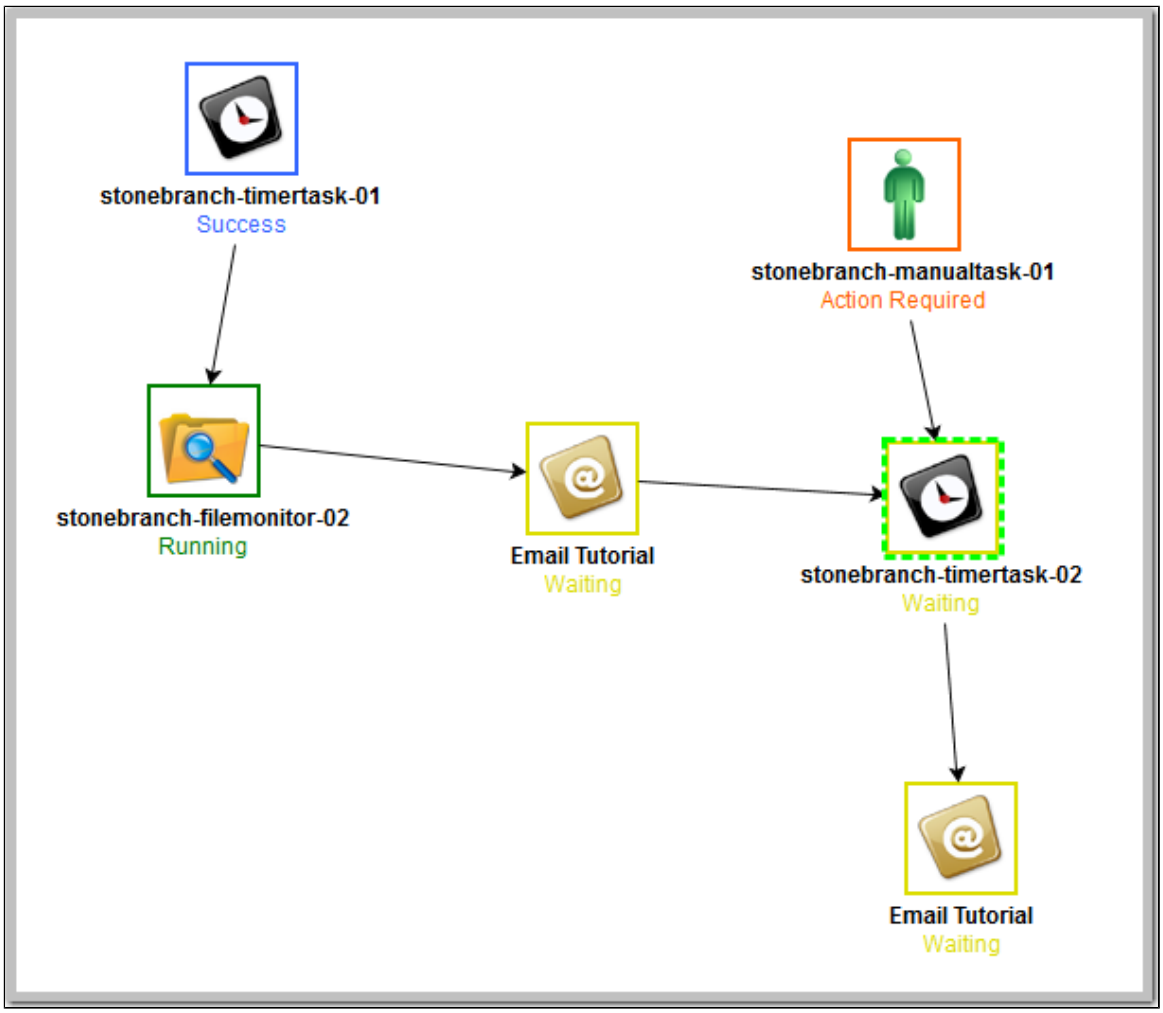


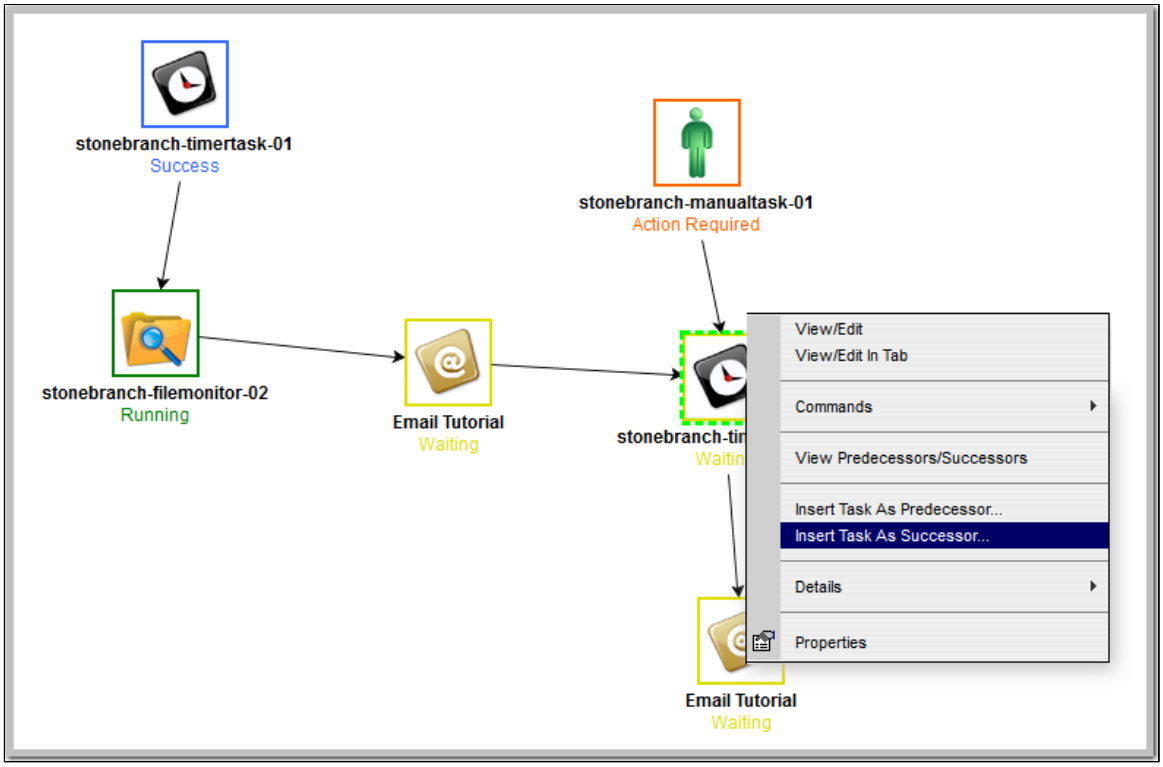
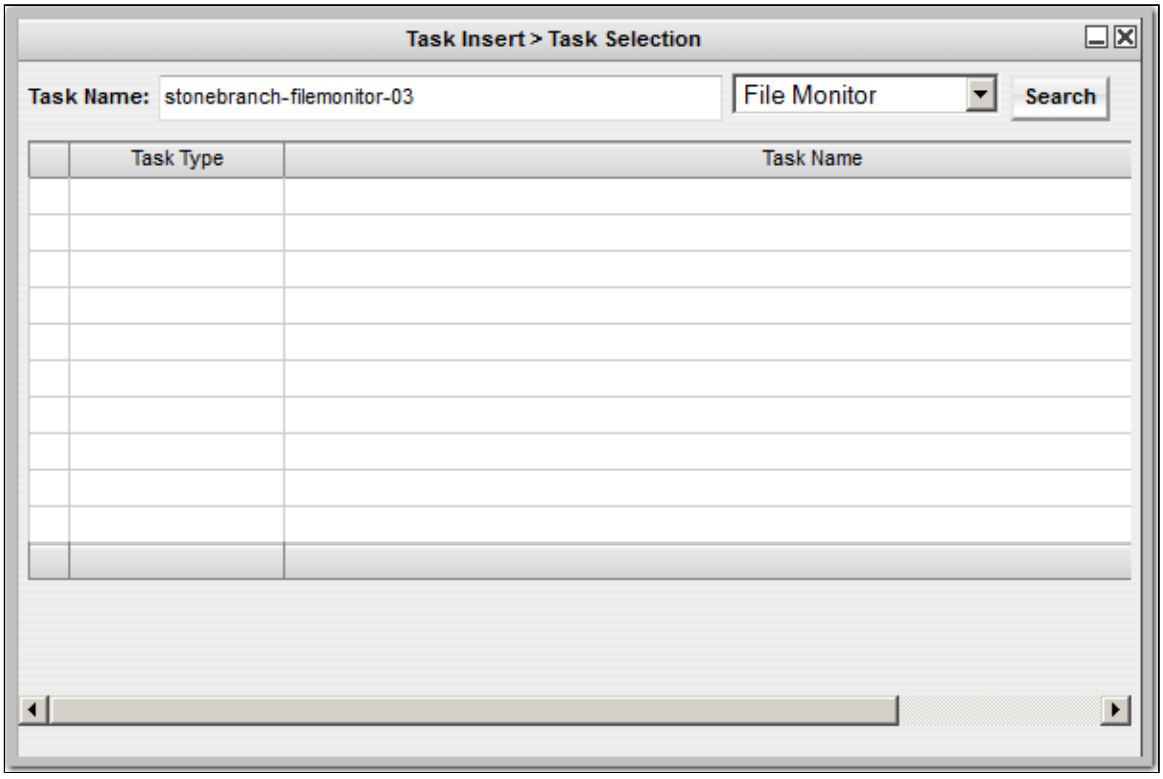
Step 4	Select an operator for the search (**contains**, **starts with**, or **equals**), enter the full or partial name of the task(s) that you want to find, and click OK. The Controller locates and displays the first task within the Workflow that matches the search criteria. Click the **Next** and/or **Previous** buttons to find any other tasks that match the search criteria. 

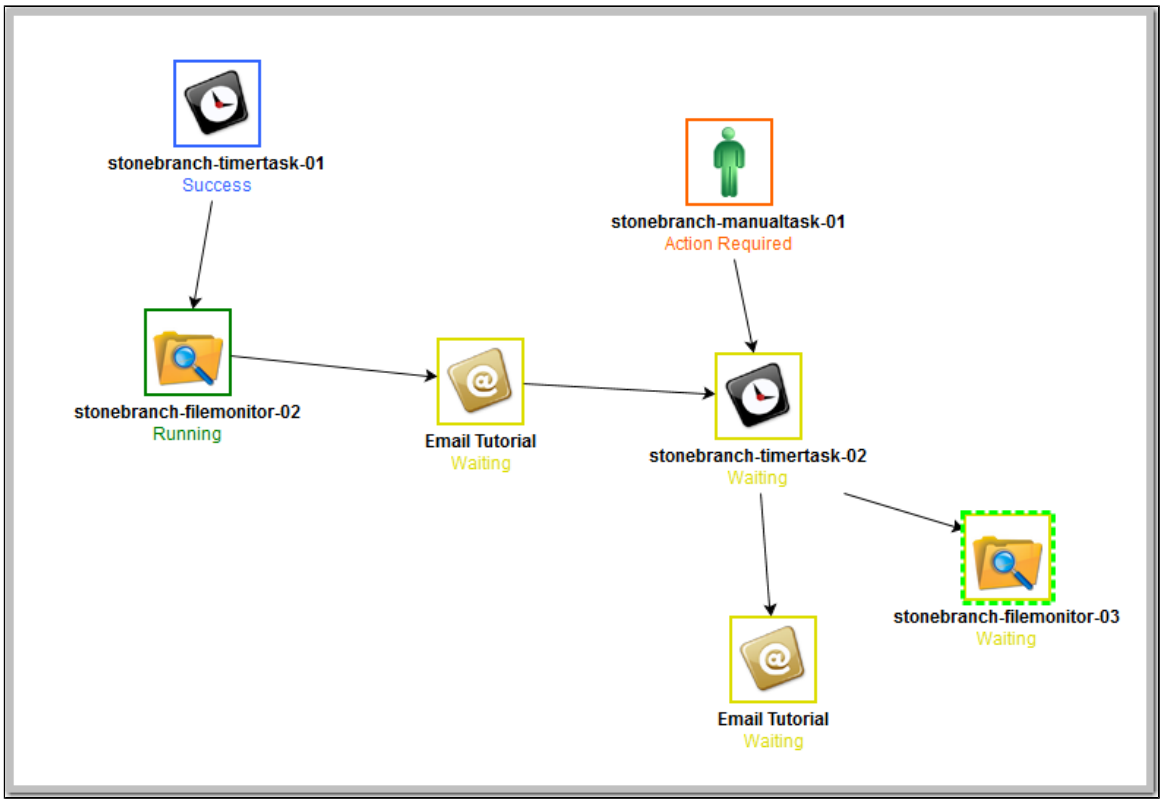
Step 5 Right-click the found task (**stonebranch-timertask-02**) to display a menu of actions available for that task.



Step 6	Click **Insert Task As Predecessor...**. The Task Insert > Task Selection dialog displays. 
Step 7	Find a task you want to insert and drag the task's icon to the workflow canvas. In this example, we have selected **stonebranch-manualtask-01**.

Step 8	The Controller inserts **stonebranch-manualtask-01** as a precedent to the Waiting **stonebranch-timertask-02** task, and **stonebranch-manualtask-01** begins running (as a Manual Task, it requires a user action). When **stonebranch-manualtask-01** completes, **stonebranch-timertask-02** begins running. 

Step 9	Once again, right click **stonebranch-timertask-02** to display a menu of actions available for that task. 
Step 10	Select **Insert Task As Successor...**. The Task Insert > Task Selection dialog displays. 
Step 11	Find the task you want to insert and drag the task's icon to the workflow canvas. In this example, we have selected **stonebranch-filemonitor-03**.

Step 12	The Controller inserts **stonebranch-filemonitor-03** as a successor to **stonebranch-timertask-02**. When **stonebranch-timertask-02** completes successfully, the inserted **stonebranch-filemonitor-03** task begins running. 

For additional information, see:

- Finding a Task in a Workflow
- Inserting a Task in a Workflow
- Searching for and Adding Tasks

Tutorial - Skipping, Unskipping, and Showing-Hiding Skipped Task Instances

You can skip (and unskip) individual task instances and task instances within a workflow that have been launched but have not yet started to run. For skipped tasks within a workflow, you can choose to show or hide those tasks in the Workflow Monitor.

You also can skip a task instance so that all dependent task instances of that task instance automatically are skipped as well.

Although there are several methods for skipping, unskipping, and showing/hiding skipped task instances, in this exercise we will:

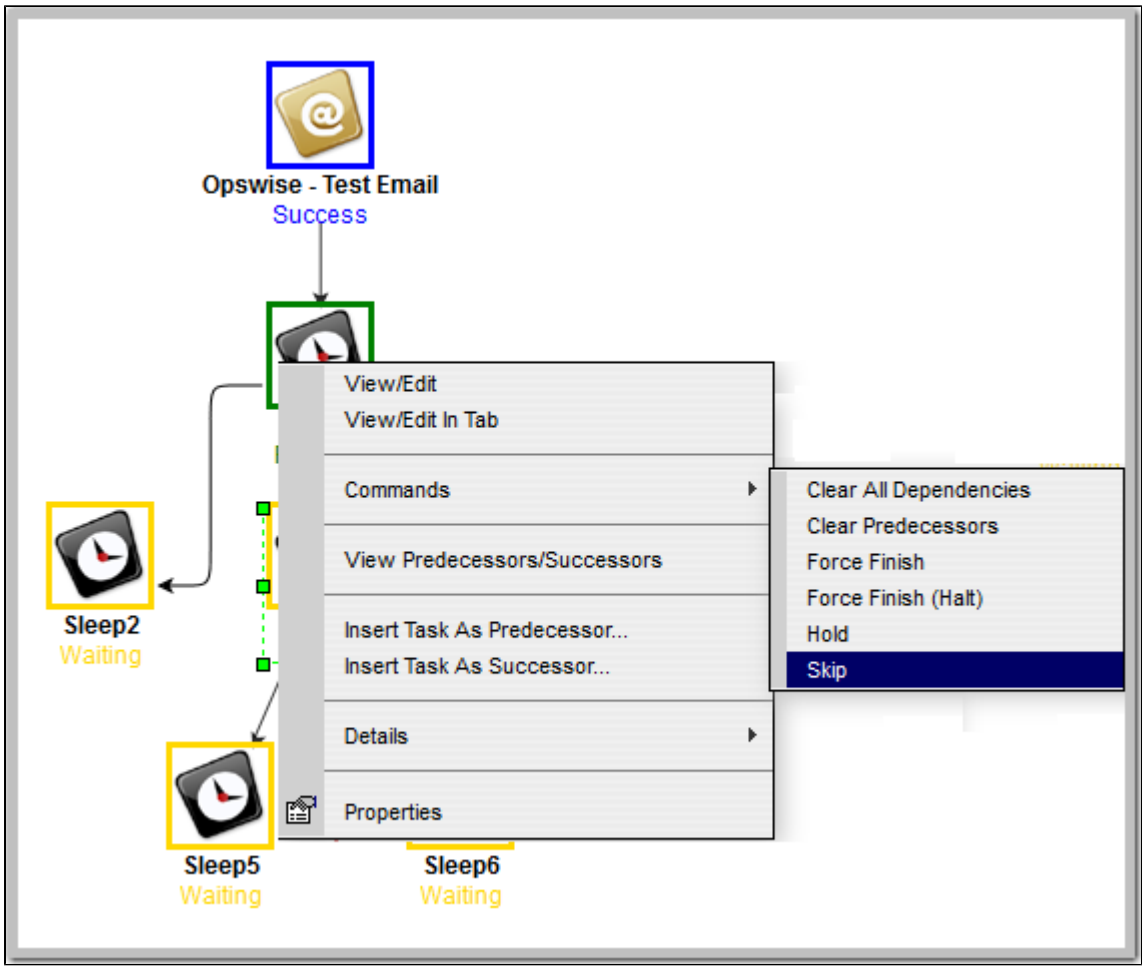
- Skip a task instance in a workflow.
- Unskip a previously skipped task instance in a workflow.
- Show and hide a skipped task instance in a workflow.

Note
There also are methods for skipping a task and showing/hiding a skipped task before it becomes a task instance (that is, before it or the workflow in which it resides has been launched). See the links at the bottom of this page.

Step 1	From the Automation Center navigation pane, select **Tasks > Workflow Tasks**. The Workflow Tasks list displays a list of all workflow tasks.
Step 2	Right-click **Simple Workflow** (created in the Creating a Simple Workflow tutorial), and on the Action menu, click **Launch Task**.
Step 3	On the Activity Monitor, select Active Workflow Task Instances from the drop-down list.
Step 4	Click **Simple Workflow**. The Workflow Monitor displays for this running workflow.

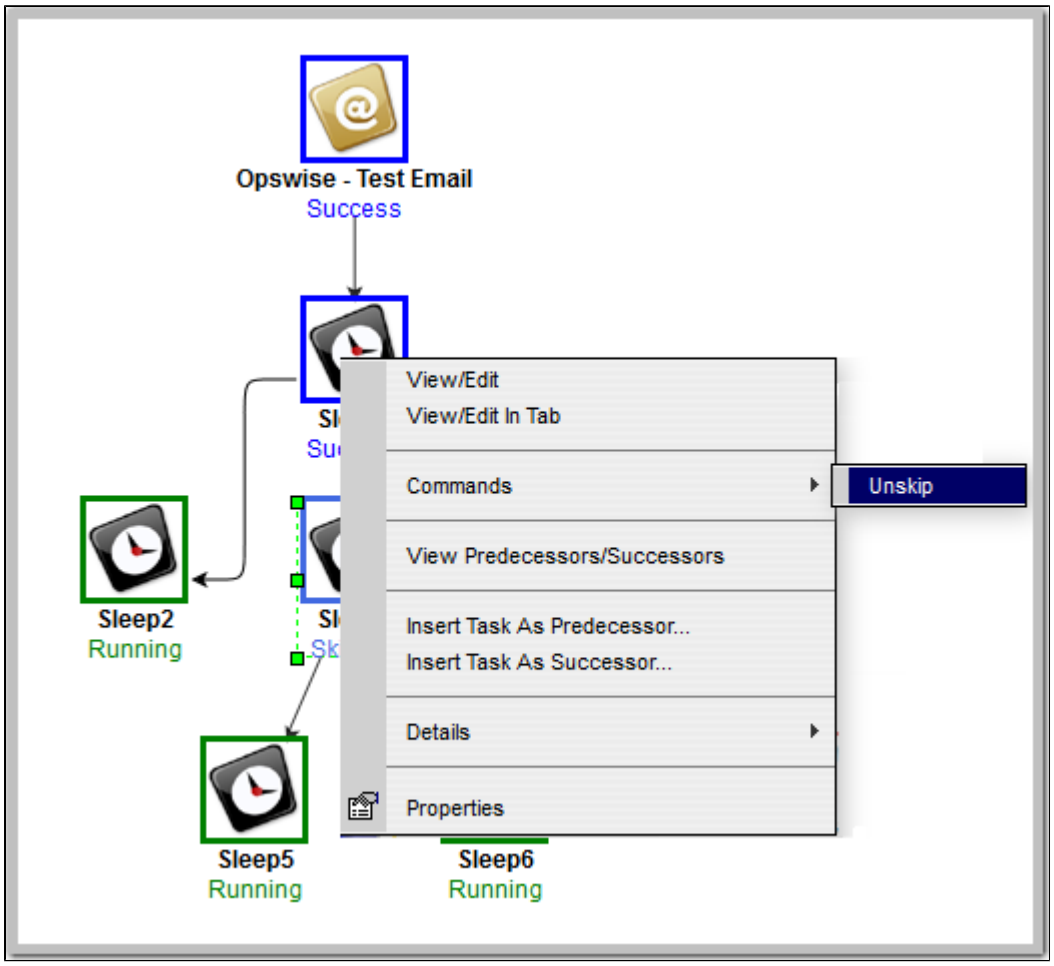
Step 5 Right-click the Sleep3 task while it is in Waiting status and, from the pop-up menu that displays, click Commands / Skip.

> **Note**
> If you wanted to skip the Sleep3 task and its dependent Sleep5 and Sleep6 tasks, you would click Commands / Skip Path.



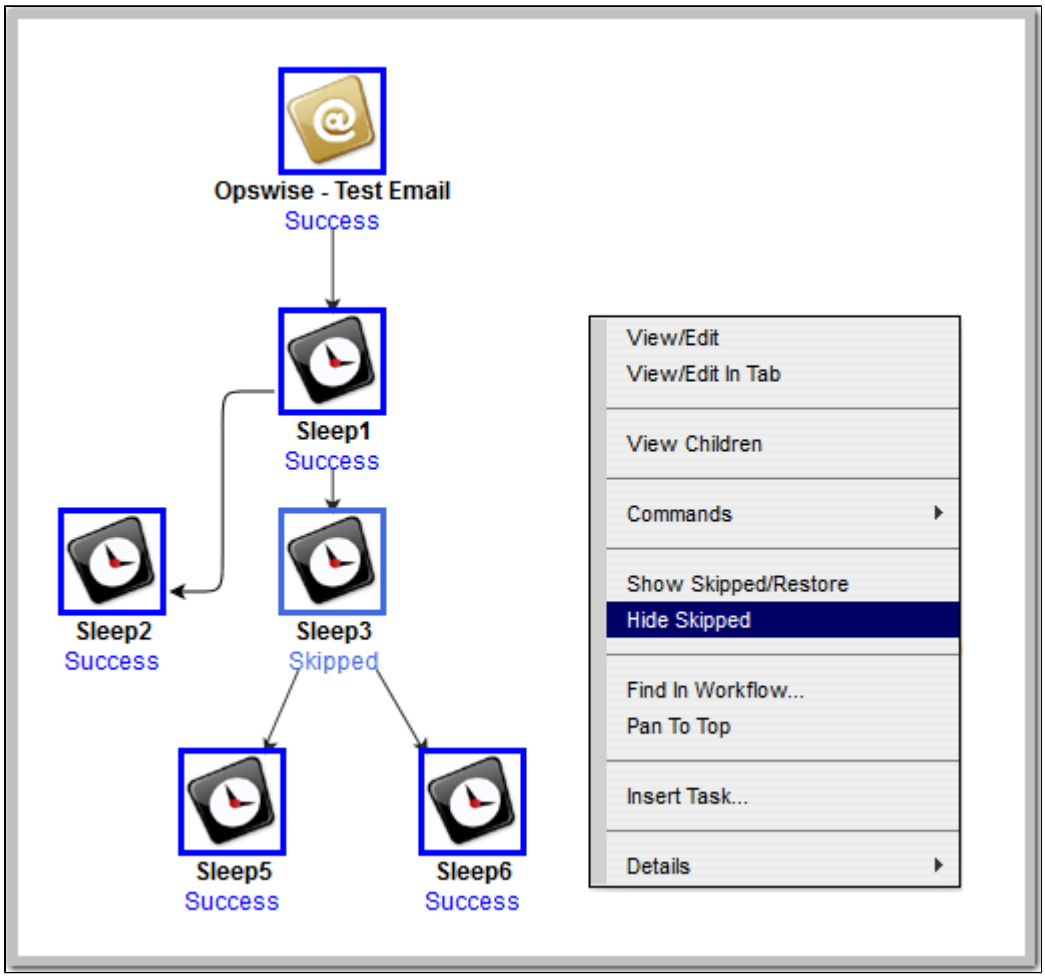
The Sleep3 status changes from Waiting to Skipped. When Sleep1 completes successfully, Universal Controller will skip Sleep3 and start running Sleep5 and Sleep6.

Step 6 To un-skip Sleep3, right-click it and, from the pop-up menu that displays, click Commands / Unskip.

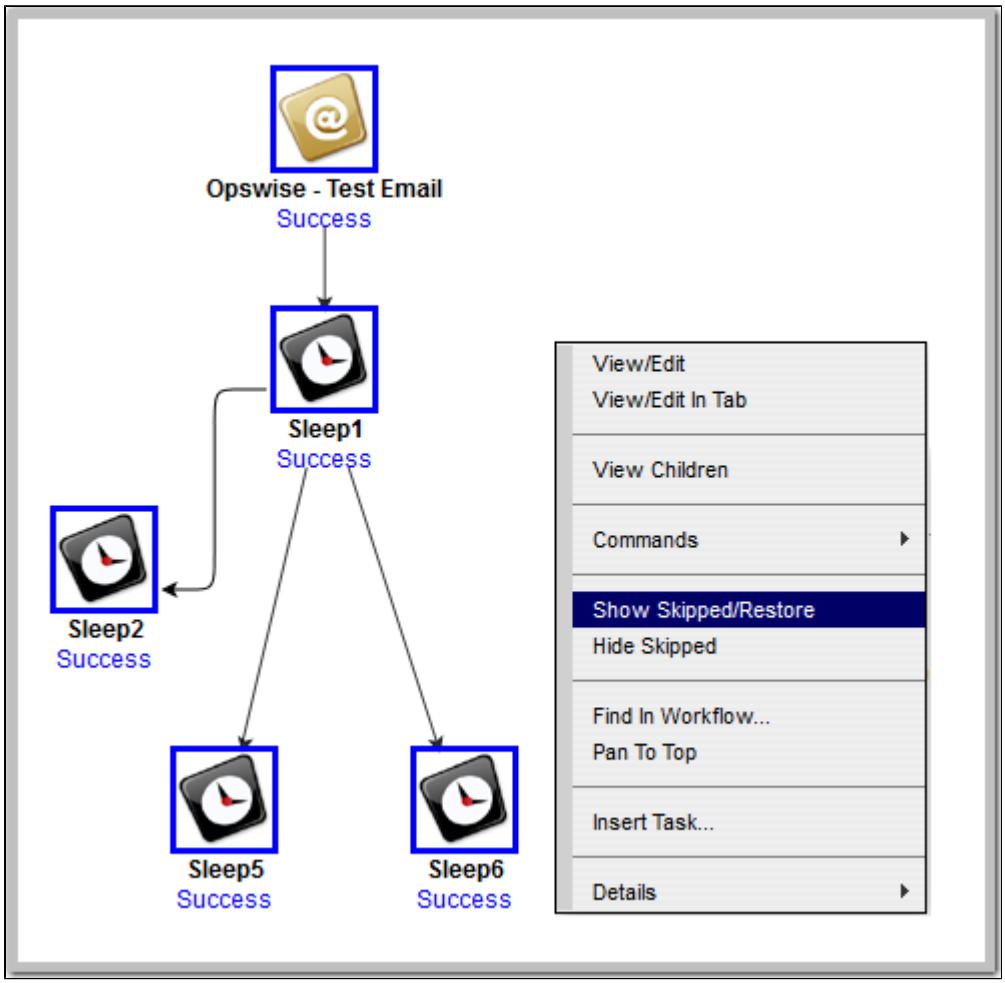


The task status changes from Skipped to Waiting. When Sleep1 completes successfully, the Controller will not skip Sleep3.

Step 7	To hide the skipped Sleep3 task in the Workflow Editor, right-click in the Workflow Monitor canvas and click Hide Skipped.



Step 8 To show the skipped Sleep3 task in the Workflow Monitor, right-click in the Workflow Editor canvas and click Show Skipped/Restore.



For additional information, see:

- Skipping a Task
- Unskipping a Task
- Showing or Hiding Skipped Tasks
- Adding Skip/Run Criteria for Specific Tasks

Tutorial - Using Variables in a Simple Task

- Introduction
- Resolving Variable Using Value from Global Variable Table
- Resolving Variable Using Value from Task
- Resolving Variable Using Value from Trigger

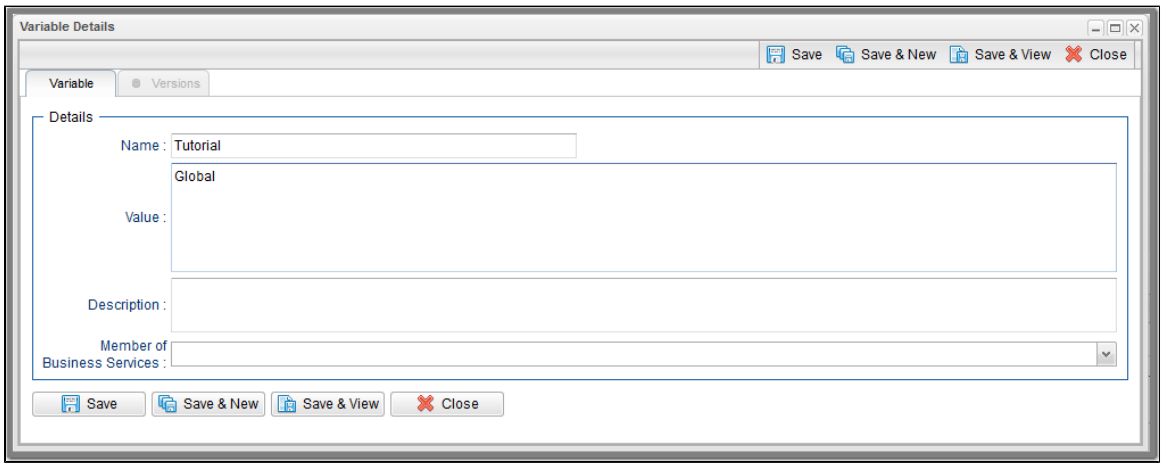
Introduction

Note
You need an Email Connection to perform this exercise.

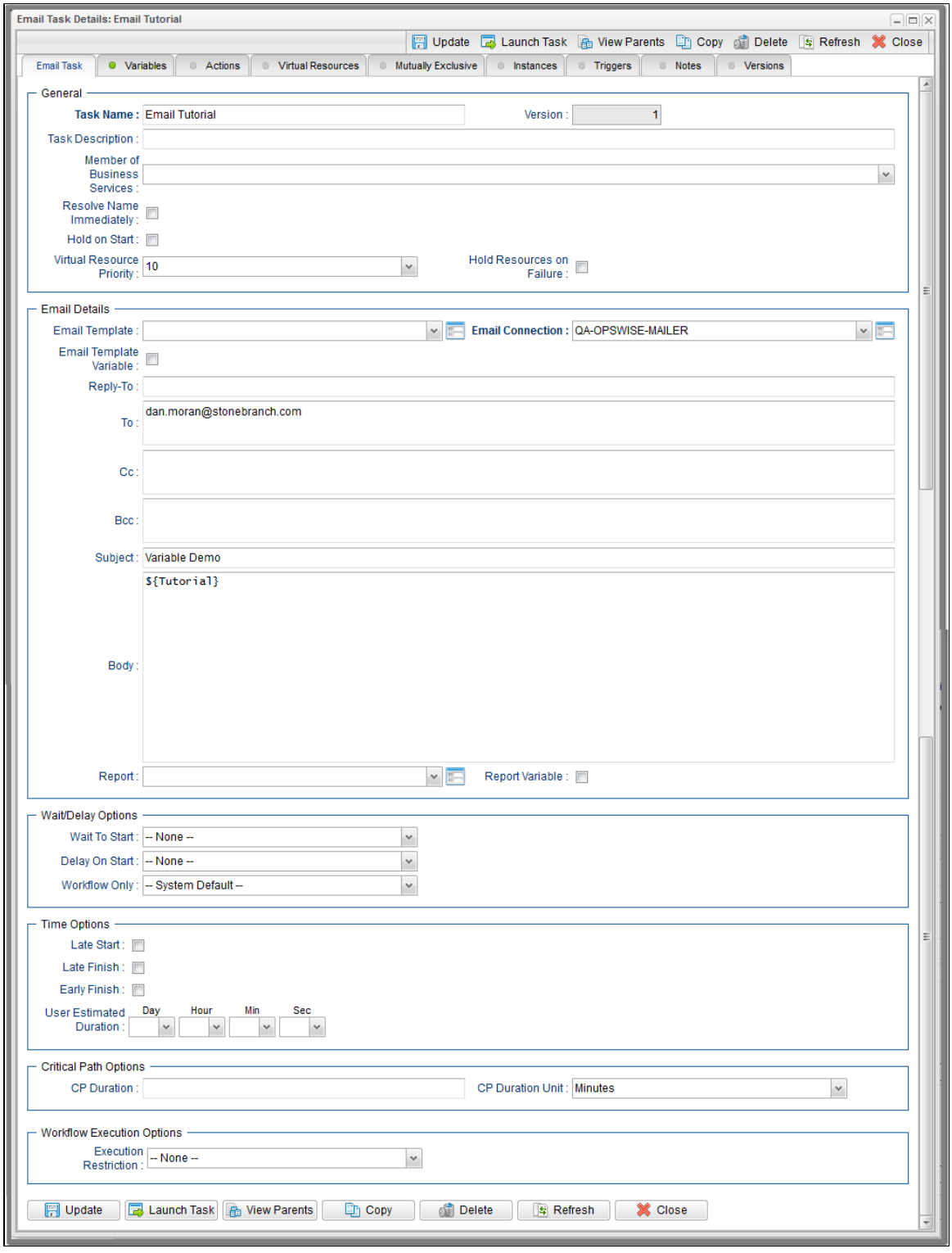
In the Launching an Email Task Based on a File Monitor tutorial, a built-in variable called `${ops_trigger_name}` and a system variable called `${_date}` were included to pass information into an Email message. Those variables were resolved using system information when the email task instance was created.

In this exercise, we will create a new user-defined variable, use it in a task, and run the task both manually and via a trigger to illustrate how such variables are resolved.

Resolving Variable Using Value from Global Variable Table

Step 1	Create a Variable with the following values: • **Name** = Tutorial • **Value** = Global
Step 2	Click the **Save** button. 
Step 3	Create an Email task with the following values: • Task Name = Email Tutorial • Email Connection = your Email connection • To = your Email address • Subject = Variable demo • Body=${Tutorial}

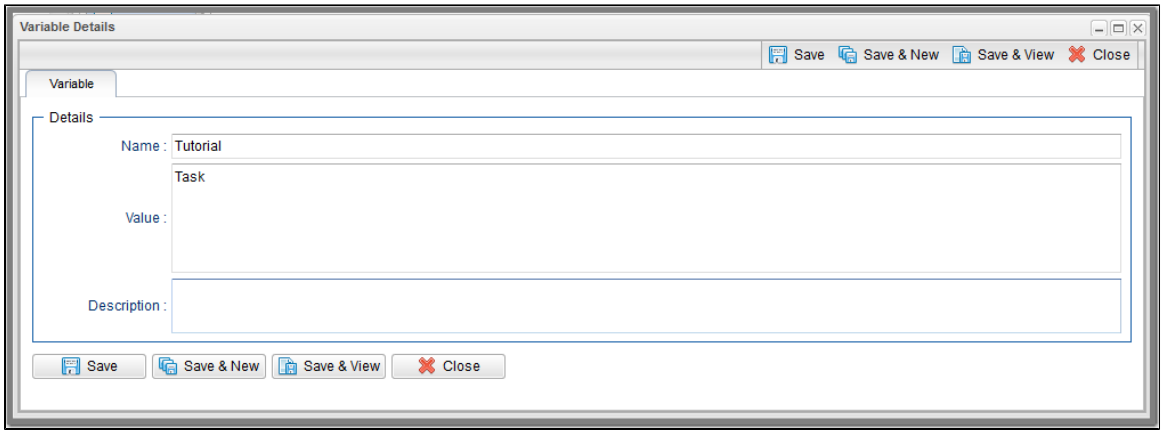
Step 4	Click the **Save** button, re-open the task, and click the **Launch Task** button.



Step 5	You should receive an email with **Global** in the body of the email.

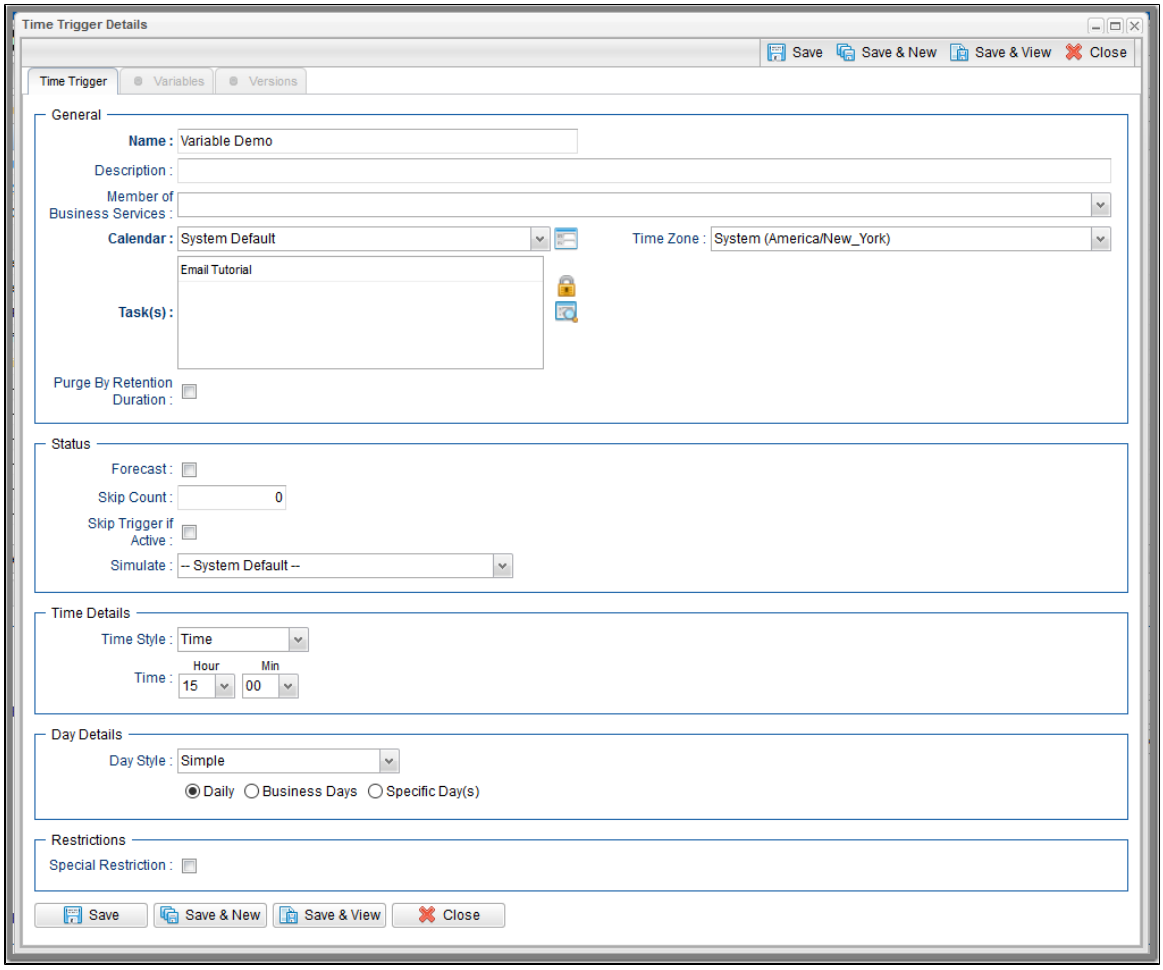
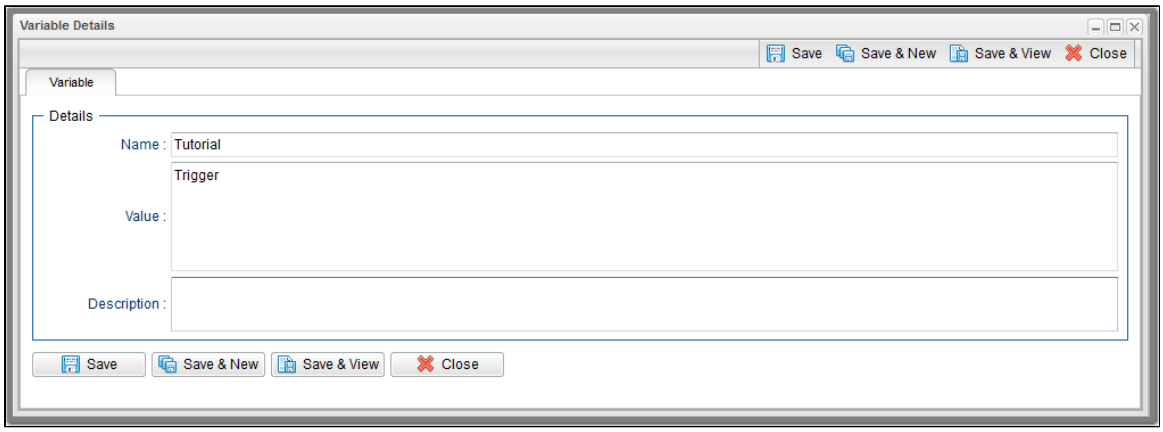
Resolving Variable Using Value from Task

Step 1	Open the Email Tutorial task and click the **Variables** tab.

Step 2	Click the **New** button to display Variable Details for a new Variable and enter the following values: • Name = Tutorial • Value = Task
Step 3	Click the **Save** button. 
Step 4	In the **Email Tutorial** Details, click the **Update** button and then click the **Launch Task** button.
Step 5	You should receive an email with **Task** in the body of the email.

Resolving Variable Using Value from Trigger

Step 1	Create a Time trigger with the following values: • Trigger Name = Variable Demo • Tasks = Email Tutorial • Time = (Five minutes from the present time.)

Step 2	Click the **Save** button. 
Step 3	Re-open the **Variable Demo** trigger and click the **Variables** tab.
Step 4	Click the **New** button to display Variable Details for a new Variable and enter the following values: • **Name** = Tutorial • **Value** = Trigger
Step 5	Click the **Save** button. 
Step 6	In the trigger Details, click the **Update** button and then the **Enable** button.
Step 7	You should receive an email with **Trigger** in the body of the email.

For additional information, see:

- User-Defined Variables

Tutorial - Using Variables in a Workflow

> **Note**
> You need a working Database Connection for this tutorial.

For tasks executing within a Workflow, the order of precedence for resolving user-defined variables differs.

As the following procedure demonstrates, the variable definition in the task takes precedence, then Universal Controller looks within the Workflow or parent Workflow(s), with the global variable coming last.

Step 1	Create a SQL task called **SQL with Variable** with the following SQL command: `CREATE TABLE ${tutorial}${_date("yyyyMMdd",5) } (name varchar(128), value varchar(128));`
Step 2	Click the **Save** button and then re-open the task.
Step 3	Click the **Variables** tab and create a Variable with the following values: • **Name** = tutorial • *Value+ = task
Step 4	Click the **Save** button and in the task Details, click the **Update** button.
Step 4	Create a Workflow called **Variable Workflow**.
Step 5	Add the **SQL With Variable** task to the Workflow and save it.
Step 6	Launch **Variable Workflow**. and open the SQL With Variable task instance on the Activity Monitor. Note that the SQL command resembles the following, with the value from the task variable. `CREATE TABLE task20090913 (name varchar(128), value varchar(128));`
Step 7	Open the task and delete the task variable.
Step 8	Go back to Variable Workflow and add the following variable: `tutorial/workflow`
Step 9	Open the task instance. The SQL command used the variable from the workflow because the task no longer had a variable. `CREATE TABLE workflow20090913 (name varchar(128), value varchar(128));`

For additional information, see:

- User-Defined Variables

Tutorial - Creating Custom Days and Periods

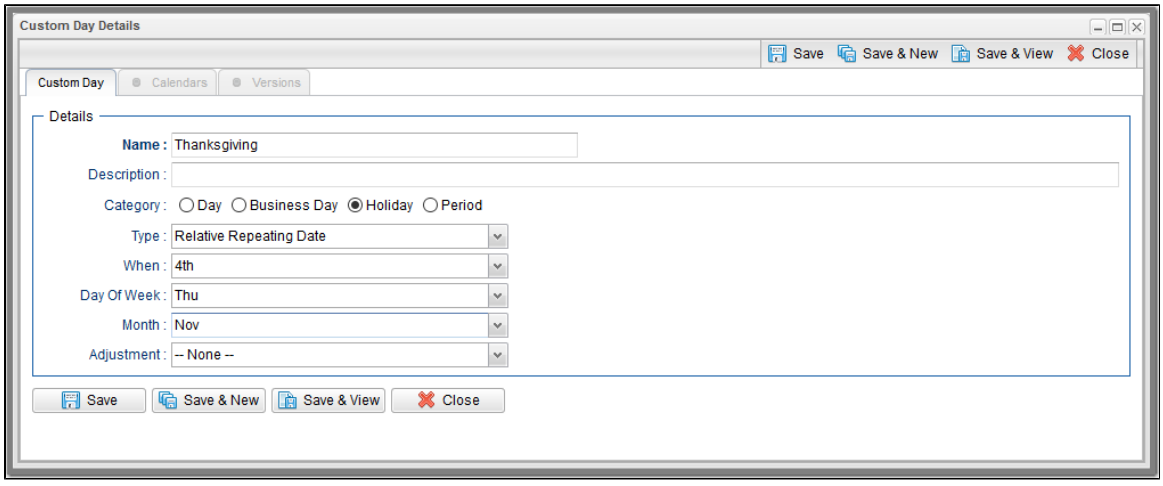
- Introduction
- Create a Custom Day
- Create a Custom Period
- Assigning Custom Day and Custom Period to a Calendar
- Creating a Local Custom Day for a Calendar
- Selecting a Custom Calendar for a Trigger

Introduction

In this tutorial, we will create a Custom Day and period for a Calendar, and assign that Calendar to a Trigger.

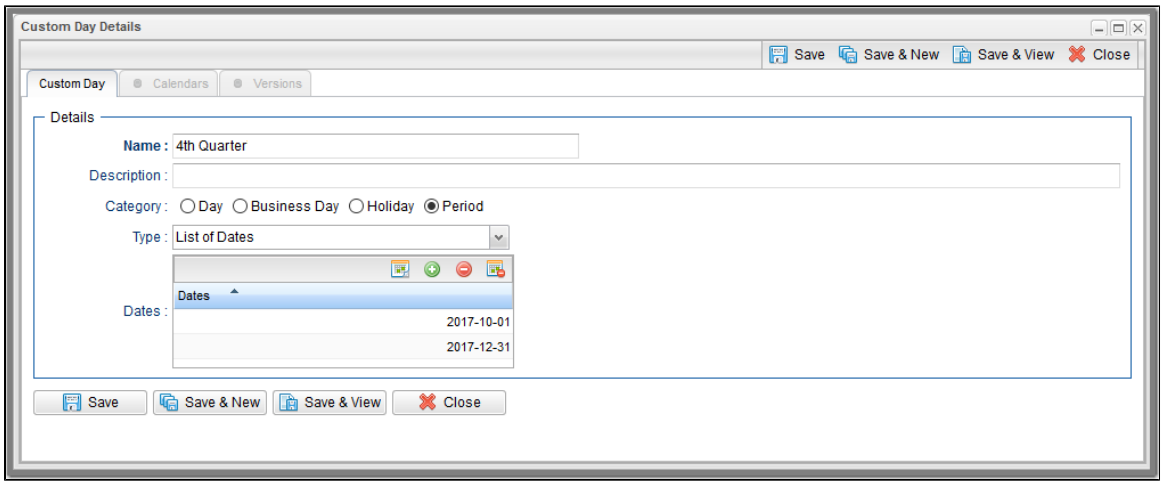
Create a Custom Day

In this procedure, we will create a Custom Day, which can be applied to any Calendar.

Step 1	From the Automation Center navigation pane, select **Other > Custom Days**. The Custom Days list displays.
Step 2	Click the **New** button to display Custom Day Details for a new Custom Day and enter/select the following values: • **Name** = Thanksgiving • **Holiday** is enabled • **Type** = Relative Repeating Date • **When** = 4th • **Day of Week** = Thu • **Month** = Nov
Step 3	Click the **Save** button. 

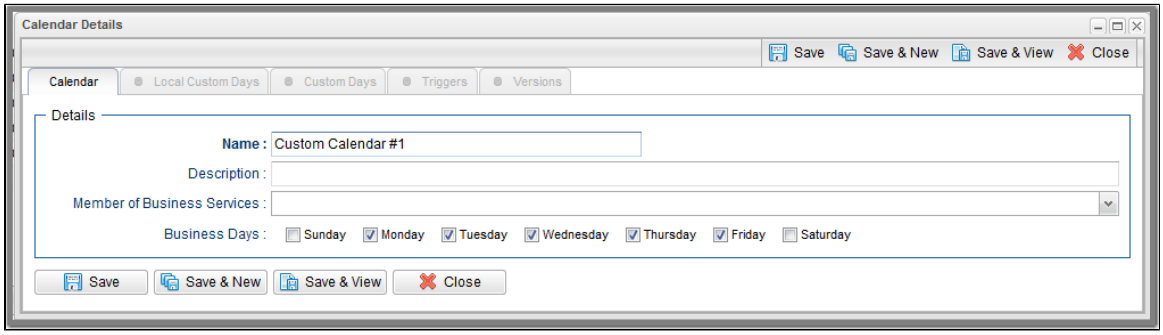
Create a Custom Period

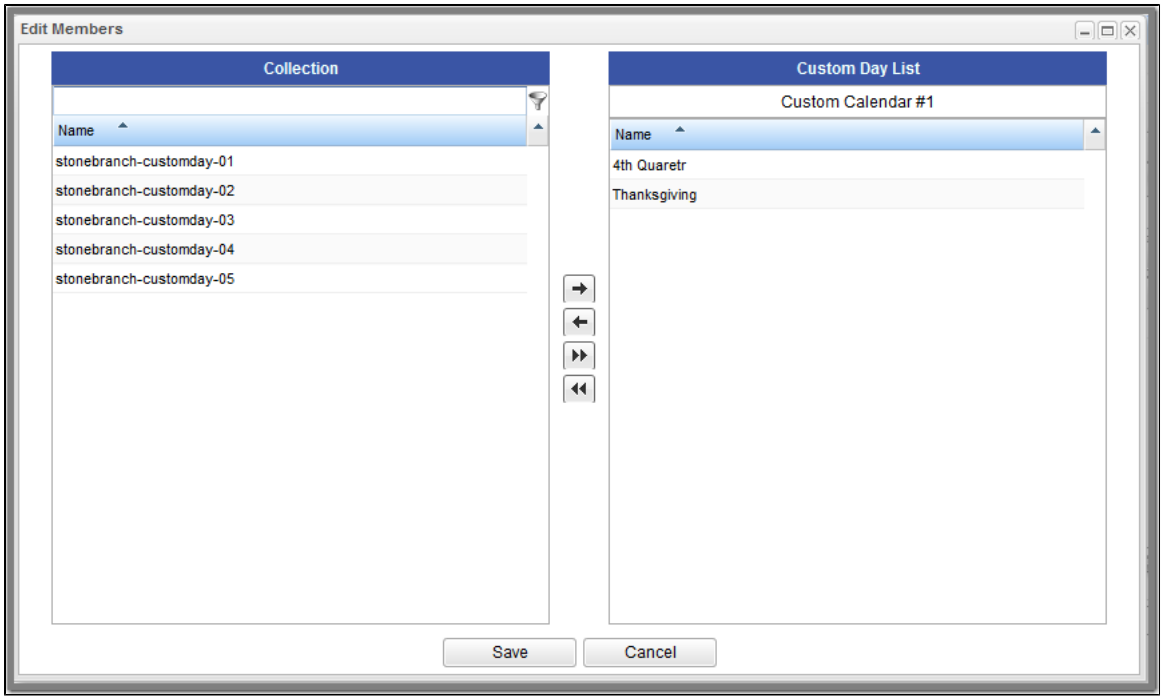
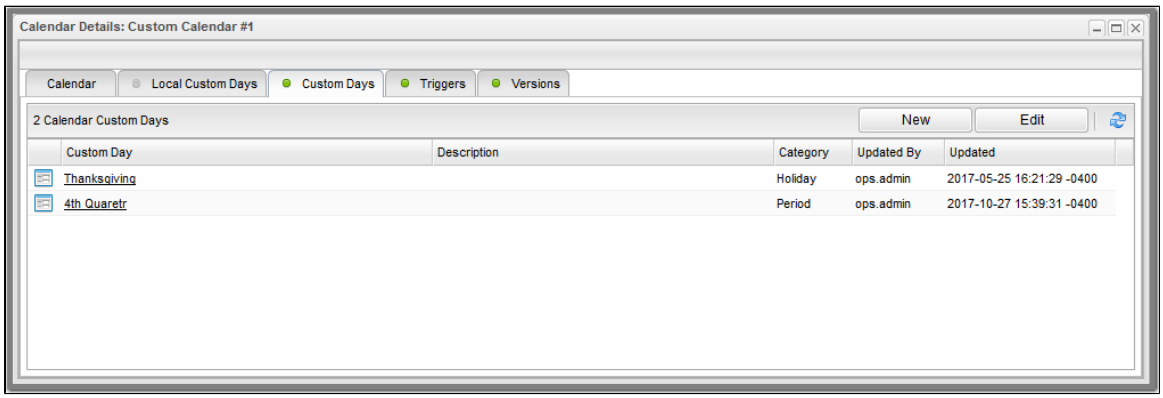
In this procedure, we will create a custom period of days, which can be applied to any calendar.

Step 1	Click the **New** button on the Custom Days list and to display Custom Day Details for a new Custom Day and enter/select the following values: • **Name** = 4th Quarter • **Period** is enabled. • **Type** = List of Dates • **Dates** = 2017-10-01 and 2017-12-31.
Step 2	Click the **Save** button. 

Assigning Custom Day and Custom Period to a Calendar

In this procedure, we will assign the custom day and custom period to a new Calendar.

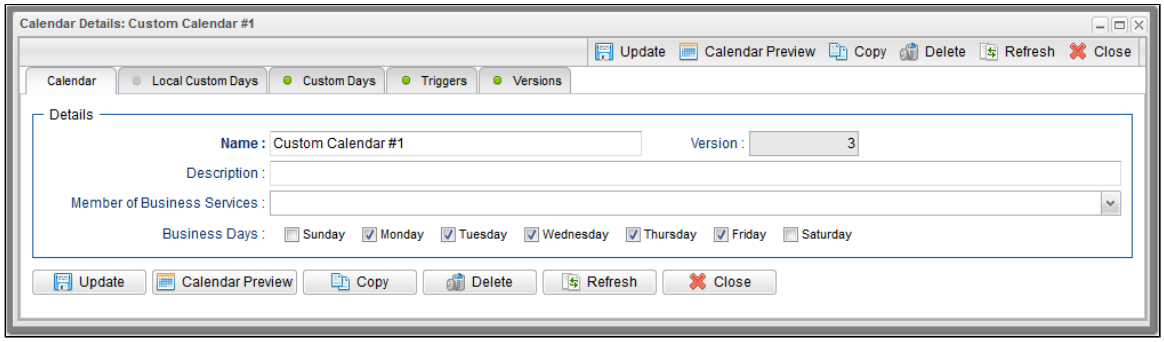
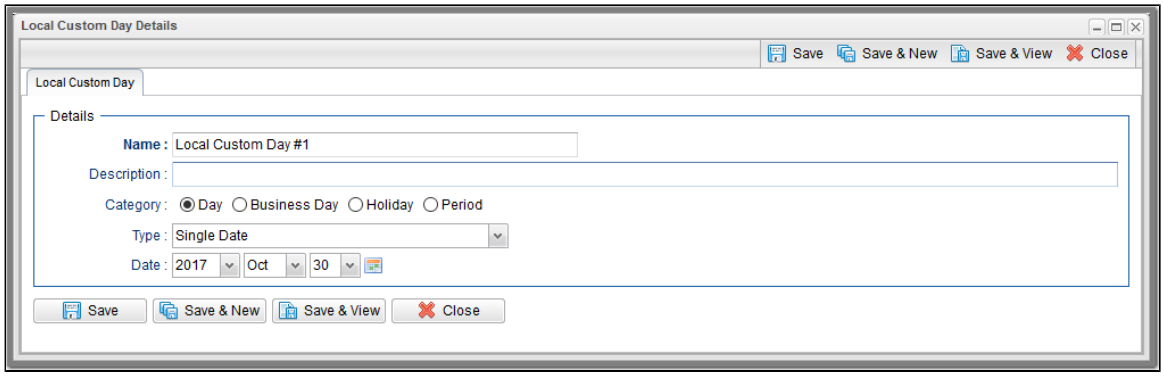
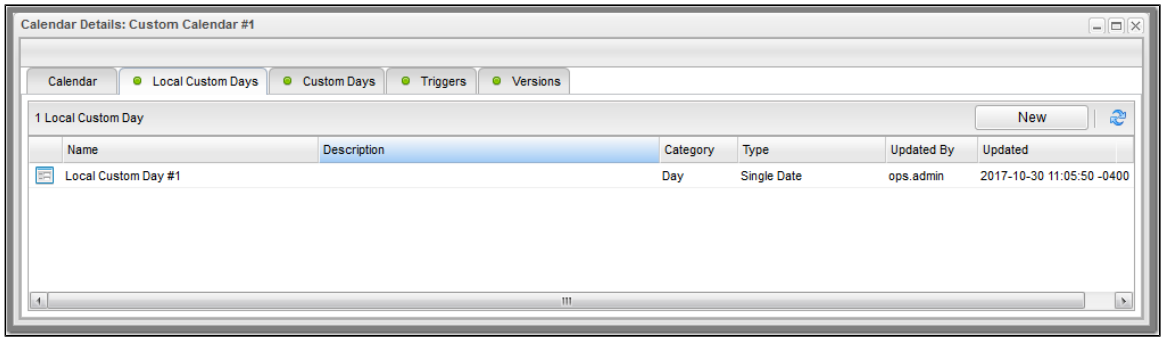
Step 1	From the **Automation Center** navigation pane, select **Other > Calendars**. The Calendars list displays.
Step 2	Click the **New** button to display Calendar Details for a new Calendar and enter/select the following values: • **Name** = Custom Calendar #1
Step 3	Click the **Save** button. 
Step 4	Re-open **Custom Calendar #1** and click the **Custom Days** tab. The Custom Days list for this Calendar displays.

Step 5	Click the **Edit** button to display an **Edit Members** pop-up dialog of all Custom Days. 
Step 6	Select **Thanksgiving** and **4th Quarter**, and then click the **Save** button. Those two Custom Days now appear on the Custom Days list for **Custom Calendar #1**. 
Step 7	In the Calendar Details, click the **Update** button.

Note
You also can assign a Custom Day to a Calendar by clicking the Calendars tab in the Custom Day Details.

Creating a Local Custom Day for a Calendar

In this procedure, we will create a local custom day for an existing Calendar.

Step 1	Open **Custom Calendar #1** that you created in the previous procedure. 
Step 2	Click the **Local Custom Days** tab to display an empty Local Custom Days list.
Step 3	Click the **New** button to display Local Custom Details for a new Local Custom Day, and enter the following values: • **Name** = Local Custom Day #1
Step 4	Click the **Save** button.  **Local Custom Days #1** now appears on the Local Custom Days list. 

Selecting a Custom Calendar for a Trigger

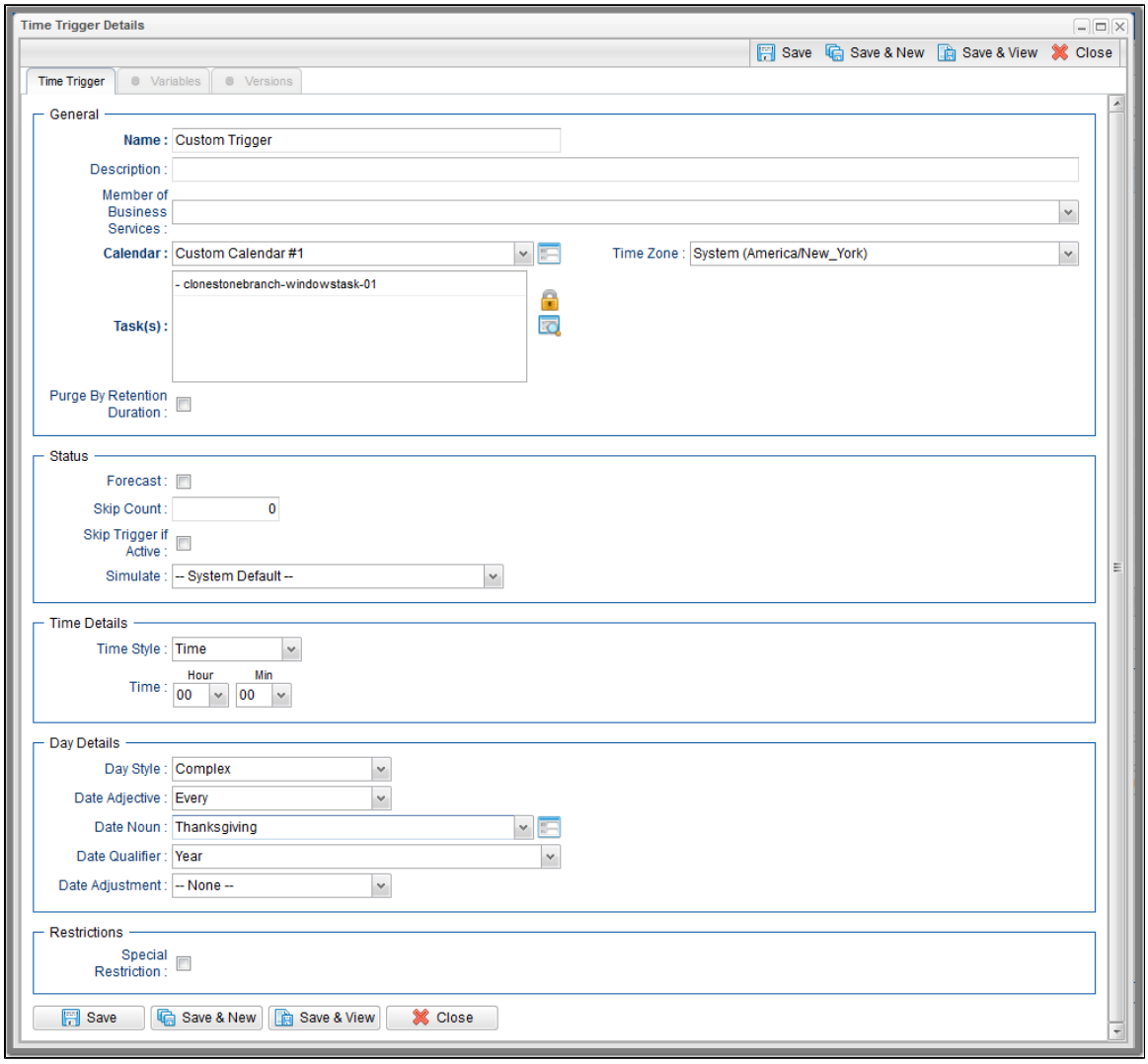
In this procedure, we will assign the custom calendar to a trigger.

Step 1	From the **Automation Center** navigation pane, click **Triggers > Time Triggers**. The Time Triggers list displays.

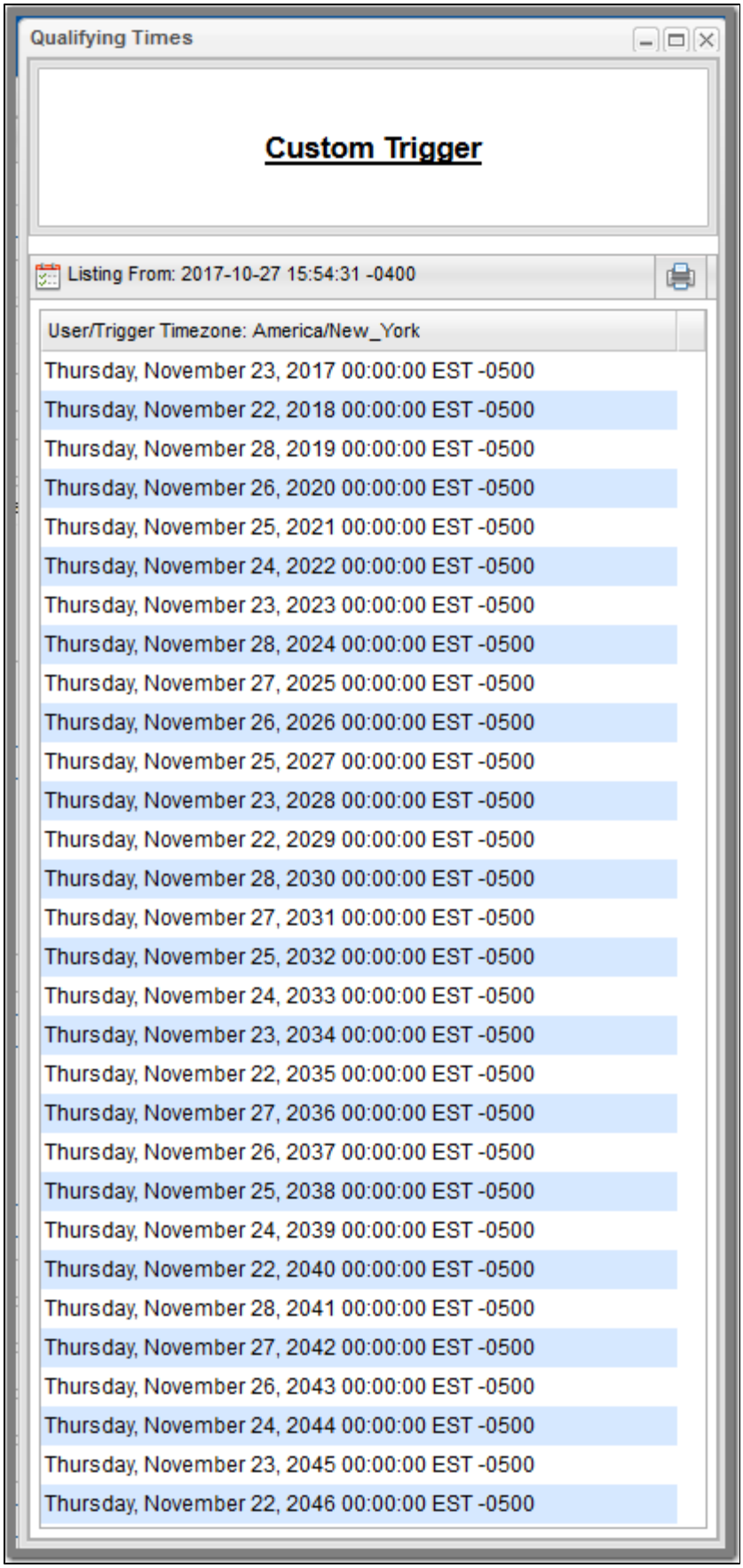
Step 2 Click the **New** button to display Time Trigger Details for a new Time Trigger and enter/select the following values:

- **Name** = Custom Trigger
- **Calendar** = Custom Calendar #1
- **Task(s)** = (any task)
- **Day Style** = Complex
- **Date Noun** = Thanksgiving (a Custom Day created for Custom Calendar #1)

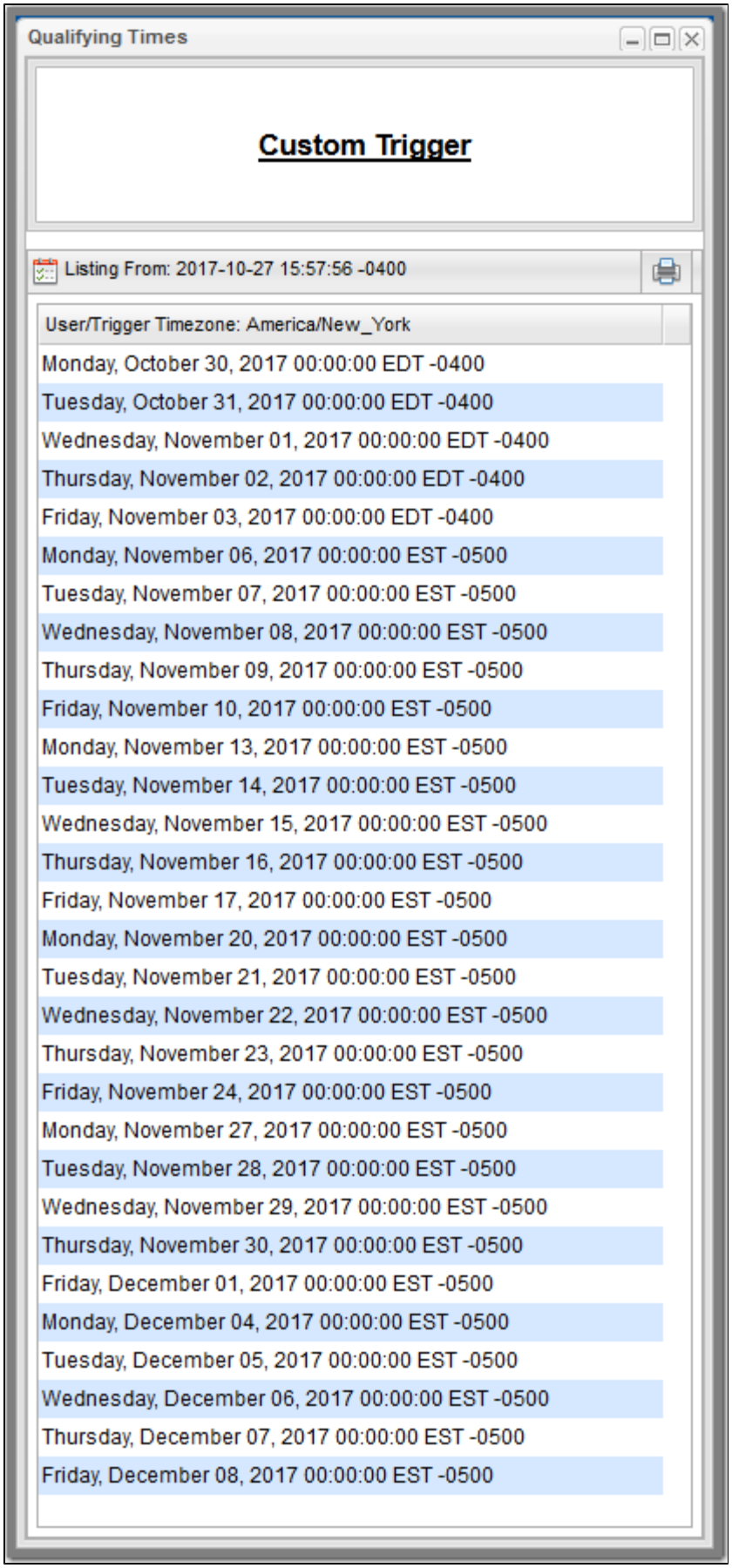
Step 3 Click the **Save** button.



Step 4 Re-open **Custom Trigger** and click the **List Qualifying Times** button to see that the trigger will run the task every year on Thanksgiving.


Qualifying Times
Custom Trigger
Listing From: 2017-10-27 15:54:31 -0400
User/Trigger Timezone: America/New_York
Thursday, November 23, 2017 00:00:00 EST -0500
Thursday, November 22, 2018 00:00:00 EST -0500
Thursday, November 28, 2019 00:00:00 EST -0500
Thursday, November 26, 2020 00:00:00 EST -0500
Thursday, November 25, 2021 00:00:00 EST -0500
Thursday, November 24, 2022 00:00:00 EST -0500
Thursday, November 23, 2023 00:00:00 EST -0500
Thursday, November 28, 2024 00:00:00 EST -0500
Thursday, November 27, 2025 00:00:00 EST -0500
Thursday, November 26, 2026 00:00:00 EST -0500
Thursday, November 25, 2027 00:00:00 EST -0500
Thursday, November 23, 2028 00:00:00 EST -0500
Thursday, November 22, 2029 00:00:00 EST -0500
Thursday, November 28, 2030 00:00:00 EST -0500
Thursday, November 27, 2031 00:00:00 EST -0500
Thursday, November 25, 2032 00:00:00 EST -0500
Thursday, November 24, 2033 00:00:00 EST -0500
Thursday, November 23, 2034 00:00:00 EST -0500
Thursday, November 22, 2035 00:00:00 EST -0500
Thursday, November 27, 2036 00:00:00 EST -0500
Thursday, November 26, 2037 00:00:00 EST -0500
Thursday, November 25, 2038 00:00:00 EST -0500
Thursday, November 24, 2039 00:00:00 EST -0500
Thursday, November 22, 2040 00:00:00 EST -0500
Thursday, November 28, 2041 00:00:00 EST -0500
Thursday, November 27, 2042 00:00:00 EST -0500
Thursday, November 26, 2043 00:00:00 EST -0500
Thursday, November 24, 2044 00:00:00 EST -0500
Thursday, November 23, 2045 00:00:00 EST -0500
Thursday, November 22, 2046 00:00:00 EST -0500

Step 5	In the **Custom Trigger** Details, change the following values: • **Date Noun** = Business Day. • **Date Qualifier** = 4th Quarter (a Custom Day period created for Custom Calendar #1)
Step 6	Click the **Update** button.
Step 7	Re-open **Custom Trigger** and click the **List Qualifying Times** button to see that the trigger will run the task every business day during the custom period, October 1 to December 31.


Qualifying Times
Custom Trigger
Listing From: 2017-10-27 15:57:56 -0400
User/Trigger Timezone: America/New_York
Monday, October 30, 2017 00:00:00 EDT -0400
Tuesday, October 31, 2017 00:00:00 EDT -0400
Wednesday, November 01, 2017 00:00:00 EDT -0400
Thursday, November 02, 2017 00:00:00 EDT -0400
Friday, November 03, 2017 00:00:00 EDT -0400
Monday, November 06, 2017 00:00:00 EST -0500
Tuesday, November 07, 2017 00:00:00 EST -0500
Wednesday, November 08, 2017 00:00:00 EST -0500
Thursday, November 09, 2017 00:00:00 EST -0500
Friday, November 10, 2017 00:00:00 EST -0500
Monday, November 13, 2017 00:00:00 EST -0500
Tuesday, November 14, 2017 00:00:00 EST -0500
Wednesday, November 15, 2017 00:00:00 EST -0500
Thursday, November 16, 2017 00:00:00 EST -0500
Friday, November 17, 2017 00:00:00 EST -0500
Monday, November 20, 2017 00:00:00 EST -0500
Tuesday, November 21, 2017 00:00:00 EST -0500
Wednesday, November 22, 2017 00:00:00 EST -0500
Thursday, November 23, 2017 00:00:00 EST -0500
Friday, November 24, 2017 00:00:00 EST -0500
Monday, November 27, 2017 00:00:00 EST -0500
Tuesday, November 28, 2017 00:00:00 EST -0500
Wednesday, November 29, 2017 00:00:00 EST -0500
Thursday, November 30, 2017 00:00:00 EST -0500
Friday, December 01, 2017 00:00:00 EST -0500
Monday, December 04, 2017 00:00:00 EST -0500
Tuesday, December 05, 2017 00:00:00 EST -0500
Wednesday, December 06, 2017 00:00:00 EST -0500
Thursday, December 07, 2017 00:00:00 EST -0500
Friday, December 08, 2017 00:00:00 EST -0500

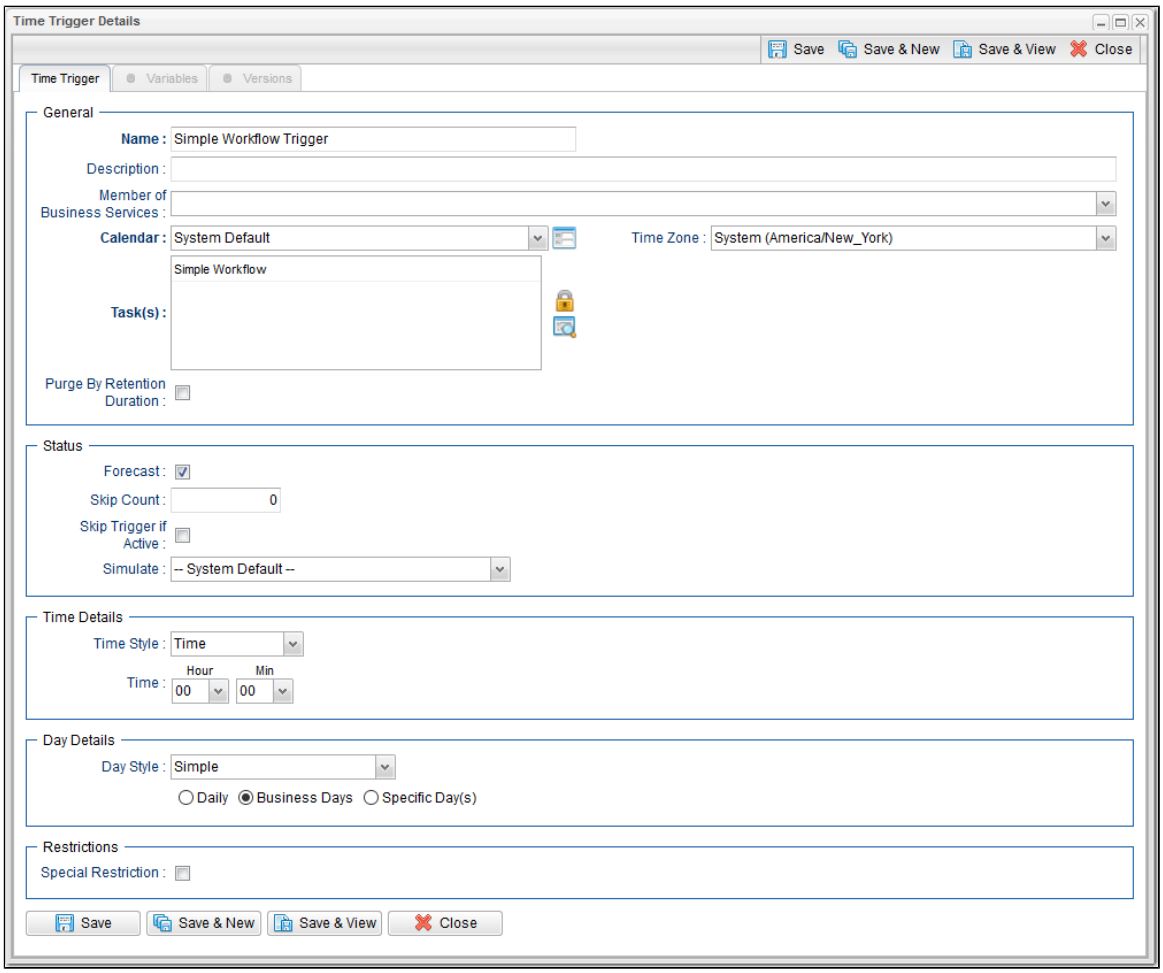
For additional information, see:

- Triggers
- Creating Calendars
- Creating Custom Days

Tutorial - Generating Forecast Data

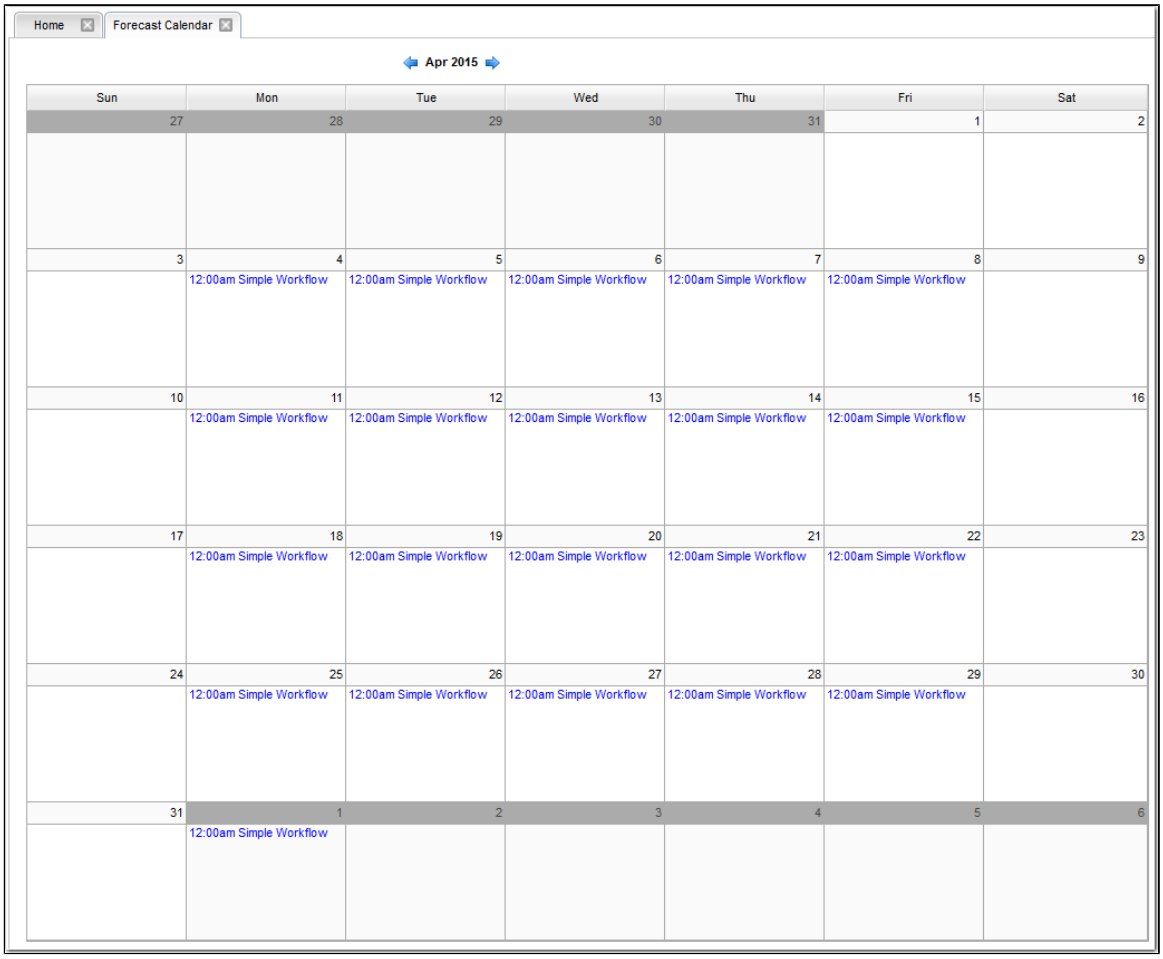
In this exercise, we will:

- Add a workflow and its tasks to the Forecast calendar.
- Run the workflow and display its forecast information.
- Update a task in the workflow and then re-run the workflow.
- Recalculate the forecast information for the workflow.

Step 1	Create a Time trigger and enter/select the following values: • **Name** = Simple Workflow Trigger. • **Task(s)** = Simple Workflow • **Forecast** = enabled • **Business Days** = enabled 
Step 2	Save the trigger and then enable it. A trigger must be enabled in order to generate forecast data for it.

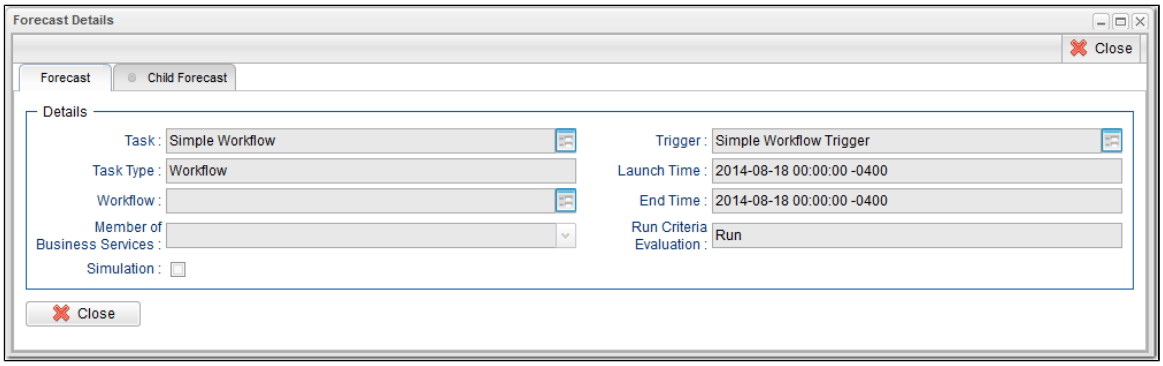
Step 3

From the Automation Center navigation pane, select **Triggers > Forecast Calendar**. The Forecast Calendar identifies Simple Workflow on the days in the forecast period, for the current month, when it will be launched by Simple Workflow Trigger. (By default, tasks within a Workflow are not displayed.)



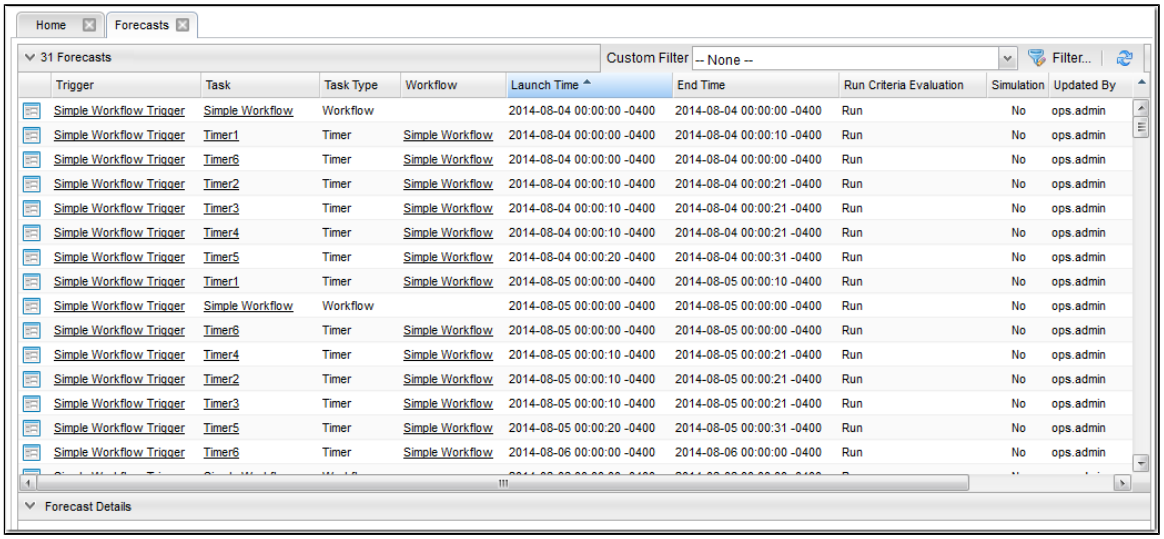
Step 4

Click any Simple Workflow link in the Forecast Calendar to display a Forecast Details pop-up. Note that the Launch Time and End Time are identical; since the workflow has never been run, there is no data to support estimated times.



Step 5

From the Automation Center navigation pane, select **Triggers > Forecasts**. The Forecasts list identifies Simple Workflow and every task in Simple Workflow, as well as their Launch Times and End Times, for every day in the forecast period when Simple Workflow will be launched by Simple Workflow Trigger.



Trigger	Task	Task Type	Workflow	Launch Time	End Time	Run Criteria Evaluation	Simulation	Updated By
Simple Workflow Trigger	Simple Workflow	Workflow		2014-08-04 00:00:00 -0400	2014-08-04 00:00:00 -0400	Run	No	ops.admin
Simple Workflow Trigger	Timer1	Timer	Simple Workflow	2014-08-04 00:00:00 -0400	2014-08-04 00:00:10 -0400	Run	No	ops.admin
Simple Workflow Trigger	Timer6	Timer	Simple Workflow	2014-08-04 00:00:00 -0400	2014-08-04 00:00:00 -0400	Run	No	ops.admin
Simple Workflow Trigger	Timer2	Timer	Simple Workflow	2014-08-04 00:00:10 -0400	2014-08-04 00:00:21 -0400	Run	No	ops.admin
Simple Workflow Trigger	Timer3	Timer	Simple Workflow	2014-08-04 00:00:10 -0400	2014-08-04 00:00:21 -0400	Run	No	ops.admin
Simple Workflow Trigger	Timer4	Timer	Simple Workflow	2014-08-04 00:00:10 -0400	2014-08-04 00:00:21 -0400	Run	No	ops.admin
Simple Workflow Trigger	Timer5	Timer	Simple Workflow	2014-08-04 00:00:20 -0400	2014-08-04 00:00:31 -0400	Run	No	ops.admin
Simple Workflow Trigger	Timer1	Timer	Simple Workflow	2014-08-05 00:00:00 -0400	2014-08-05 00:00:10 -0400	Run	No	ops.admin
Simple Workflow Trigger	Simple Workflow	Workflow		2014-08-05 00:00:00 -0400	2014-08-05 00:00:00 -0400	Run	No	ops.admin
Simple Workflow Trigger	Timer6	Timer	Simple Workflow	2014-08-05 00:00:00 -0400	2014-08-05 00:00:00 -0400	Run	No	ops.admin
Simple Workflow Trigger	Timer4	Timer	Simple Workflow	2014-08-05 00:00:10 -0400	2014-08-05 00:00:21 -0400	Run	No	ops.admin
Simple Workflow Trigger	Timer2	Timer	Simple Workflow	2014-08-05 00:00:10 -0400	2014-08-05 00:00:21 -0400	Run	No	ops.admin
Simple Workflow Trigger	Timer3	Timer	Simple Workflow	2014-08-05 00:00:10 -0400	2014-08-05 00:00:21 -0400	Run	No	ops.admin
Simple Workflow Trigger	Timer5	Timer	Simple Workflow	2014-08-05 00:00:20 -0400	2014-08-05 00:00:31 -0400	Run	No	ops.admin
Simple Workflow Trigger	Timer6	Timer	Simple Workflow	2014-08-06 00:00:00 -0400	2014-08-06 00:00:00 -0400	Run	No	ops.admin

Step 6

Re-display the Time Triggers list, right-click Simple Workflow Trigger, and then click Trigger Now to launch Simple Workflow.

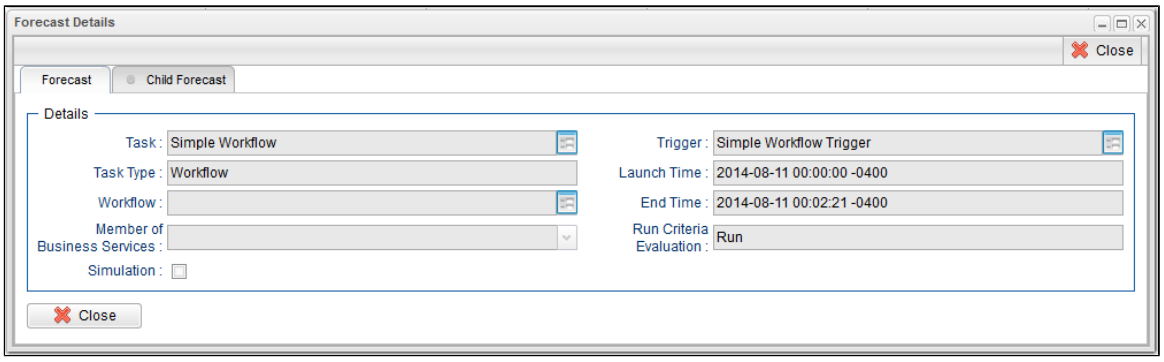
Step 7

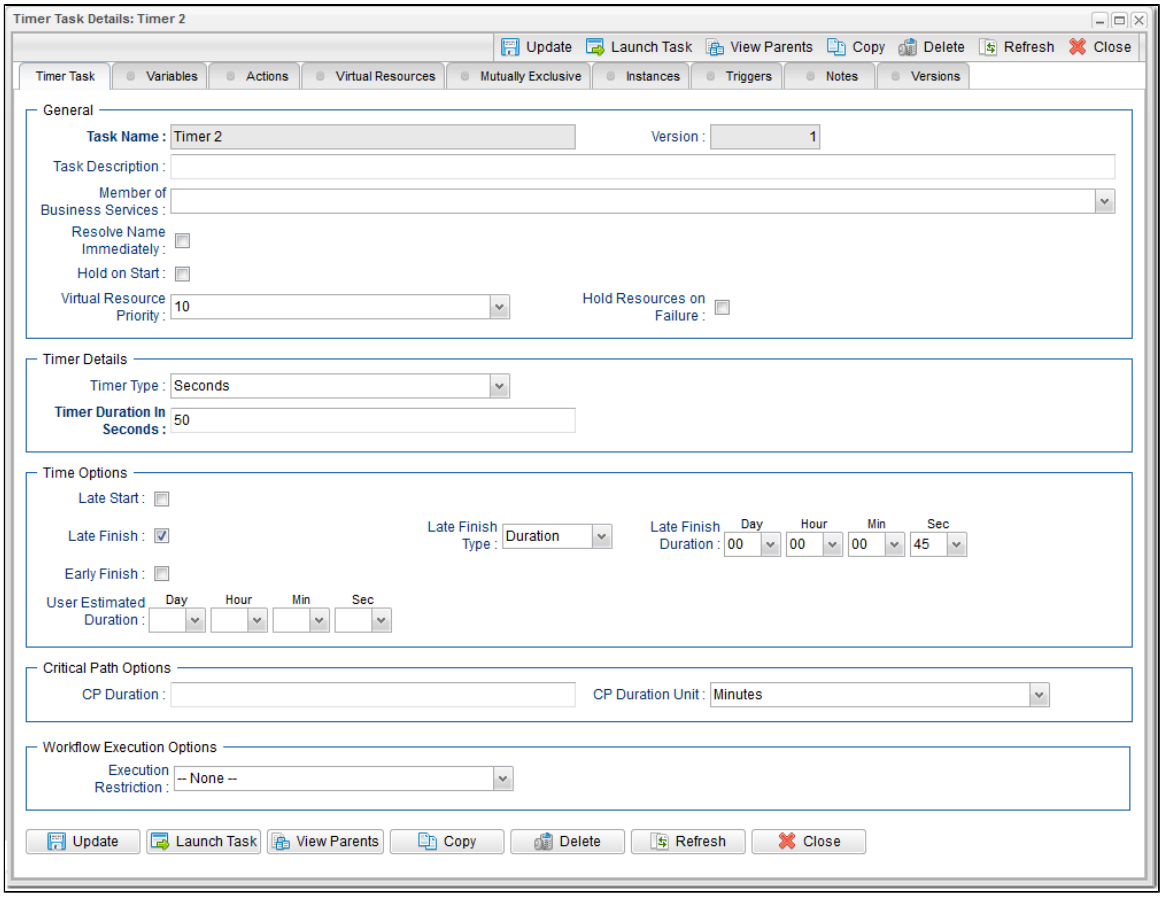
When Simple Workflow has completed, re-display the Time Triggers list, right-click Simple Workflow Trigger, and then click Recalculate Forecast.

> **Note**
> You also can recalculate the forecast by right-clicking Simple Workflow on the Workflow Tasks list and then clicking Recalculate Forecast.

Step 8

Re-display the Forecast Calendar and click any Simple Workflow link in the Forecast Calendar to display a Forecast Details pop-up, which now contains estimated Launch Time and End Time information based on the Workflow run just completed.



Step 9	Open the Timer Task and change the **Timer Duration in Seconds** value from 10 to 50, and then click Update. 
Step 10	Re-run Simple Workflow.
Step 11	When Simple Workflow has completed, re-calculate its Forecast; since the updated time for the Timer1 task affected the Simple Workflow End Time (the time it took to run the Workflow), the Forecast information for Simple Workflow is now obsolete.
Step 12	Re-open the Forecast Calendar and click any Simple Workflow link to see that 20 seconds has been added to the End Time.

For additional information, see:

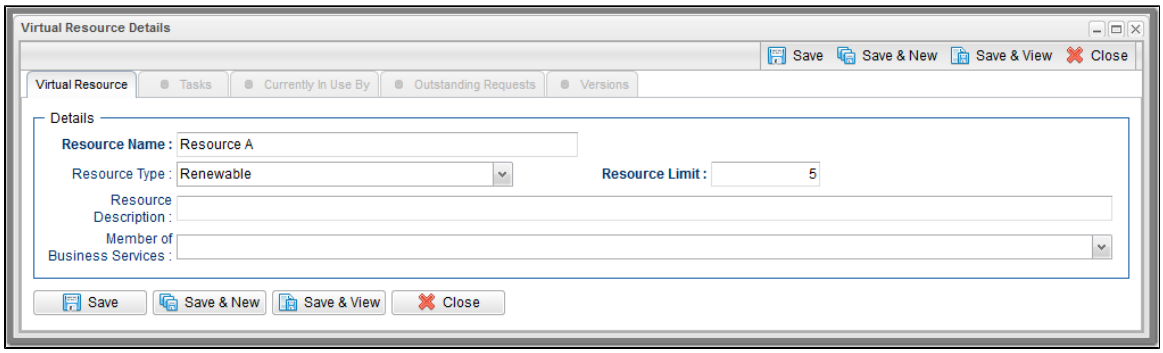
- Creating and Maintaining Workflows
- Triggers
- Displaying Trigger Forecast Information

Tutorial - Setting Up a Virtual Resource

In this exercise, we will set up an imaginary resource and three imaginary tasks.

In this scenario, two of our tasks are resource-intensive, and they run on the same machine. Therefore, if one is already running when the other is launched, we want the second task to wait until the first is finished before running. However, our third task is not so resource-intensive, so we will allow this one to run at the same time as either of the other two.

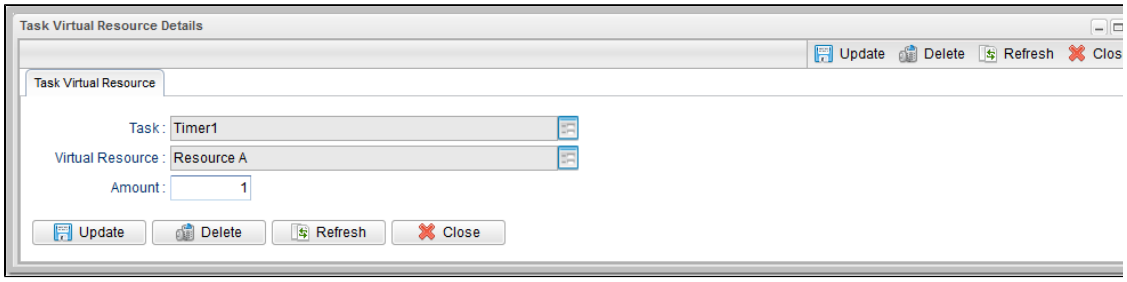
To simplify the exercise, we will use Timer tasks.

Step 1	From the Automation Center navigation pane, click **Other > Virtual Resources**. The Virtual Resources list displays.
Step 2	Click the **New** button to display an empty Virtual Resource Details and enter the following values: • **Resource Name** = Resource A • **Resource Limit** = 5
Step 3	Click the **Save** button. 

Step 4 Update the Timer1, Timer2, and Timer3 tasks that you created in the Creating a Simple Workflow tutorial.

For Timer1:

1. Change **Time in Seconds** to 60.
2. Click the Virtual Resources tab to display the Virtual Resources list for Timer1:
   a. Click the **Edit** button, add **Resource A** to the Virtual Resources list, and click **Save**.
   b. Click the Details icon for Resource A to display its Task Virtual Resource Details.



   c. Change the **Amount** (number of resource units required from the virtual resource) from 1 to 4 and click the **Update** but

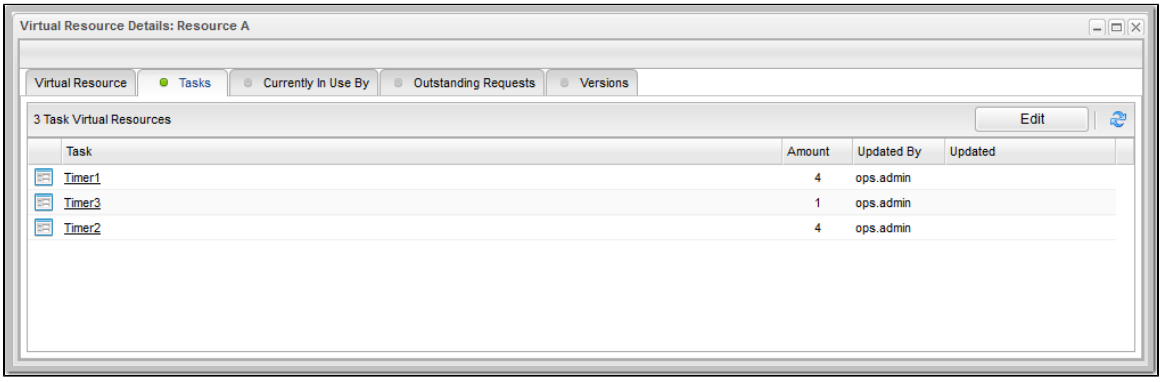
For Timer2:

1. Change **Time in Seconds** to 30.
2. Add Resource A and change the **Amount** to 4.

For Timer3:

1. Change **Time in Seconds** to 20.
2. Add Resource A and keep the **Amount** at 1.

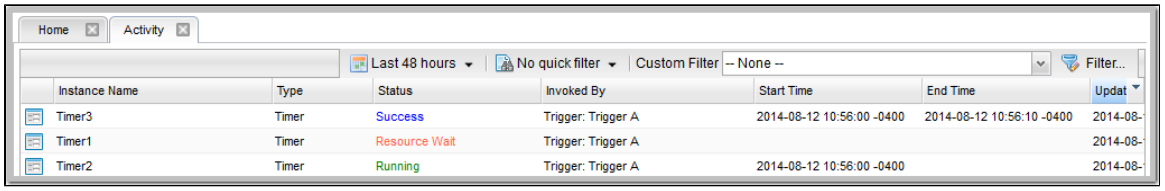
Step 5 Open Resource A and click the Tasks tab to see that Timer1, Timer2, and Timer3 are listed.



Step 6 Create a Time Trigger called **Trigger A**, and select Task1, Task2, and Task 3 in the **Task(s)** field, and enter a **Time** three minutes from the current time.

Step 7 Click the **Save** button, right-click Trigger A on the Triggers list, and click **Enable**.

Step 8 When the trigger is satisfied, only Timer1 or Timer 2 will run, along with Timer3. When Timer1 or Timer 2 finishes, the other will run run. Display the Activity Monitor and note that Timer1 or Timer 2 is waiting in **Resource Wait** status.



Step 9	In the Virtual Resource Details, click the **Currently In Use By** tab and observe which tasks are running on this virtual resource. Note that this display does not automatically refresh.

For additional information, see:

Virtual Resources

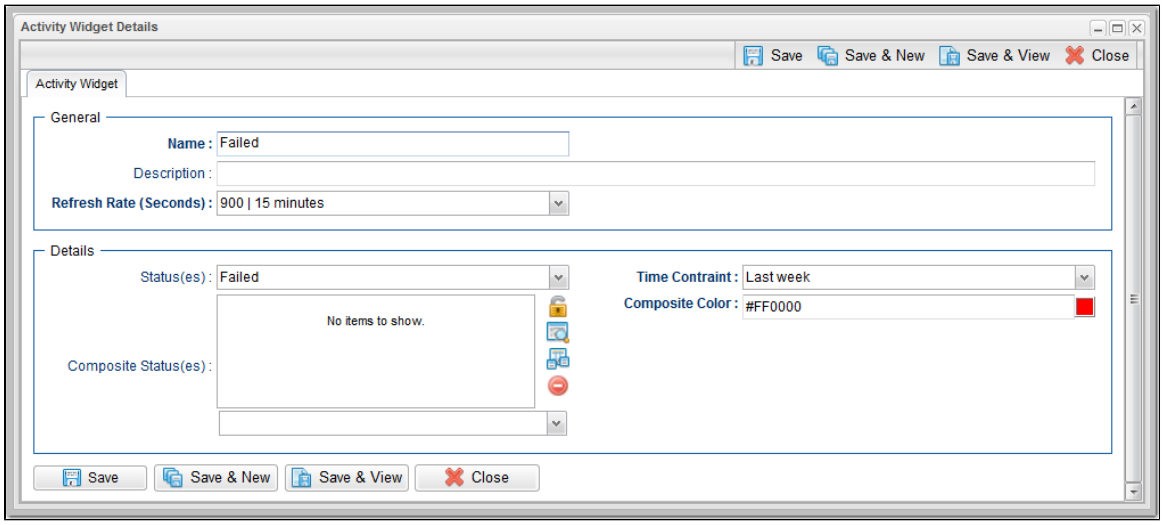
Tutorial - Creating a Widget

There are three types of Widgets, all of which can be placed on one or more Dashboards:

- System
- Activity
- Report

In this exercise, we will create an Activity Widget that identifies any failed task instances in the past week.

(You cannot create, modify, or delete a System Widget; they are provided by the Controller.)

Step 1	From the Reporting navigation pane, select **Widgets**. The Widgets list displays.
Step 2	Click the **New** button and then click **Activity Widget**. An Activity Widget Details pop-up displays.
Step 3	Enter / select the following values: • **Name** = Failed • **Refresh Rate (Seconds)** = 900 | 15 minutes • **Status(es)** = Failed • **Time Constraint** = Last week • **Composite Color** = Red 
Step 4	Click the **Save** button. This Widget can now be added to any Dashboard and can be selected for any Composite Widget.

For additional information, see:

- Widgets

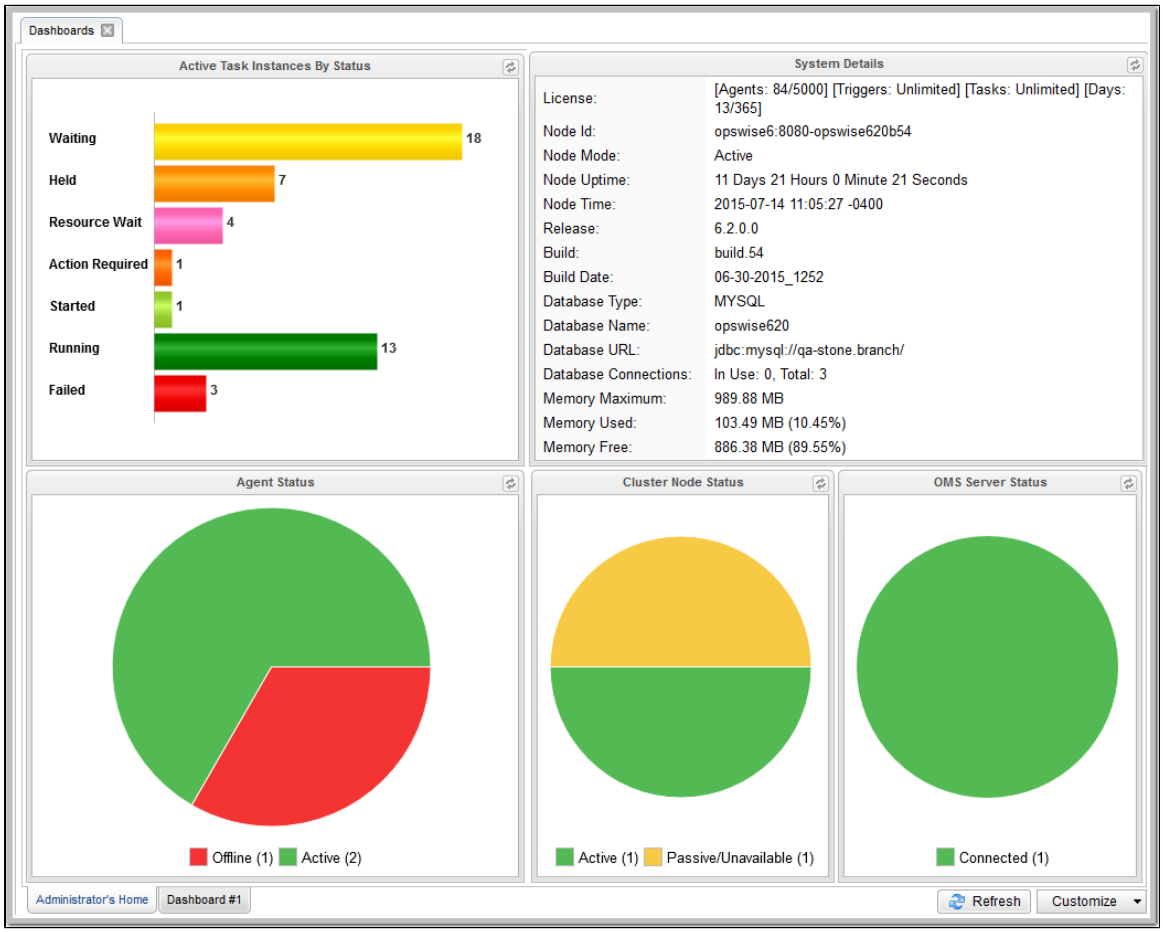
Tutorial - Creating a Dashboard and Adding Widgets

Creating a Dashboard and Adding Widgets

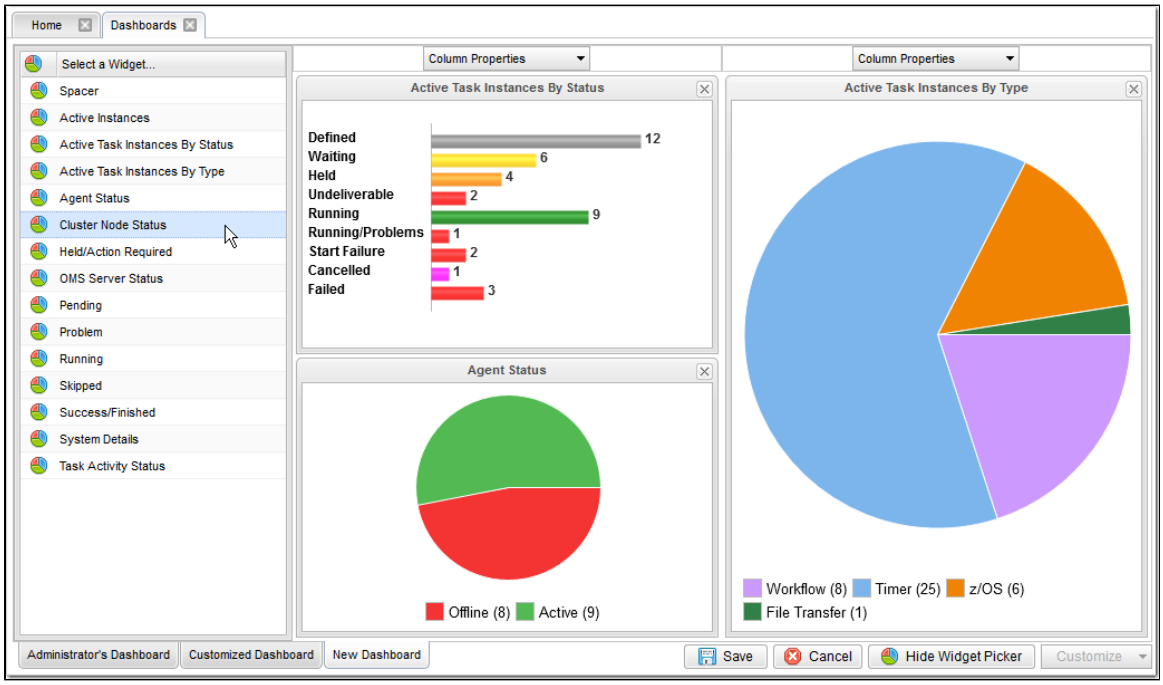
Universal Controller provides a default Dashboard containing multiple Widgets that displays as the Home Dashboard when you log in to Universal Controller.

You can create your own Dashboards containing any available Widgets.

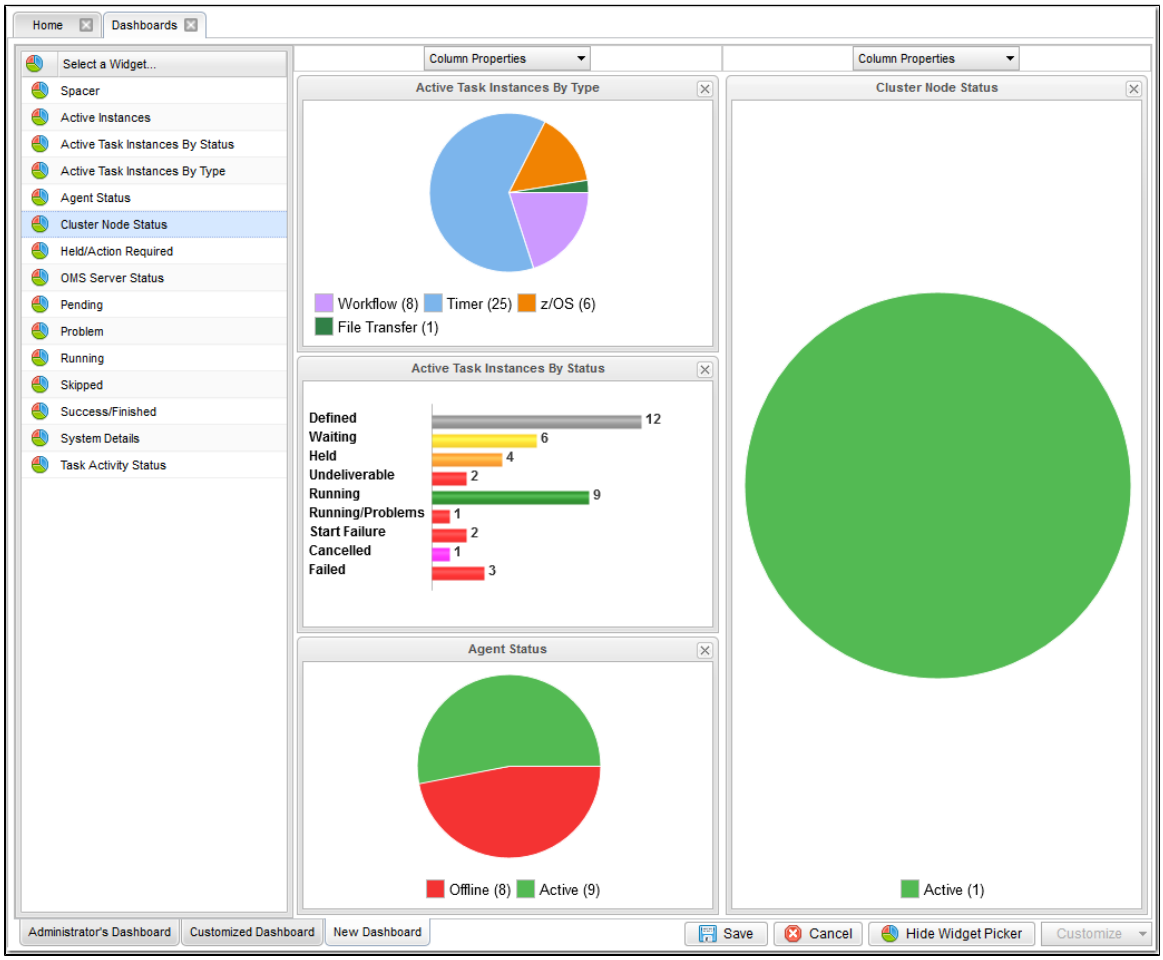
In this exercise, we are going to create a Dashboard and select Widgets for the Dashboard.

Step 1	Click the Home icon in the top right corner of any page to display the Home dashboard. 
Step 2	On the **Customize** drop-down list at the bottom of the Home Page, click **New**.
Step 3	Select a name for the new Dashboard on the New Dashboard pop-up and click **OK**. A new, empty Dashboard displays.

Step 4 From the list of Widgets to the left of the Dashboard, drag and drop any of the Widgets into either column. (If you want to add more columns, click either **Column Properties** drop-down list and then click **Add Column**.)



Step 5 After you have added Widgets to your new Dashboard, you can click the **x** icon in the top right corner of any Widget to remove it from the Dashboard, or drag and drop any Widget to a new location in the Dashboard.



Step 6	Click the **Save** button to add the Dashboard.

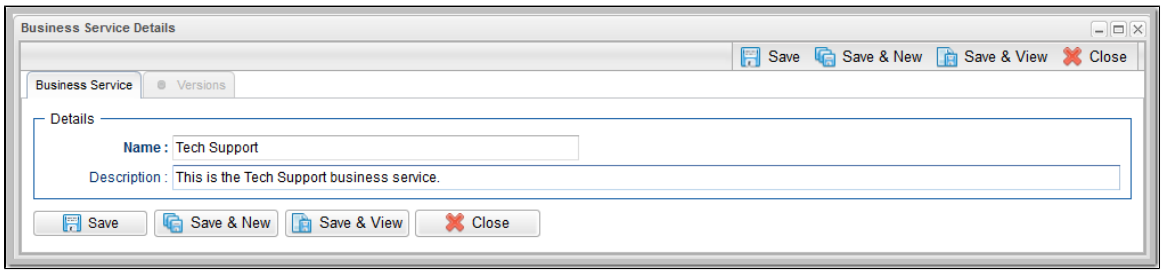
For additional information, see:

- Home Dashboard
- Dashboards
- Widgets

Tutorial - Creating Business Services

Business Services are used to group records into business functions.

In this exercise, we will create two hypothetical Business Services: Tech Support and Operations.

Step 1	From the Administration navigation pane, select **Security > Business Services**. The Business Services list displays.
Step 2	Click the **New** button to display an empty Business Service Details.
Step 3	Enter the following values: • **Name** = Tech Support • **Description** = This is the Tech Support business service.
Step 4	Click the **Save** button. 
Step 5	Repeat steps 2 to 4 for a Business Service called **Operations**.

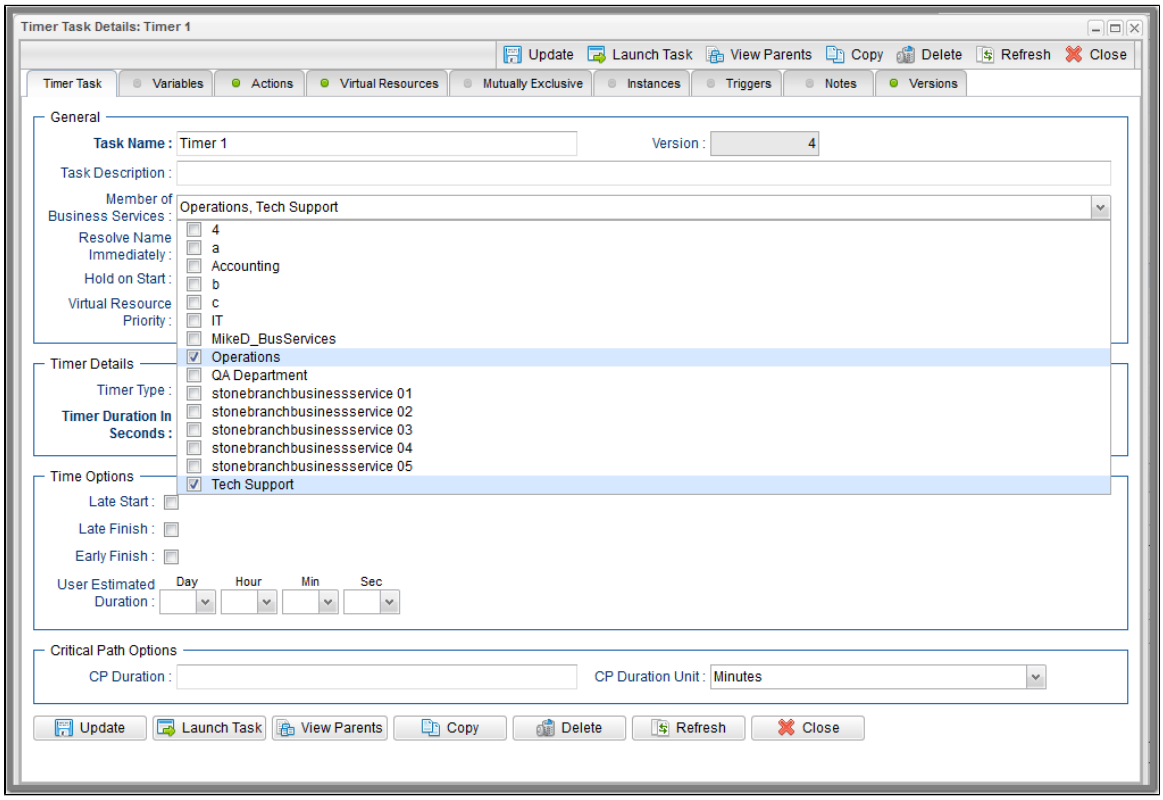
For additional information, see:

- Business Services

Tutorial - Assigning Records to Business Services

In this exercise, we will assign the Timer and Simple Workflow tasks created in the Creating a Simple Workflow tutorial to the Operations Business Service, and the SQL task and Bigger Workflow tasks created in the Running a Workflow with a Conditional Path tutorial) to the Tech Support Business Service.

See the Creating Business Services tutorial to see how these Business Services were created.

Step 1	Open the Timer task called **Timer1**, which you created in the Creating a Simple Workflow tutorial.
Step 2	From the **Member of Business Services** drop-down list, select Tech Support and Operations. 
Step 3	Click the **Update** button.
Step 4	Repeat steps 1 to 3 for the **Timer2** and **Timer3** tasks and the **Simple Workflow** Workflow, all of which you also created in the Creating a Simple Workflow tutorial.

For additional information, see:

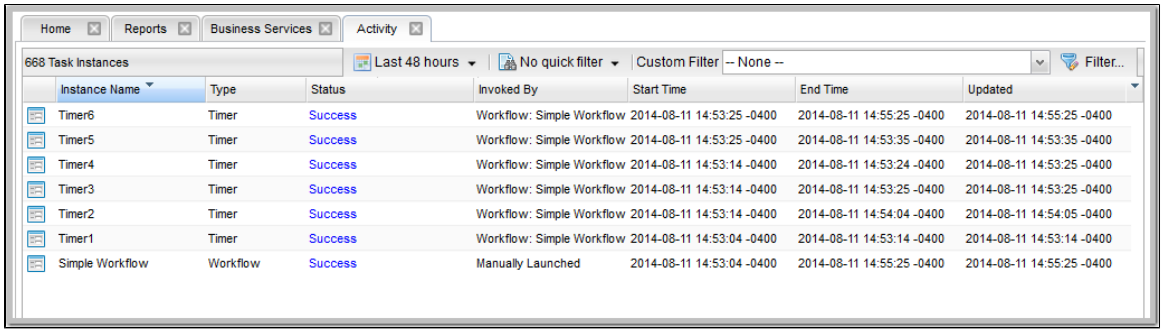
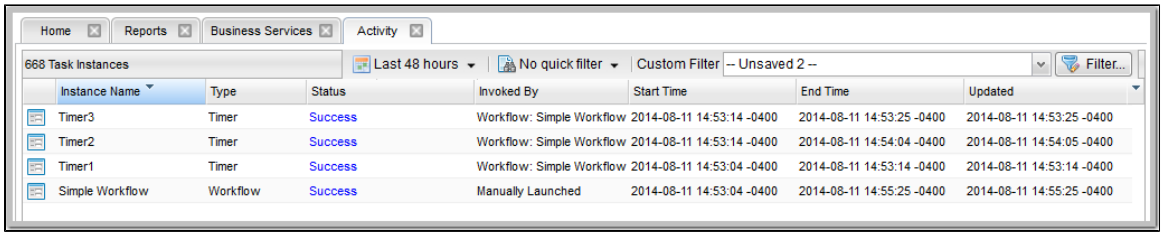
- Business Services

Tutorial - Viewing Activity by Business Service

- Introduction

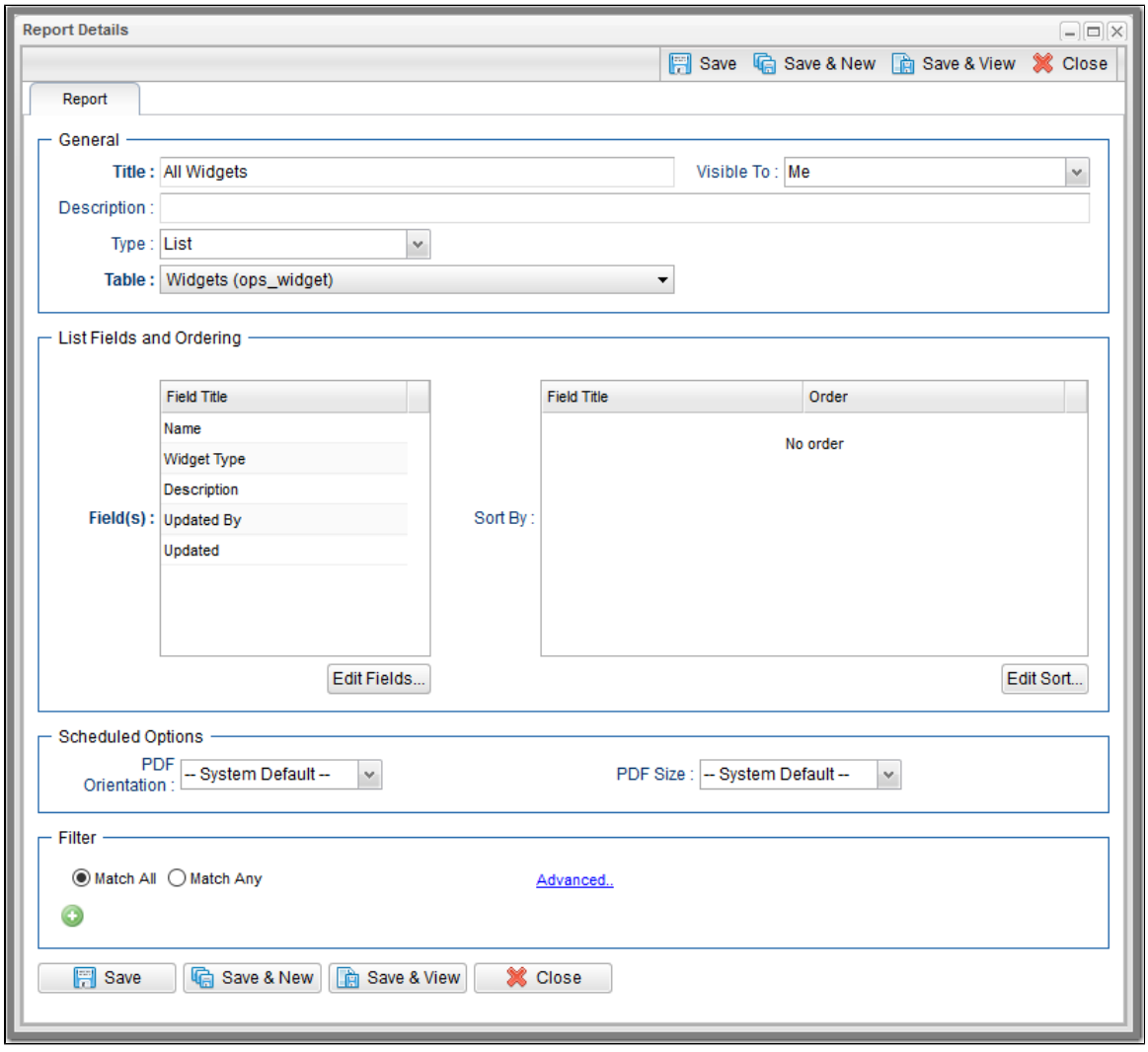
Introduction

In this exercise, we will launch the **Simple Workflow** Workflow (created in the Creating a Simple Workflow tutorial and, on the Activity Monitor, display only tasks assigned to the
Operations Business Service (see Assigning Records to Business Services|Tutorial - Assigning Records to Business Services] tutorial.

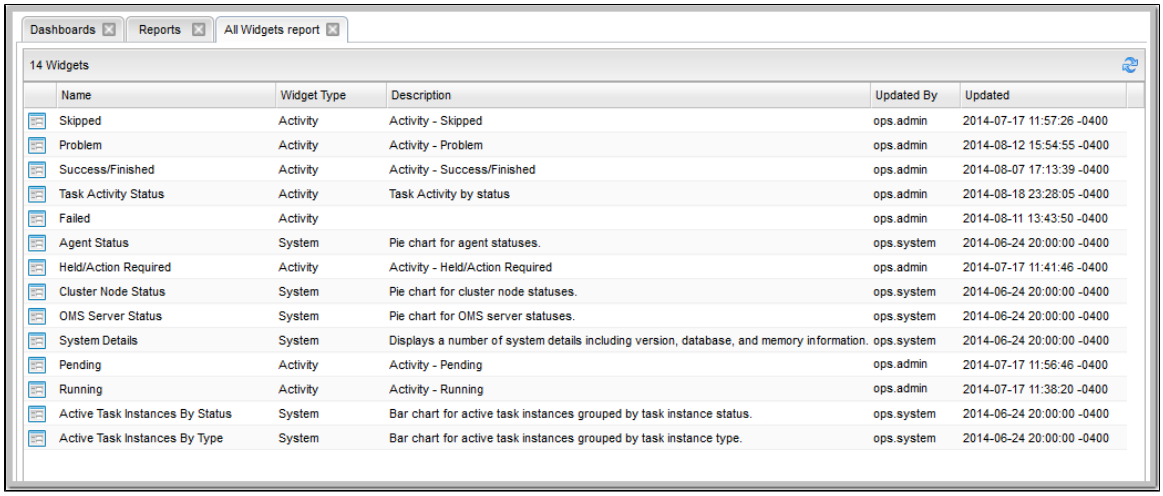
Step 1	Run the **Simple Workflow** Workflow.
Step 2	Display the Activity Monitor, which will list the **Simple Workflow** and all of its tasks.  Instance Name | Type | Status | Invoked By | Start Time | End Time | Updated Timer6 | Timer | Success | Workflow: Simple Workflow | 2014-08-11 14:53:25 -0400 | 2014-08-11 14:55:25 -0400 | 2014-08-11 14:55:25 -0400 Timer5 | Timer | Success | Workflow: Simple Workflow | 2014-08-11 14:53:25 -0400 | 2014-08-11 14:53:35 -0400 | 2014-08-11 14:53:35 -0400 Timer4 | Timer | Success | Workflow: Simple Workflow | 2014-08-11 14:53:14 -0400 | 2014-08-11 14:53:24 -0400 | 2014-08-11 14:53:25 -0400 Timer3 | Timer | Success | Workflow: Simple Workflow | 2014-08-11 14:53:14 -0400 | 2014-08-11 14:53:25 -0400 | 2014-08-11 14:53:25 -0400 Timer2 | Timer | Success | Workflow: Simple Workflow | 2014-08-11 14:53:14 -0400 | 2014-08-11 14:54:04 -0400 | 2014-08-11 14:54:05 -0400 Timer1 | Timer | Success | Workflow: Simple Workflow | 2014-08-11 14:53:04 -0400 | 2014-08-11 14:53:14 -0400 | 2014-08-11 14:53:14 -0400 Simple Workflow | Workflow | Success | Manually Launched | 2014-08-11 14:53:04 -0400 | 2014-08-11 14:55:25 -0400 | 2014-08-11 14:55:25 -0400
Step 3	Apply the following Filter to the list of tasks: • Member of Business Services • contains • Operations The Activity Monitor now displays only tasks assigned to **Operations** Business Service.  Instance Name | Type | Status | Invoked By | Start Time | End Time | Updated Timer3 | Timer | Success | Workflow: Simple Workflow | 2014-08-11 14:53:14 -0400 | 2014-08-11 14:53:25 -0400 | 2014-08-11 14:53:25 -0400 Timer2 | Timer | Success | Workflow: Simple Workflow | 2014-08-11 14:53:14 -0400 | 2014-08-11 14:54:04 -0400 | 2014-08-11 14:54:05 -0400 Timer1 | Timer | Success | Workflow: Simple Workflow | 2014-08-11 14:53:04 -0400 | 2014-08-11 14:53:14 -0400 | 2014-08-11 14:53:14 -0400 Simple Workflow | Workflow | Success | Manually Launched | 2014-08-11 14:53:04 -0400 | 2014-08-11 14:55:25 -0400 | 2014-08-11 14:55:25 -0400

Tutorial - Creating a Report

In this exercise, we will create a report for Widgets.

Step 1	From the Reporting navigation pane, select **Reports**. The Reports list displays.
Step 2	In the empty Report Details below the list, enter / select the following values: • **Title** = All Widgets • **Table** = Widgets (ops_widget) • **Field(s)** = (pre-selected fields display when you click **Edit Fields**)
Step 3	Click the **Save** button. 

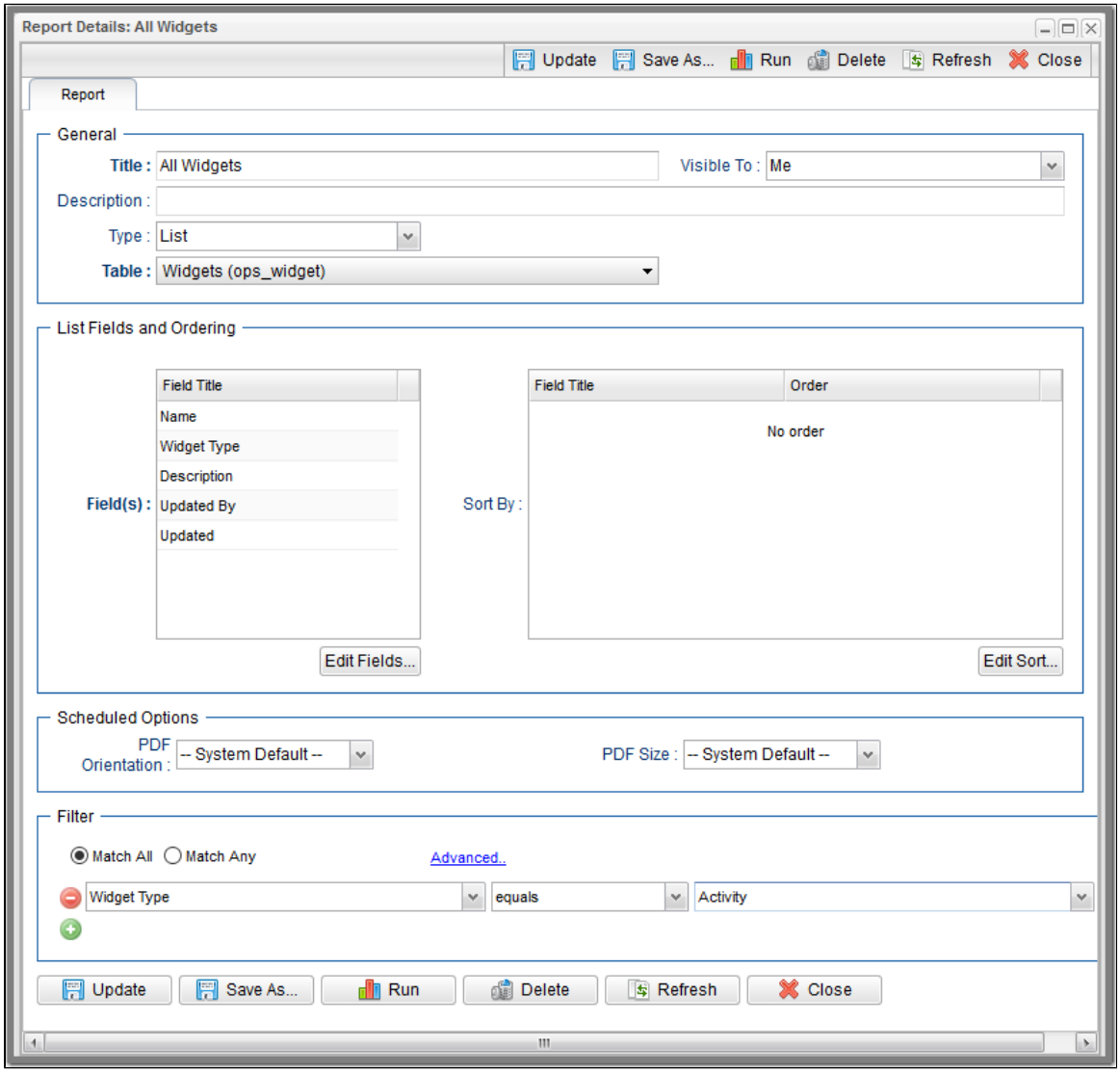
Step 4 Click the **Run** button to run the report, which displays under a new tab.



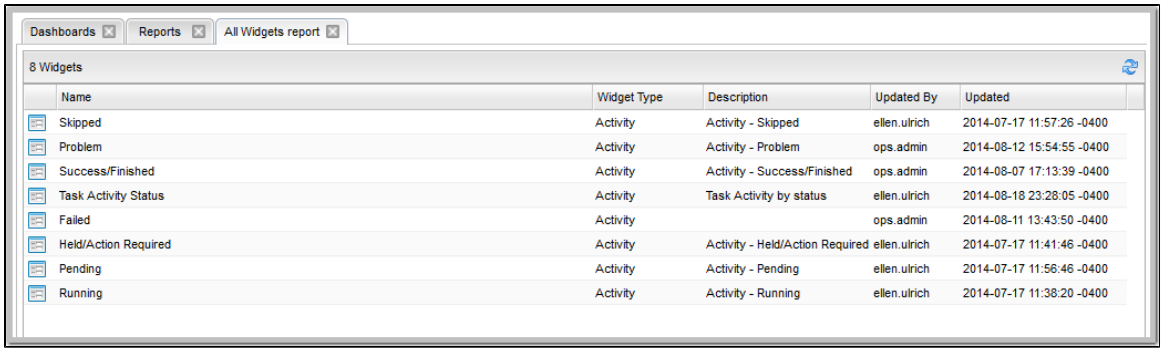
Name	Widget Type	Description	Updated By	Updated
Skipped	Activity	Activity - Skipped	ops.admin	2014-07-17 11:57:26 -0400
Problem	Activity	Activity - Problem	ops.admin	2014-08-12 15:54:55 -0400
Success/Finished	Activity	Activity - Success/Finished	ops.admin	2014-08-07 17:13:39 -0400
Task Activity Status	Activity	Task Activity by status	ops.admin	2014-08-18 23:28:05 -0400
Failed	Activity		ops.admin	2014-08-11 13:43:50 -0400
Agent Status	System	Pie chart for agent statuses.	ops.system	2014-06-24 20:00:00 -0400
Held/Action Required	Activity	Activity - Held/Action Required	ops.admin	2014-07-17 11:41:46 -0400
Cluster Node Status	System	Pie chart for cluster node statuses.	ops.system	2014-06-24 20:00:00 -0400
OMS Server Status	System	Pie chart for OMS server statuses.	ops.system	2014-06-24 20:00:00 -0400
System Details	System	Displays a number of system details including version, database, and memory information.	ops.system	2014-06-24 20:00:00 -0400
Pending	Activity	Activity - Pending	ops.admin	2014-07-17 11:56:46 -0400
Running	Activity	Activity - Running	ops.admin	2014-07-17 11:38:20 -0400
Active Task Instances By Status	System	Bar chart for active task instances grouped by task instance status.	ops.system	2014-06-24 20:00:00 -0400
Active Task Instances By Type	System	Bar chart for active task instances grouped by task instance type.	ops.system	2014-06-24 20:00:00 -0400

Step 5 Return to the Report Details and select the following Filter for the report:

- Widget Type
- equals
- Activity



Step 6 Click the **Update** button, and then click the **Run** button to display the report under a new tab.

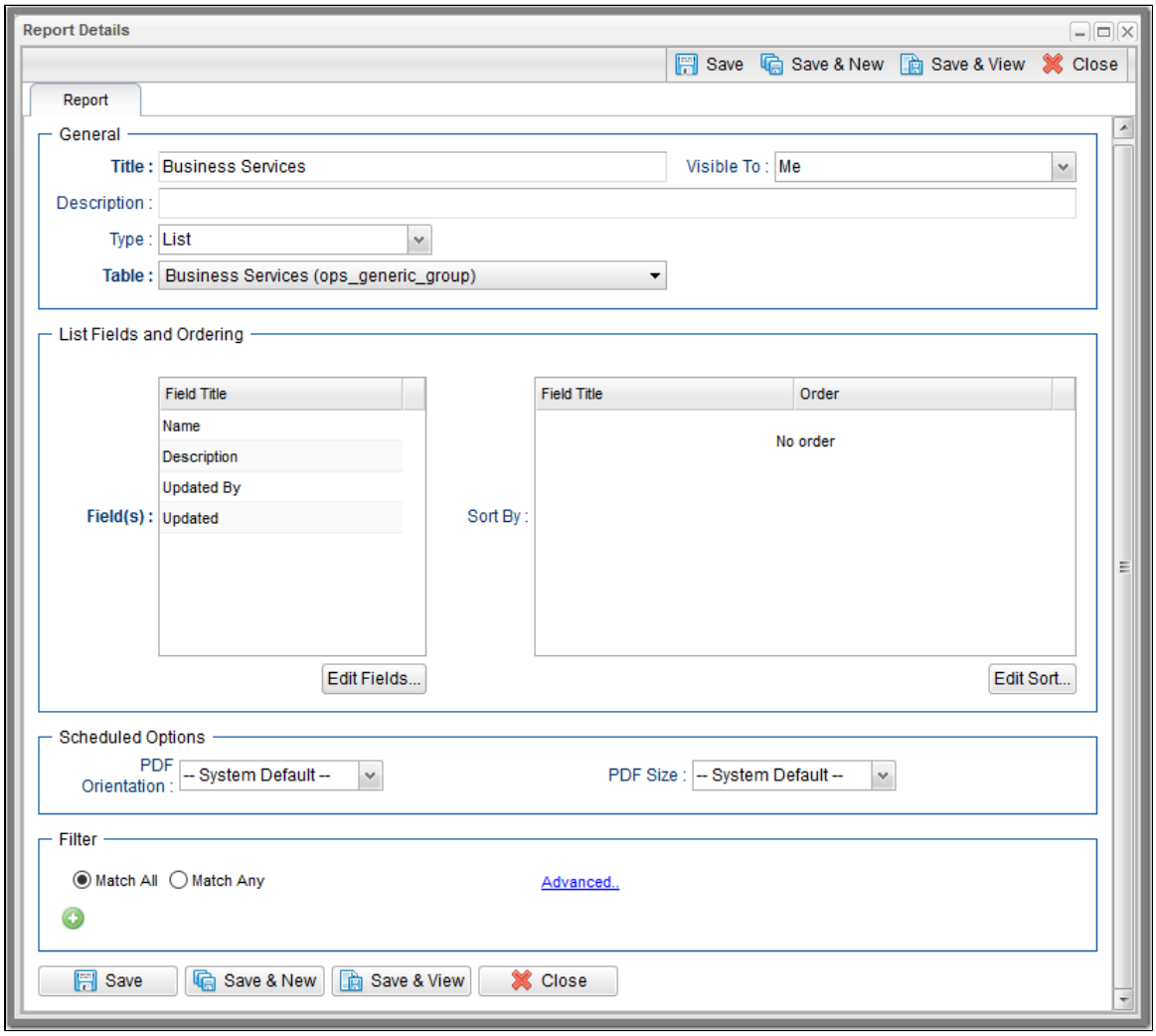


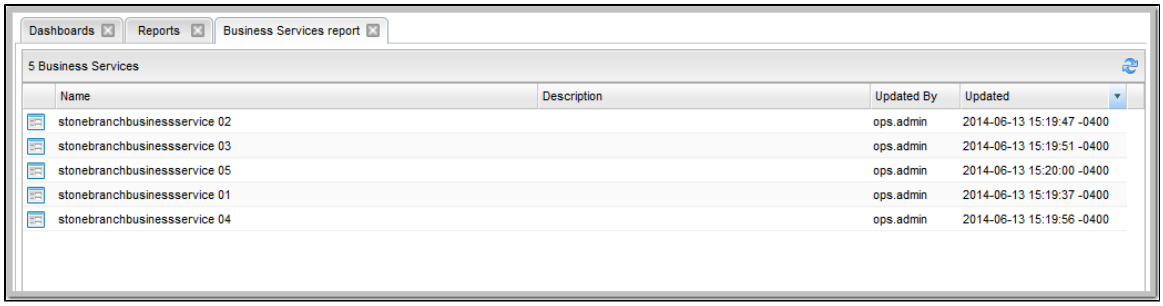
For additional information, see:

- Reports

Tutorial - Creating a Report Based on Business Services

In this exercise, we will create two Activity reports so that users from our hypothetical Operations and Tech Support departments (see the Creating Business Services tutorials) can view activity related to their organizations.

Step 1	From the Reporting navigation pane, select **Reports**. The Reports list displays.
Step 2	Click the **New** button to display empty Report Details.
Step 3	Enter / select the following values: • **Title** - Business Services • **Table** = Business Services (ops_generic_group) • **Field(s)** = (pre-selected fields that display when you click the **Edit Fields** button)
Step 4	Click the **Save** button. 

Step 5	Click the **Run** button. A Business Services report displays under a new tab. 

For additional information, see:

- Reports

Tutorial - Scheduling a Report

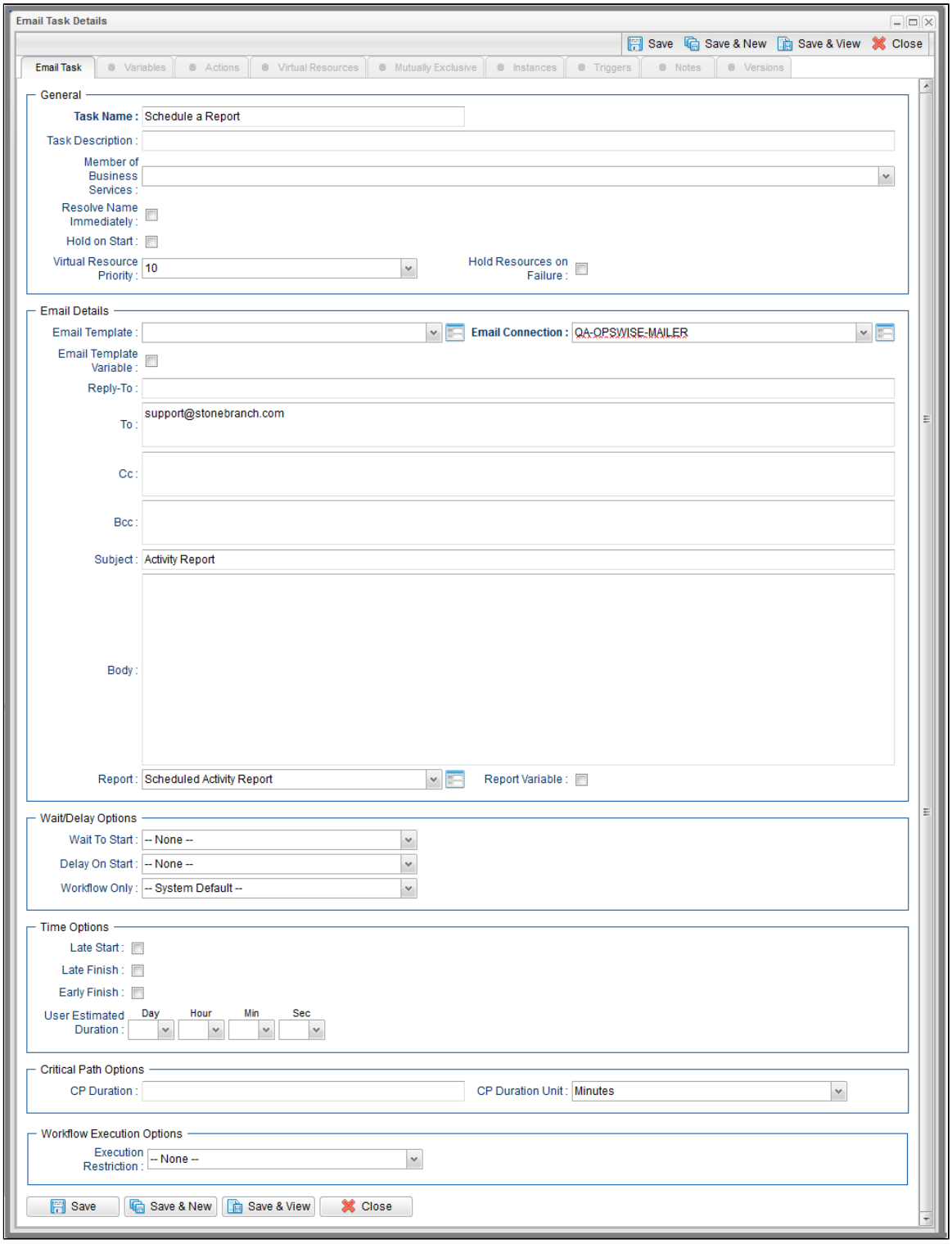
In this exercise, we will schedule a report by triggering an Email Task that specifies the report.

To simplify this exercise, we will manually trigger the Email task. However, you can schedule a report using several methods.

Step 1	Create a Report with the following values: • **Title** = Scheduled Activity Report • **Visible to** = Me • **Type** = List • **Table** = All Task Instances (ops_exec) • **Field(s)** = Instance Name, Type, Status • In **Scheduling Options**, select any **PDF Orientation** and **PDF Size**.
Step 2	From the Automation Center navigation pane, select **Email Tasks**. The Email Tasks list displays.

Step 3 Click the **New** button and enter / select the following values in the empty Email Tasks Details:

- **Task Name** = Schedule a Report
- **Email Connection** = (Select a valid email connection for your environment.)
- **To** = (Enter the email address where you want to send the email.)
- **Subject** = Activity Report
- **Report** = Scheduled Activity Report

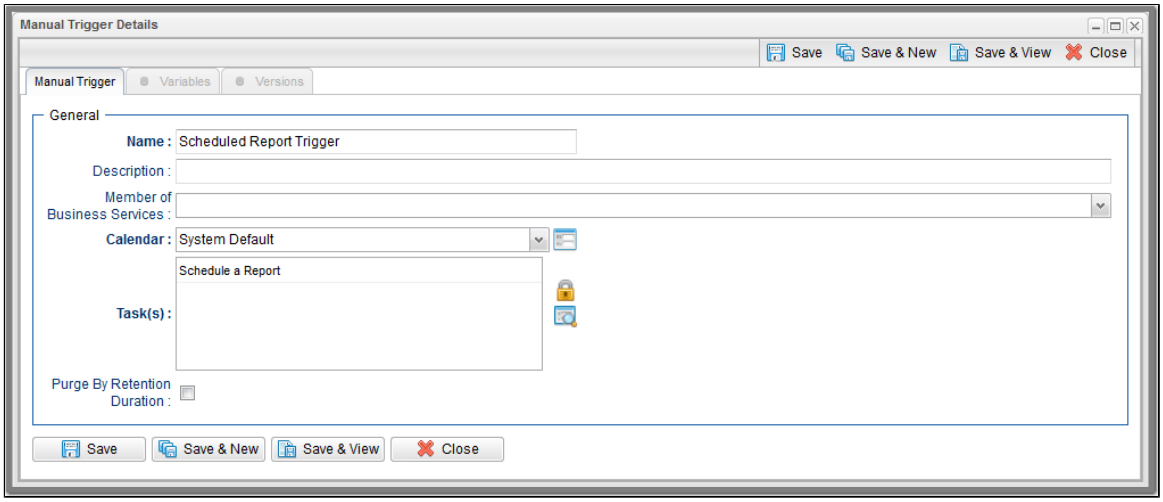


Step 4 Click the **Save** button.

Step 5 From the Automation Center navigation pane, select **Manual Triggers**. The Manual Triggers list displays.

Step 6 Click the **New** button and enter / select the following values in the empty Manual Trigger Details:

- **Name** = Scheduled Report Trigger
- **Tasks(s)** = Schedule a Report



Step 7 Click the **Save** button.

Step 8 In the Time Triggers list, right-click the **Schedule Report Trigger** to display an Action menu.

Step 9 Click **Trigger Now...** and then, on the **Trigger Now...** pop-up dialog, click **Submit**.

Step 10 Open the email sent to the recipient selected in Step 3 and double-click the attached Scheduled Activity Report.pdf to see the report.

Scheduled Activity Report		(17 Task Instances)
Instance Name	**Type**	**Status**
workflow-regression-one-of-each-tasks	Workflow	Success
sap-task-simple-variable	SAP	Success
Taskmon-workflow-simple	Task Monitor	Success
sql-task-mysql-sproc1	Stored Procedure	Success
email-task-builtin-variables	Email	Success
udm-task-simple-variable	File Transfer	Success
Pause for Manual	Manual	Success
ftp-filemon-simple-variable	FTP File Monitor	Success
nix-filemon-simple-variable	File Monitor	Success
Linux check for vsFTP #QUERY#	Application Control	Finished
win-task-launch-simple-variables	Windows	Success
nix-task-launch-simple-variables	Linux/Unix	Success
sysmon-diskspace-simple-variable	System Monitor	Success
Sleep Variable	Timer	Success
sql-task-mysql-select-all	SQL	Success
Indesa-task-simple-variables	Universal Command	Success
Scheduled Activity Report	Email	Running

For additional information, see:

- Reports

Tutorial - Creating Users and Assigning Roles and Permissions

- Introduction
- Create New Users
- Assign Permissions to Groups of Users

Introduction

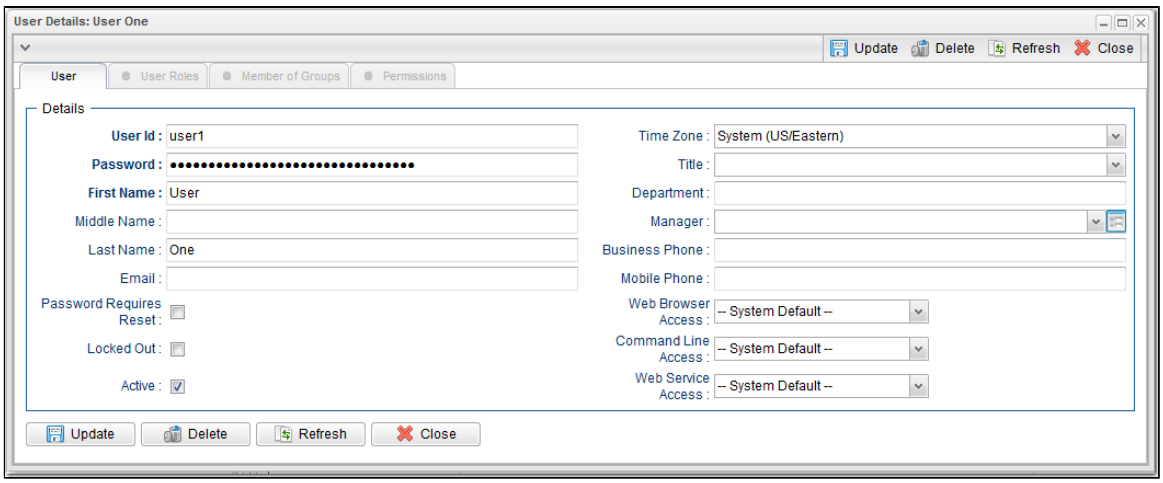
In this exercise, we will create some users related to the Operations and Tech Support departments created in the Creating Business Services tutorial.

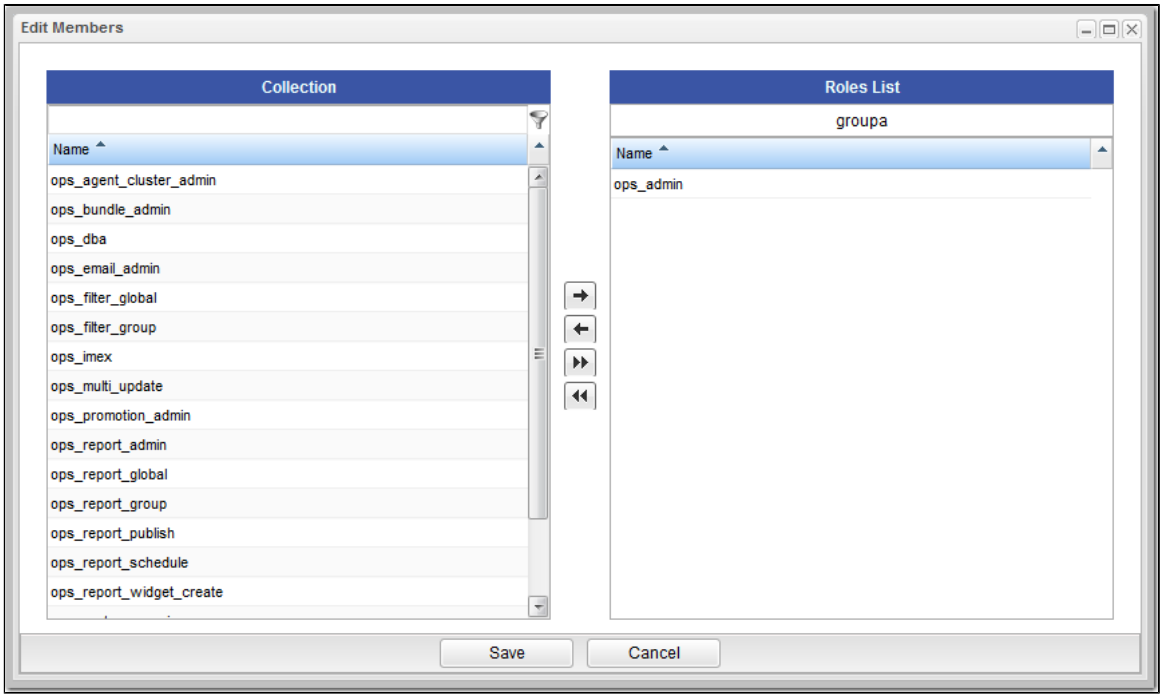
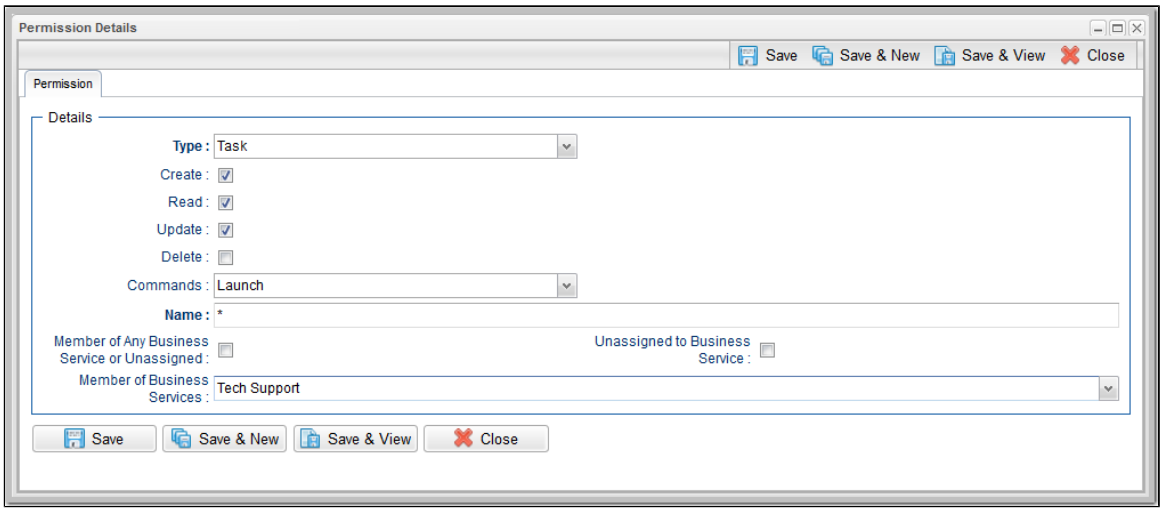
We also will assign access and management rights via user roles and Universal Controller permissions:

- Roles are pre-defined groups of permissions that control access to users, reports (filters), gauges, bundles, and promotions.
- Permissions control who can add, change, delete and control Controller tasks, task instances, triggers, Agents, calendars, and credentials.

Create New Users

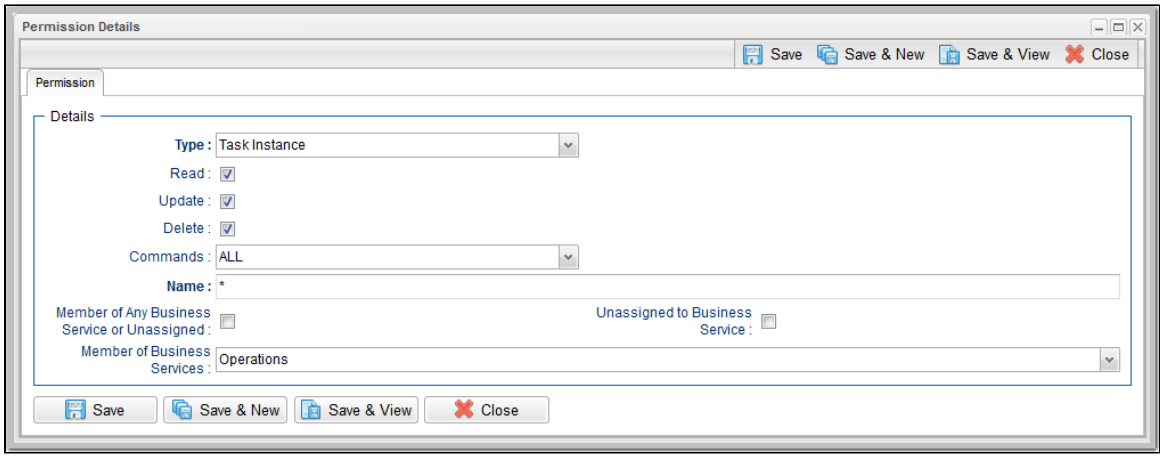
In this exercise, we will create a new user and assign different permissions to it.

Step 1	From the Administration navigation pane, click **Security > Users**. The User list displays.
Step 2	Click the **New** button to display empty User Details for a new user and enter / select the following values: • **User Id** = user1 • **First Name** = User • **Last Name** = One • **Password** = 123 
Step 3	Click the **Save** button, log out of the Controller and then log in as user1.
Step 4	Click on several areas of the user interface. Since user1 has not been assigned any permissions, user1 can view only a limited number of lists and records, and cannot create, modify, or delete any records.
Step 5	Log out of the Controller and log in as ops.admin.
Step 6	Open the user1 record and click the **User Roles** tab.
Step 7	Click **Edit**.

Step 8	Move the **ops_admin** role to the Roles List and click **Save** to provide user1 with full administrative permissions. 
Step 9	Click the User tab and then click the **Update** button.
Step 10	Log out and log back in as user1 to verify that user1 has the same permissions as ops.admin.
Step 11	Log out and log back in as ops.admin, open the user1 record, and remove the **ops.admin** role.
Step 12	Click the Permissions tab, click the **New** button, and in the Permissions Details select the following values: • **Type** = Task • **Create** = enabled • **Read** = enabled • **Update** = enabled • **Commands** = Launch • **Unassigned to Business Service** = disabled • **Member of Business Services** = Tech Support 
Step 13	Click **Save**, log out of the Controller, and log in as user1.
Step 14	user1 now will be able to see the tasks assigned to the Tech Support group, and launch those tasks. However, user1 cannot see them on the Activity Monitor because user1 was not given permissions on Task Instances.

Assign Permissions to Groups of Users

In this exercise we will assign our last user to a group, then assign permissions to the group instead of to the user.

Step 1	Open the user1 record .
Step 2	Click the **Member of Groups** tab to display a list of groups that user1 belongs to.
Step 3	Click the **New** button to display Group Details for a new group.
Step 4	In the **Name** field, enter Group1, and then click the **Save** button. The Member of Groups list now shoes Group1.
Step 5	Open Group1 and click the **Permissions** tab to display the list (currently empty) of permissions for Group1.
Step 6	Click the **New** button and on the Permissions Details, enter / select the following values: • **Type** = Task Instance • **Read, Update, Delete** = enabled • **Commands** = All • **Business Services** = Operations • **Unassigned to Business Service** = disabled These permissions provides all users in the Group1 full permissions on all activity (task instances) related to the Operations Business Service. Any users you assign to Group1 will inherit these permissions. 
Step 7	Click **Save**, log out, and then log in as user1 to check the permissions.

For additional information, see:

- Users and Groups

Tutorial - Creating User Groups and Assigning Permissions

In this exercise, we will create users and user groups, then assign permissions to the groups instead of directly to the users.

Step 1

Create the following three users:

- stonebranch-user-01
- stonebranch-user-02
- stonebranch-user-03

Step 2

Select **Users > Groups** and create a group called **stonebranch-group-01**.

1. Click the **Permissions** tab
2. Click the **New** button and add the following permission:
   - Type = Task Instance
   - Read = enabled
   - Member of Any Business Service or Unassigned = enabled
3. Click the **Group Members** tab and click **Edit**.
4. Add stonebranch-user-01 to the group.

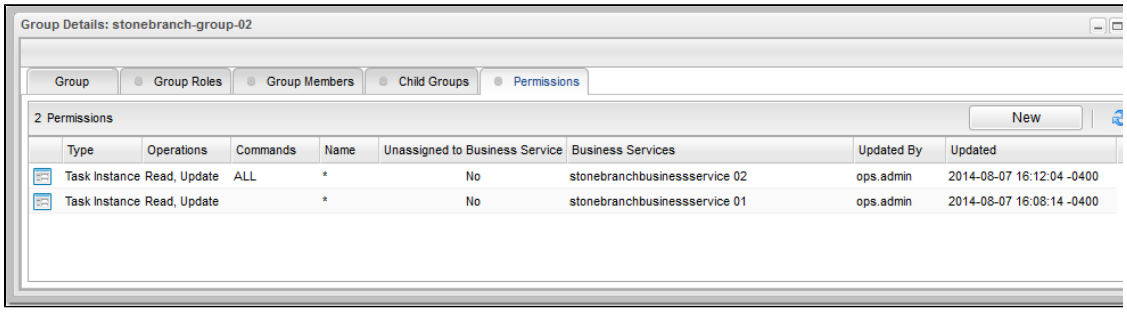
Step 3

Create a group called **stonebranch-group-02**.

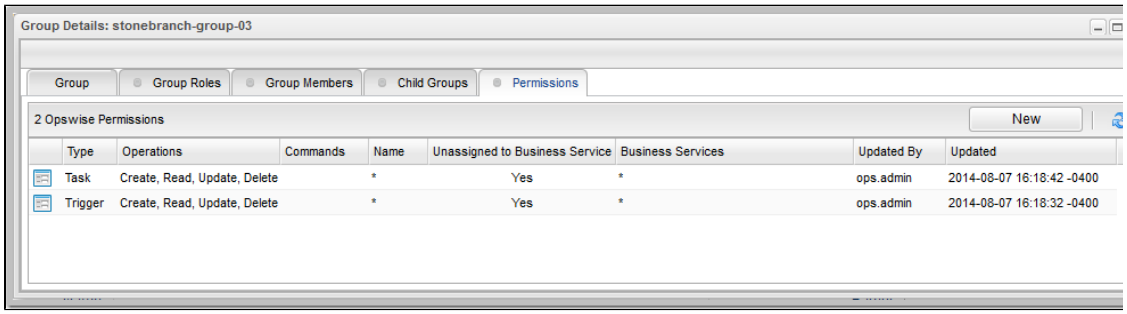
1. Add the following two permissions:
   - Type = Task Instance
   - Read = enabled
   - Update = enabled
   - Commands = None
   - Business Services = stonebranchbusinessservice 01
   - Unassigned to Business Service = disabled

   and

   - Type = Task Instance
   - Read = enabled
   - Update = enabled
   - Commands = All
   - Business Services = stonebranchbusinessservice 02
   - Unassigned to Business Service = disabled



2. Click the **Group Members** tab and **Edit** button to add stonebranch-user-02 to the group.

Step 4	Create a group called **stonebranch-group-03**. 1. Add the following two permissions: • Type = Trigger • Create = enabled • Read = enabled • Update = enabled • Delete = enabled • Commands = None • Member of Any Business Service or Unassigned = enabled and • Type = Task • Create = enabled • Read = enabled • Update = enabled • Delete = enabled • Commands = None • Member of Any Business Service or Unassigned = enabled  2. Add stonebranch-user-03 to the group.
Step 5	Log in as each user and note that each is limited to those functions assigned to the user's group.

For additional information, see:

- Users and Groups

Tutorial - Creating and Promoting a Bundle

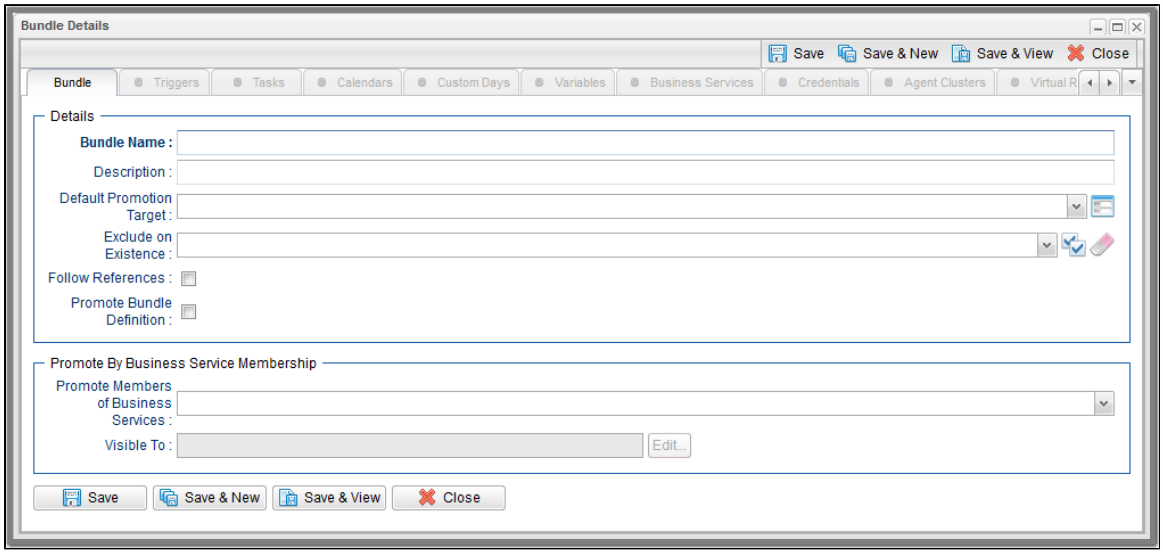
In this exercise, we will create a bundle (group) of Controller records and promote (copy) them from a source machine (a Controller cluster node) to a target machine.

Specifically, we will:

- Create a Bundle.
- Add records to a Bundle.
- Create a promotion target record for the bundle promotion.
- Map Agents on the source machine to Agents on the target machine.
- Promote the bundle to the target.

Create a Bundle

In this procedure, we will create a Bundle record to which other records can be added.

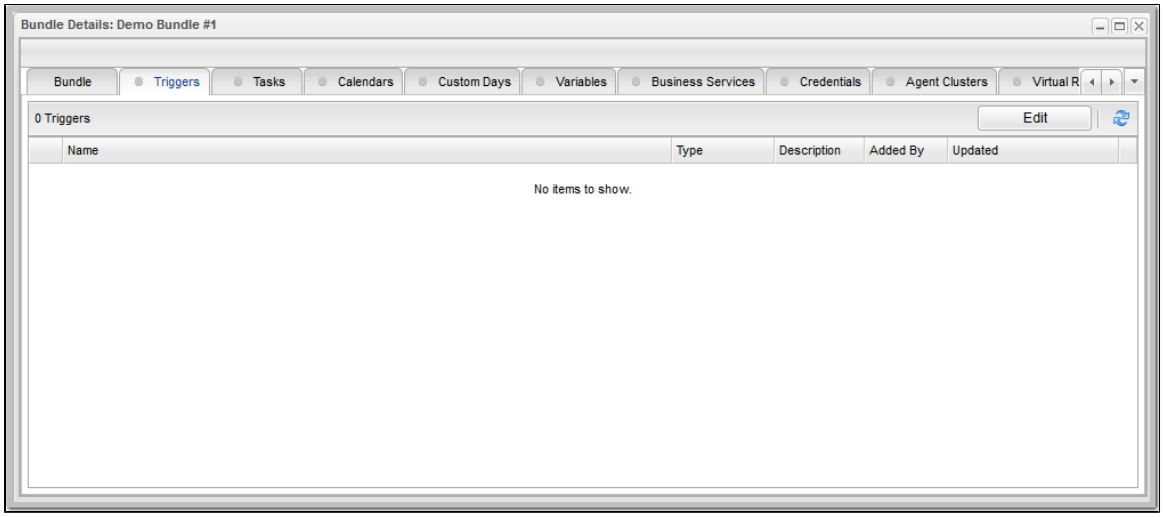
Step 1	From the Bundles & Promotion navigation pane, select **Bundles**. The Bundles list displays.
Step 2	Click the **New** button to display Bundle Details for a new Bundle. Enter a Bundle Name. 
Step 3	Click the **Save** button to create (and close) the Bundle record, which now appears on the Bundle list. You now can add records to the Bundle.

Add Records to the Bundle

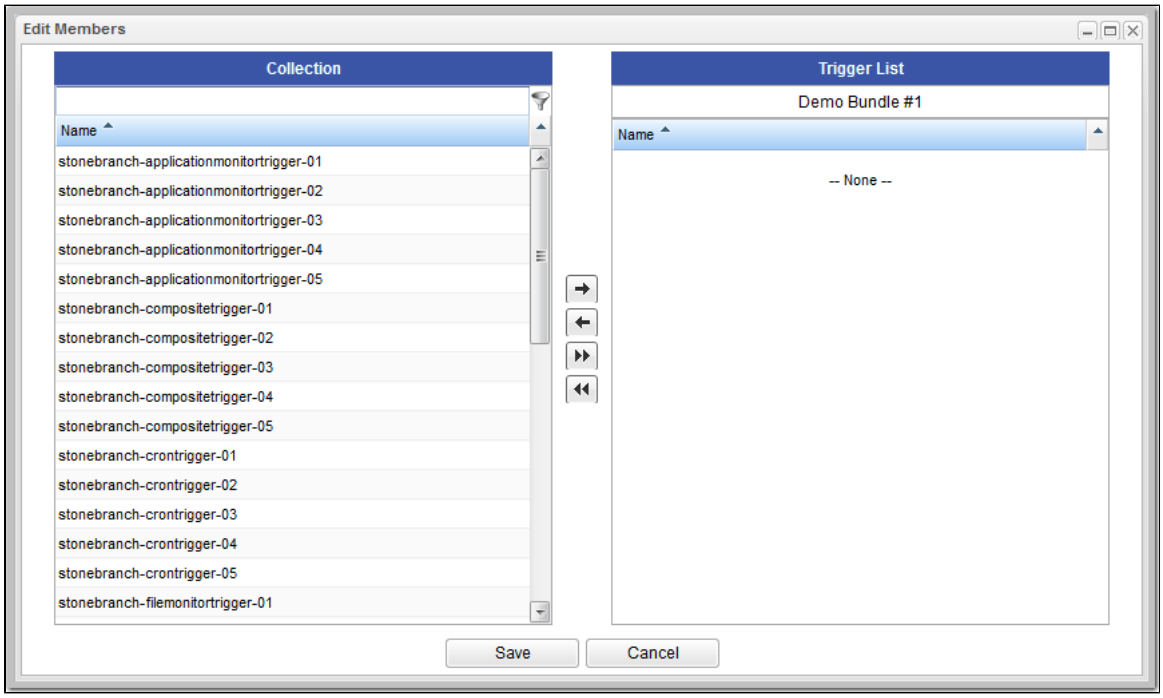
In this procedure, we will add records to the Bundle record that you just created.

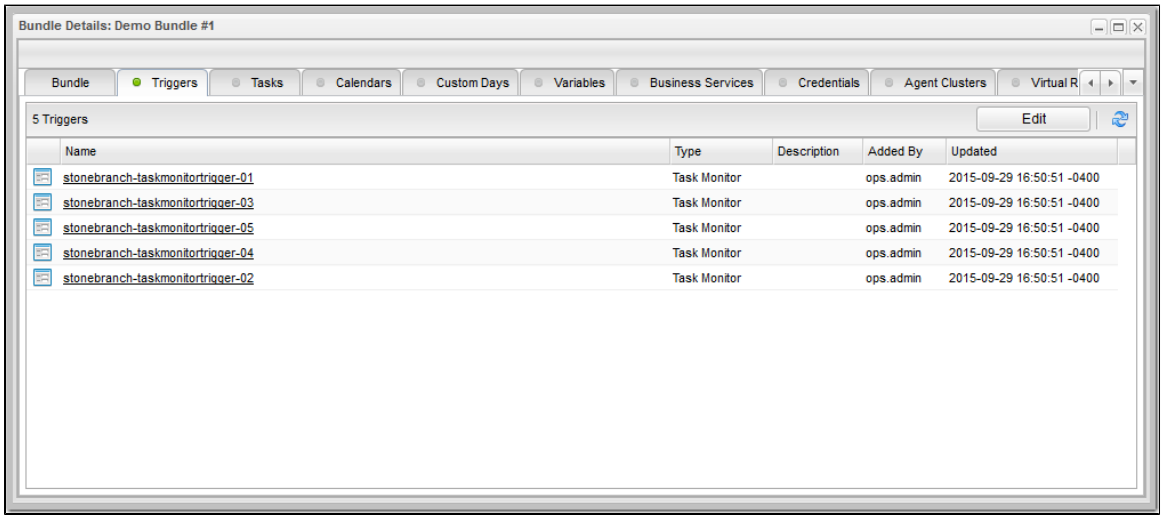
Step 1	Open the Bundle record you just created.

Step 2 Click the **Triggers** tab to display an empty Triggers list.



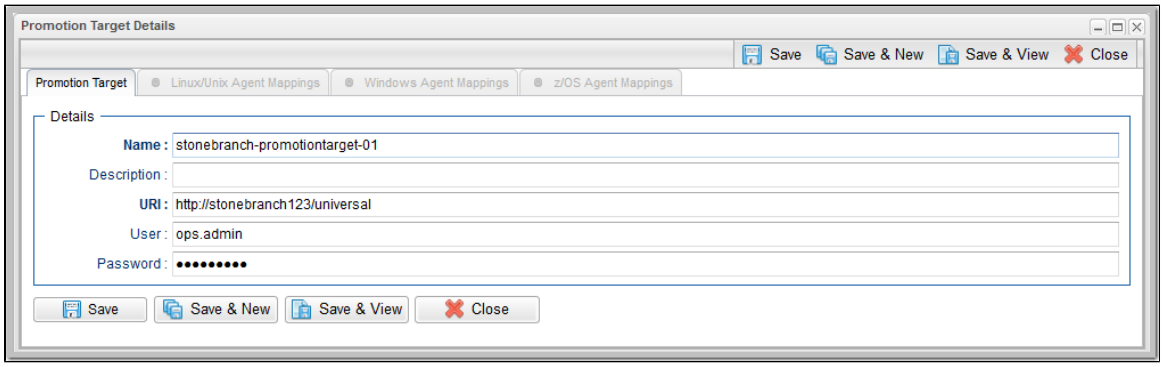
Step 3 Click the **Edit** button to display a list of all Trigger records.



Step 4	Select the Trigger records that you want to add to the Bundle and click the **Save** button. Those records then display in the Triggers list under the Triggers tab. 
Step 5	Repeat Steps 2-4 for any other record types you want to add to the Bundle.
Step 6	Click the **Bundle** tab, the **Update** button, and the **Close** button. You now must create a record identifying the target (Controller cluster node) to which the Bundle will be promoted.

Create a Promotion Target Record for the Bundle

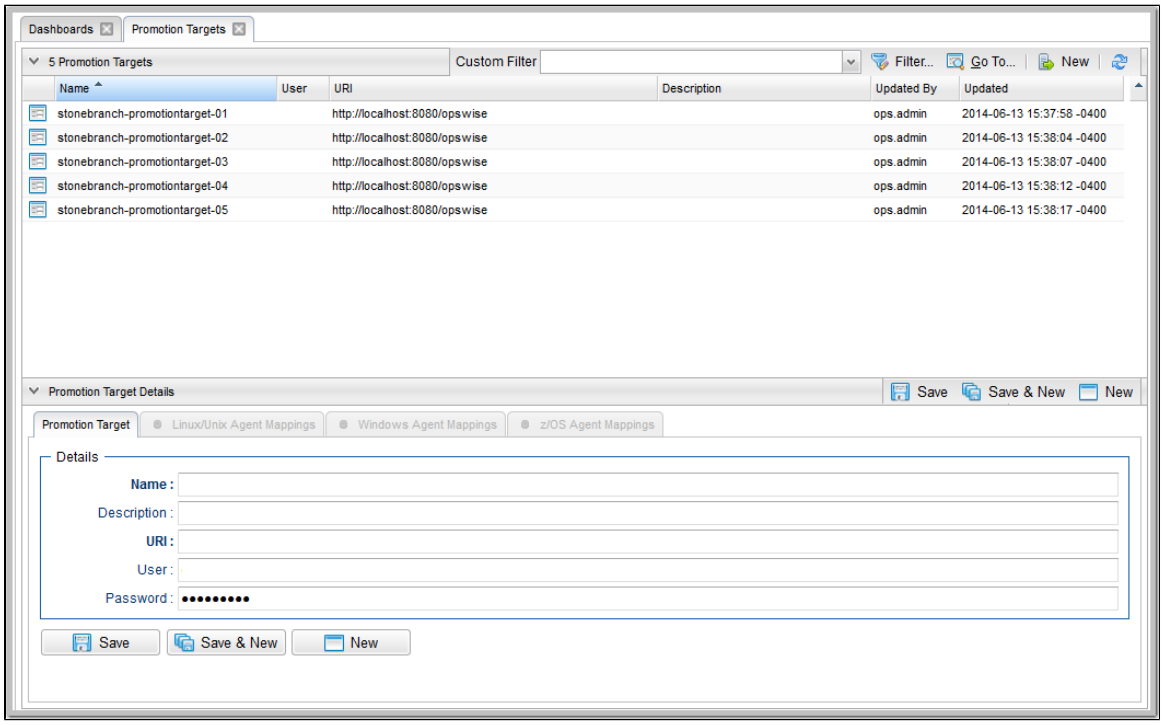
In this procedure, we will create a record identifying the target machine (a Controller cluster node) to which you will promote the Bundle.

Step 1	From the Bundles & Promotion navigation pane, select **Promotion Targets**. The Promotion Targets list displays.
Step 2	Click the **New** button to display Promotion Target Details for a new promotion target.
Step 3	Enter a name and the URI for the promotion target. **Note** By default, the URI of the cluster node that you are logged into displays. You must change this to the URI of a target cluster node, using the default URI format. 
Step 4	Click the **Save** button. You now must map Agents on your source machine to Agents on the target machine.

Map Source Machine Agents to Target Machine Agents

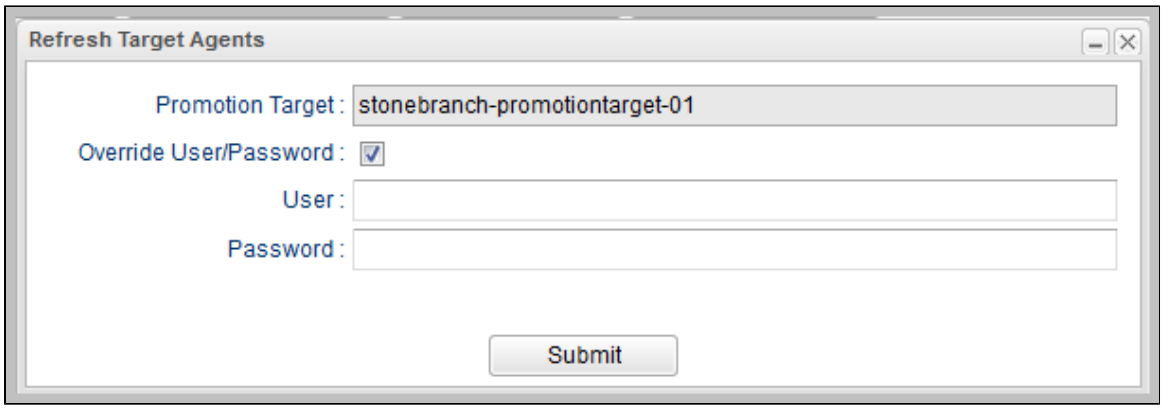
In this procedure, we will map Agents on your source machine to Agents on the selected target machine.

Step 1 From the Bundles & Promotion navigation pane, select **Promotion Targets**. The Promotion Targets list displays.



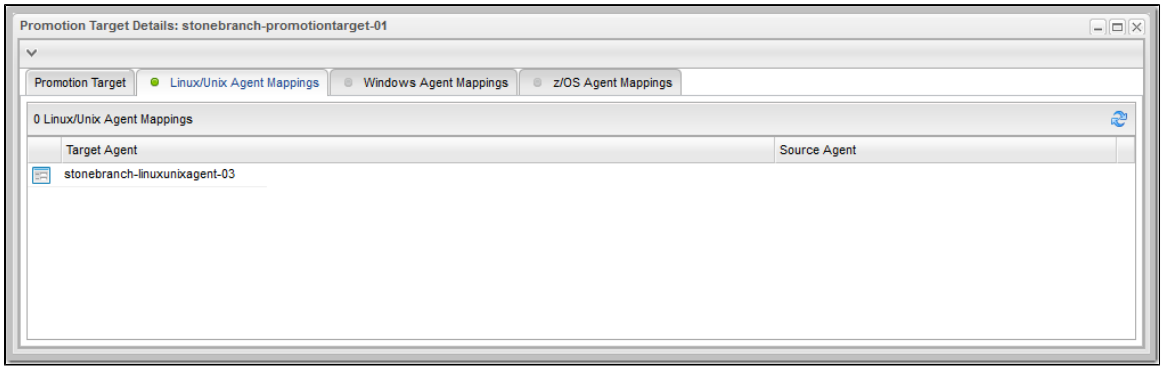
Step 2 Right-click the Promotion Target record that you just created to display an Action menu.

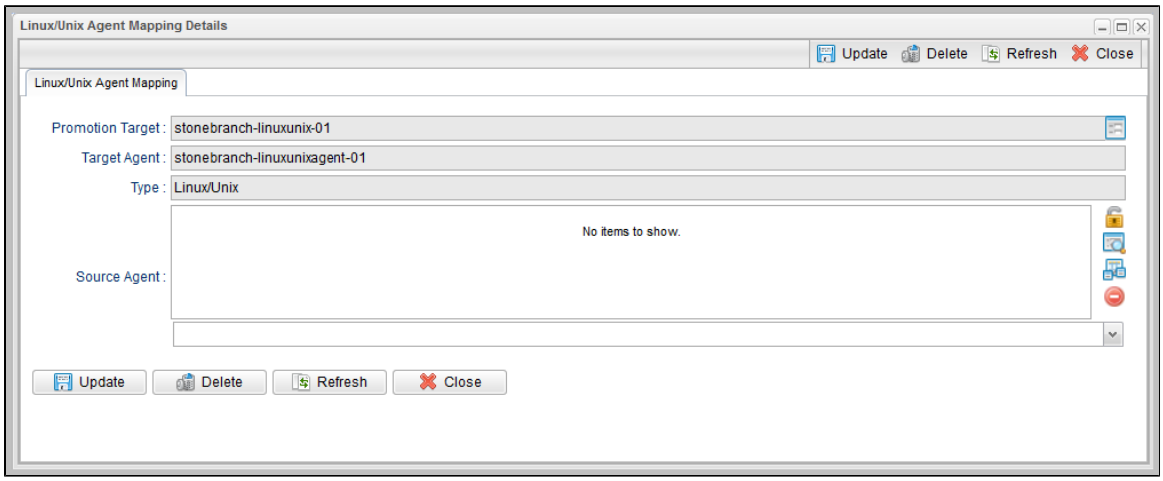
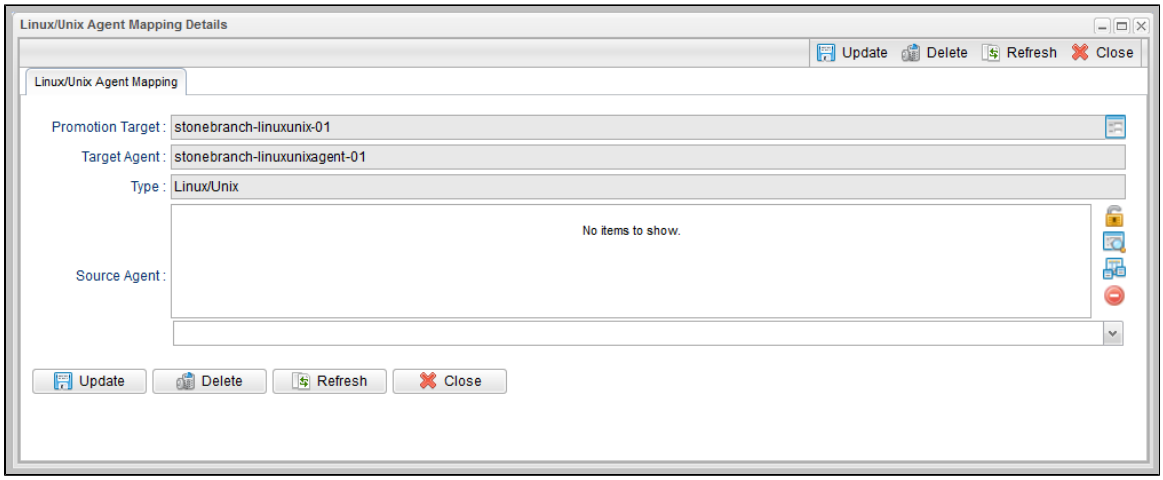
Step 3 Click **Refresh Target Agents** to display a Refresh Target Agents pop-up dialog.



Step 4 Enter the User and Password credentials for the target machine (if they are not provided) and click **Submit**.

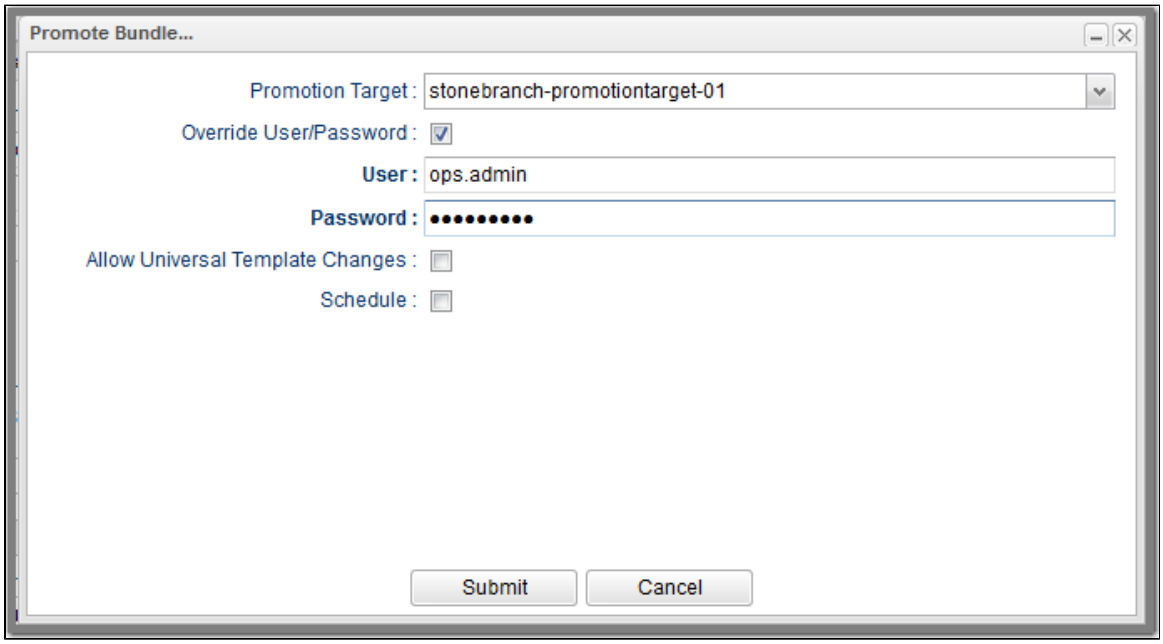
Step 5 Open the Promotion Target record and click an Agent Mappings tab for which there are Agents listed.



Step 6	Click the Details icon next to a Target Agent name to display Agent Mapping Details for that Target Agent. 
Step 7	Click the **Add-Remove Multiple** icon next to the **Source** field to display a list of all Agents on the source machine. 
Step 8	Select the source machine Agent(s) that you want mapped to the selected target machine Agent, and then click **Save**.
Step 9	Repeat Steps 6 to 8 for every Target Agent under all Agent Mappings tab(s) for which there are Agents listed.

Promote the Bundle

In this procedure, we will promote the Bundle that you created to the selected Promotion Target machine.

Step 1	On the Bundles list, right-click the Bundle to be promoted and click **Promote Bundle...** on the Action menu to display a Promote Bundle... pop-up dialog. 
Step 2	Click **Submit**. The source machine Controller logs in to the target machine specified in the URI field of the Promotion Target record and copies the bundled records to the target machine Controller. Based on the specified Agent mapping, the target Agent replaces the source Agent where required.

For additional information, see:

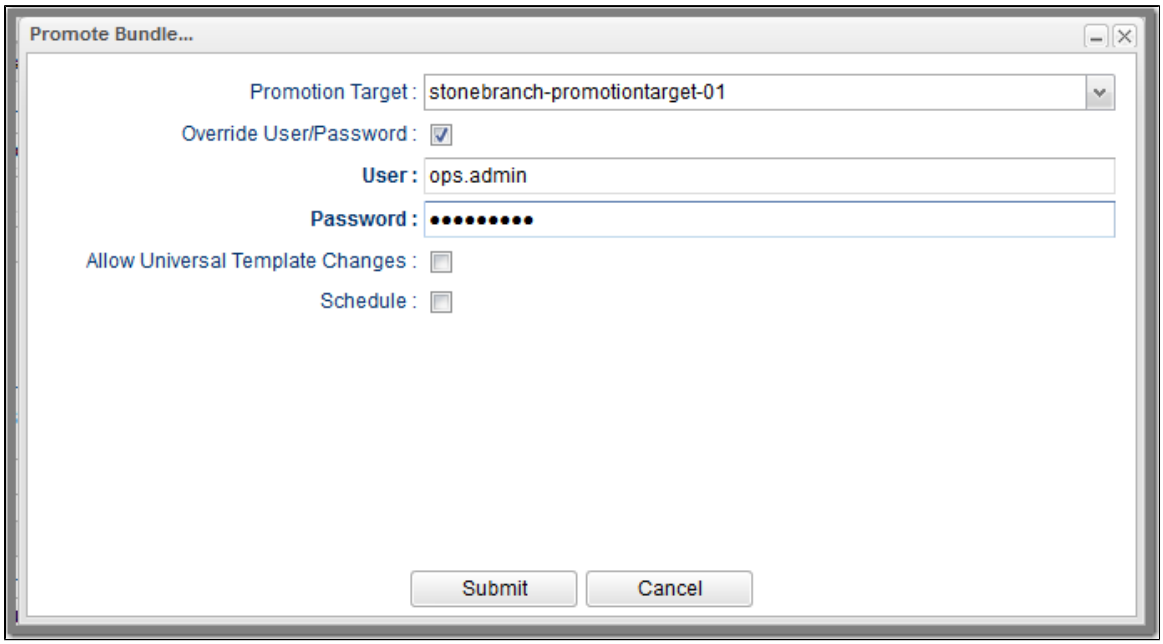
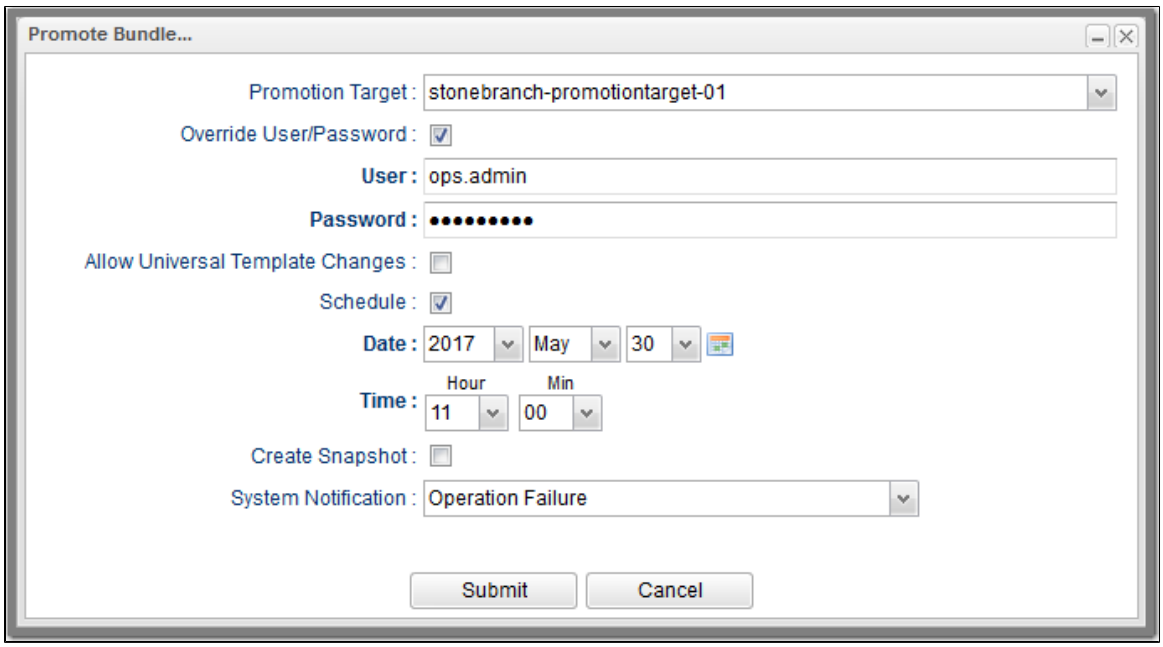
- Bundles and Promotion

Tutorial - Scheduling the Promotion of a Bundle

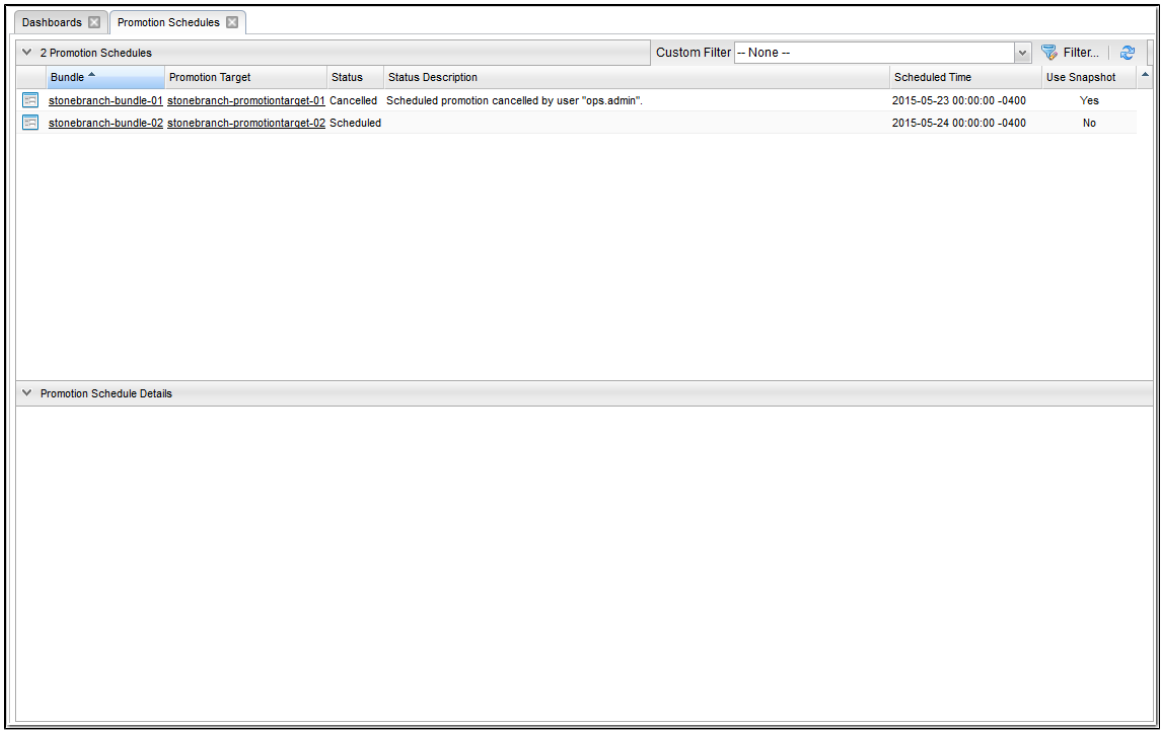
In this exercise, we will select a date and time for the automatic promotion of the Bundle that was created in the Creating and Promoting a Bundle tutorial.

Schedule a Bundle Promotion

In this procedure, we will schedule the promotion of a Bundle to a target machine.

Step 1	From the Bundles & Promotion navigation pane, select **Bundles**. The Bundles list displays.
Step 2	Right-click the Bundle to be scheduled for promotion and click **Promote Bundle...** on the Action menu to display a Promote Bundle... pop-up dialog. 
Step 3	Click the **Schedule** field check box to display additional fields for specifying a promotion schedule. 

Step 4	Select a **Date** and **Time** for the promotion.
Step 5	Click the **Submit** button. The Bundle is listed on the Promotions Schedules list and will be promoted automatically at the scheduled date and time.



For additional information, see:

- Bundles and Promotion