

FORTRA

JAMS
7.7

Installation Guide

Copyright Terms and Conditions

Copyright © Fortra, LLC and its group of companies. All trademarks and registered trademarks are the property of their respective owners.

The content in this document is protected by the Copyright Laws of the United States of America and other countries worldwide. The unauthorized use and/or duplication of this material without express and written permission from Fortra is strictly prohibited. Excerpts and links may be used, provided that full and clear credit is given to Fortra with appropriate and specific direction to the original content.

202402180955

Table of Contents

Installed Components	5
System Requirements	7
JAMS TCP/IP Ports	12
Installing JAMS	12
After Installing JAMS	25
High Availability	49
Disaster Recovery Configuration	52
Agents in JAMS	57
Agent Definitions screen	58
Defining an Agent in JAMS	58
Configuring an Agent in JAMS	59
Deploying an Agent in JAMS	59
Configuring the Connection for Outgoing Agents	60
Canceling Jobs on Agents	60
Agents in Batch Queues	61
Enabling or Disabling an Agent	61
Deleting an Agent in JAMS	61
JAMS Agent for System i	61
JAMS Agent for Windows	63
JAMS Agent Interactive for Windows	65
JAMS Agentd for Linux/Unix	73

Integrations	84
Automate	84
Banner in JAMS	87
Dynamics AX Integration Pack	99
GoAnywhere MFT	108
Automating a JD Edwards Job in JAMS	111
Informatica Cloud In JAMS	113
Micro Focus in JAMS	116
The JAMS Management Pack for MS SCOM	119
NeoBatch In JAMS	127
Netezza in JAMS	131
Oracle	137
PeopleSoft Integration Pack	145
SAP	157
Running SAS Jobs with JAMS	174
Symitar in JAMS	175
z/OS Integration Pack	187

Installed Components

The JAMS environment incorporates several key components that work together to create a unique and powerful scheduling system.

JAMS Desktop Client

The JAMS Desktop Client provides the primary user controls for JAMS and includes:

- Windows-based GUI (JAMS Client)
- PowerShell Cmdlets
- JAMS PowerShell provider
- .NET-based Class Library
- JAMS.exe command line tool

JAMS Scheduler for Windows

This is essentially the heart of the JAMS system. The scheduler provides background services that maintain, schedule and execute JAMS Jobs.

JAMS Scheduler Services Installation

JAMS Scheduler installation also contains other important JAMS components, including:

- JAMS Scheduler Services
- JAMS Server Services
- JAMS Executor Services

NOTE: If this is the initial installation of the JAMS Scheduler, you will be prompted for additional information when the installer configures the JAMS Database.

JAMS Agents

There are many different Agents available for JAMS, each designed and configured for a specific platform and purpose. See [Agents in JAMS](#) for more information.

JAMS Web Client / REST API

Optional. Installing the JAMS Web Client allows you to run and manage the automation environment from any web-connected device. If you do not install the JAMS Web Client during the initial install of JAMS, you can install the Web Client at a later time. See [Installing the JAMS Web Client](#) for more information.

JAMS Desktop Client Installation Components

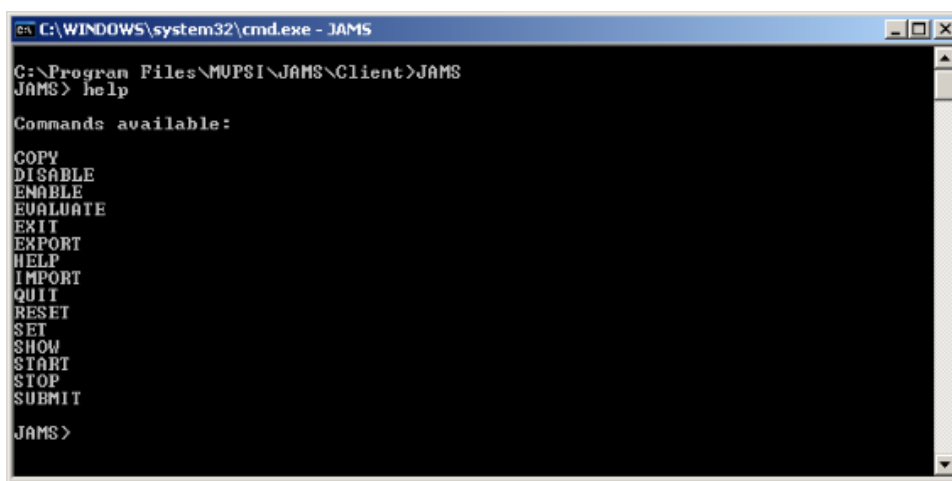
The JAMS Desktop Client installation includes the following items:

- A Start Menu shortcut to JAMSWin.exe
- The JAMSShr.dll Class Library
- The JAMS PowerShell Modules (PowerShell must be installed before you install JAMS)
- The JAMS.exe command line tool
- JAMS Help files

JAMS Command Line Environment

In addition to the JAMS Client, the JAMS PowerShell Modules, and the .NET Controls, JAMS offers a command line interpreter to manage your scheduling environment. When the JAMS Desktop Client is installed, it also includes a command line tool (JAMS.EXE).

1. Start JAMS.EXE from the command line to see the JAMS> prompt. This establishes the connection to the JAMS Server.
2. Type **HELP** to view all the supporting commands.
3. Type **HELP COMMAND** to get additional assistance for a command, such as **HELP EXPORT JOB**.



```
C:\WINDOWS\system32\cmd.exe - JAMS
C:\Program Files\MUPSI\JAMS\Client>JAMS
JAMS> help
Commands available:
COPY
DISABLE
ENABLE
EVALUATE
EXIT
EXPORT
HELP
IMPORT
QUIT
RESET
SET
SHOW
START
STOP
SUBMIT
JAMS>
```

System Requirements

	JAMS Scheduler	JAMS Agent	JAMS Client
Software Prerequisites	<ul style="list-style-type: none"> • .NET Framework 4.8 and .NET 6 Hosting Bundle • Microsoft Message Queue (installed with JAMS) • Microsoft SQL Server 2019, 2017, 2016, or 2014 • Microsoft Windows Server 2022, 2019, 2016, or 2012R2 • Java Runtime Environment (custom version installed with JAMS) 	<ul style="list-style-type: none"> • .NET Framework 4.8 	<ul style="list-style-type: none"> • .NET Framework 4.8
Hardware Server (minimum)	<ul style="list-style-type: none"> • Processor: Dual Core ~2.5 GHZ (x86 or X64) • Memory: 8 GB • Disk Space: 10 GB (100 GB if not using Common Log Location), in addition to the disk space required by the OS 	<ul style="list-style-type: none"> • Processor: 1 GHZ (x86 or x64) • Memory: 1 GB • Disk Space: 75 MB in addition to the disk space required by the OS 	<ul style="list-style-type: none"> • Processor: 1 GHZ (x86 or X64) • Memory: 1 GB • Disk Space: 300 MB in addition to the disk space required by the OS

	JAMS Scheduler	JAMS Agent	JAMS Client
Hardware Server (50,000 to 100,000 Jobs per Day)	<ul style="list-style-type: none"> • Processor: 8 Core ~3.2 GHZ (x86 or X64) • Memory: 32+ GB • Disk Space: 1+TB • Recommended: A separate SQL Cluster for the JAMS SQL back-end 	<ul style="list-style-type: none"> • Processor: 1 GHZ (x86 or x64) • Memory: 1 GB • Disk Space: 75 MB in addition to the disk space required by the OS 	<ul style="list-style-type: none"> • Processor: 1 GHZ (x86 or X64) • Memory: 1 GB • Disk Space: 300 MB in addition to the disk space required by the OS
Hardware Server (>100,000 jobs per day)	<ul style="list-style-type: none"> • Processor: 16 Cores, ~3.2GHZ • Memory: 64+ GB • Disk Space: 1+ TB • One server hosting the JAMS Scheduler • One server reserved for the SQL Server with multiple high-speed SCSI drives 	<ul style="list-style-type: none"> • Processor: 1 GHZ (x86 or x64) • Memory: 1 GB • Disk Space: 75 MB in addition to the disk space required by the OS 	<ul style="list-style-type: none"> • Processor: 1 GHZ (x86 or X64) • Memory: 1 GB • Disk Space: 300 MB in addition to the disk space required by the OS

NOTE: The JAMS Site (REST API) requires Microsoft Windows Server 2022, 2019, 2016, or 2012R2.

Permissions

To install JAMS, ensure that the user performing the installation is logged in as an account with administrator privileges to the server where JAMS will be installed. In addition, the account performing the install must have the ability to create a database on the database server. JAMS creates the database locally by default.

NOTE: Typically, giving the user dbcreator or sysadmin rights on the SQL Instance where the JAMS Database will be created provides the necessary permissions. These permissions are needed only for installing JAMS and you can remove them after the installation is complete.

Prerequisites

Before installing JAMS, review the prerequisite software and minimum hardware requirements. The JAMS installer checks for all prerequisites.

.NET Requirements

The JAMS Client, JAMS Scheduler, and JAMS Windows Agent require .NET Framework 4.8. The JAMS Scheduler also requires the .NET 6 Hosting Bundle. If it is not installed, the JAMS installer displays a message and attempts to install it. If an internet connection is not available, the JAMS installer displays a message and you will need to manually download and install the correct .NET version. Go to the [.NET Framework download site](#) or [.NET 6 download site](#) to download the correct version.

Microsoft Message Queue (MSMQ)

The JAMS Scheduler requires the Microsoft Message Queue (MSMQ). When installing the JAMS Scheduler, the installer checks for the MSMQ and, if it's not installed, will prompt you to install it.

NOTE: MSMQ is included with Microsoft Windows, but it is optional and not installed by default.

Microsoft SQL Server

The JAMS Scheduler requires Microsoft SQL Server 2019, 2017, 2016, or 2014. The SQL Server does not need to be installed on the same machine as the JAMS Scheduler. The database is fully supported in a clustered configuration.

NOTE: Ensure the SQL Collation Setting SQL_Latin1_General_CP1_CI_AS is used for the SQL Server.

If you or your DBA choose to install the JAMS Scheduler component, the installer first checks for a connection to a SQL Server. If SQL Server is not installed locally, the installer prompts you to install the SQL Server 2019 Express Edition if the operating system is Windows Server 2016/Windows 10 or higher.

You also have the option of using SQL Server on a different machine. When the installer checks for the SQL Server, it only looks for the default instance names of MSSQLSERVER and SQLEXPRESS. If you have a different instance preference, decline the SQL Express installation (clear the check box) and specify your preferred instance in the SQL Server\Instance field.

If you have a different instance preference, decline the SQL Express installation and specify your preferred instance when the JAMS installer asks for the SQL server name and instance.

In order to create/update or run SSISDirect, SQLStoredProc, SQLAgent or SSRS Execution Method Jobs, the Client Tools Backwards Compatibility and Client Tools SDK Shared Features are required. Microsoft SQL Server 2019, 2017, 2016, 2014, 2012 Shared Features are supported.

The Shared Features are required on the machine where the JAMS Client is running to create/update the Job Definitions and the machine where the Job executes to run the Job.

It's best practice to run SSISDirect, SQLStoredProc, SQLAgent or SSRS Execution Method Jobs on the Server where those Jobs reside and the SQL Libraries are present.

Java Runtime Environment

JAMS installs a custom version of the Java Runtime Environment on the Scheduler server. No internet access is required. This version does not replace any existing installation of Java, and JAMS is not affected by other pre-existing versions of Java that may be on the machine.

PowerShell

JAMS Supports PowerShell V3 and higher.

The JAMS Client component includes a JAMS PowerShell Module that contains JAMS cmdlets and a JAMS Provider. PowerShell is not required, but you must install PowerShell before installing JAMS to use the JAMS PowerShell Module. If you install PowerShell after installing the JAMS Desktop Client, reinstall the JAMS Desktop Client to access the JAMS PowerShell Module.

NOTE: JAMS V6.X Agents support PowerShell V2 and may be connected to JAMS V7.X Scheduler Servers. By leveraging a V6.X Agent, V7 users may run PowerShell V2.

Hardware Server Specifications for the JAMS Scheduler

- Processor: Dual Core ~2.5 GHZ (x86 or x64)
- Memory: 8 GB
- Disk Space: 10 GB in addition to the disk space required by the OS.

These are the minimum system requirements that can support running thousands of Jobs per day. Heavy loads, for example, over 100,000 Jobs per day, require more resources.

For database planning, the JAMS test lab runs approximately 1,000 Jobs per day using one instance of JAMS. Retaining 30 days of information generates a database approximately 110 Mb in size.

For Sites Running Over 100,000 Jobs Per Day

Based on customer implementations where more than 100,000 Jobs are executed each day, the JAMS installation should include:

- Two Windows servers, each with Dual Quad 3.2 GHz CPUs and 64 GB of RAM (or more).
- Specifically, one of the above servers hosting the JAMS Scheduler and the second server reserved for the SQL Server with multiple high-speed SCSI drives.

Hardware Server Specs for JAMS Agents

- Processor: 1 GHZ (x86 or x64)
- Memory: 1 GB
- Disk Space: 75 MB

Hardware Server Specs for the JAMS Client

- Processor: 1 GHZ (x86 or x64)
- Memory: 1 GB
- Disk Space: 300 MB

Supporting JAMS in a Virtual Environment

You can run JAMS on any supported operating system in a virtualized environment, such as VMware and HyperV. JAMS is supported in cloud-based environments such as AWS or Microsoft Azure. The JAMS database can be deployed into Microsoft Azure SQL Database or AWS RDS services.

NOTE: Azure SQL Managed Instance is not supported.

JAMS TCP/IP Ports

JAMS uses the following TCP/IP Ports for both outgoing and incoming connections.

JAMSAgent.exe: The JAMS Agent listens on port **77** for requests to execute Jobs from other machines running JAMS.

JAMSServer.exe: The JAMS Server listens on port **773** for connection requests from JAMS Clients.

JAMSScheduler.exe: The JAMS Scheduler listens on port **2773** for connections from the JAMS Server and JAMS Debugger. This is usually confined to the local machine.

JAMSExecutor.exe: The JAMS Executor listens on port **3773** for connections from the JAMS Scheduler and JAMS Debugger. This is usually confined to the local machine.

JAMSScheduler.exe: The JAMS Scheduler listens on port **4773** for connections from the Failover JAMS Scheduler. This is configured in the [Failover.config file](#).

Common Configuration: If no port is specified in the Common.config file in C:\Program Files\MVPS\JAMS\Scheduler (default location), the default SQL port of **1433** is used.

Web Interface HTTP: The JAMS Web Interface uses port **80** by default for HTTP.

Web Interface HTTPS: The JAMS Web Interface uses port **443** by default for HTTPS.

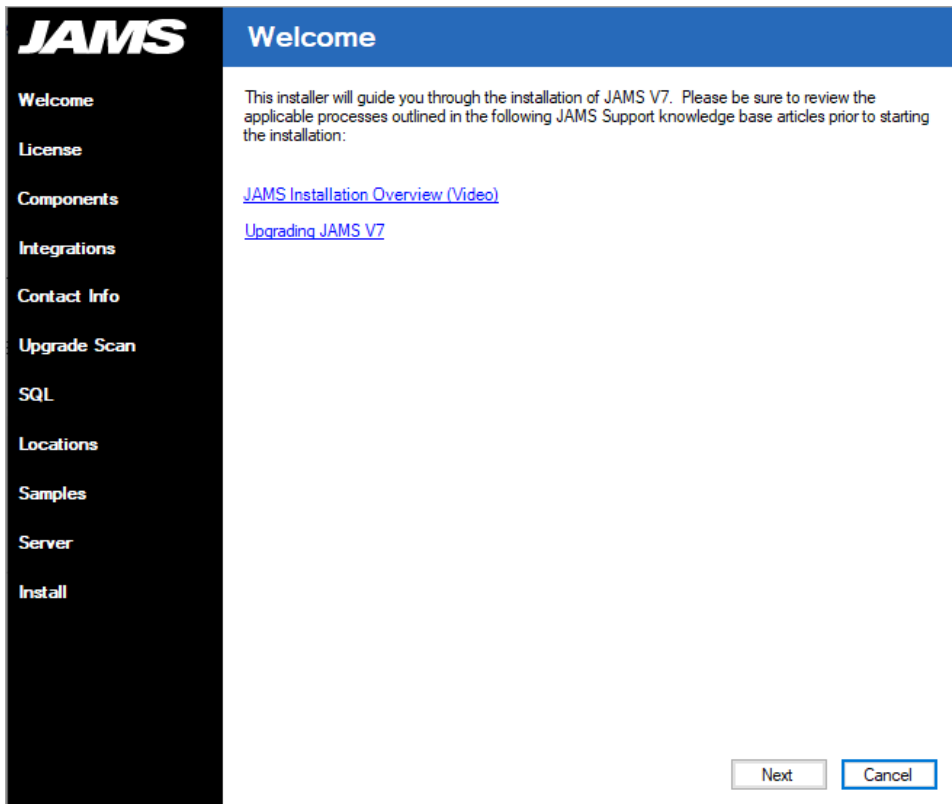
Installing JAMS

Follow the steps below to install JAMS on a Windows computer.

1. Log on to Windows using an account with administrator privileges and the ability to create the JAMS Database in SQL Server. (Typically, using an account with dbcreator or sysadmin rights in SQL Server provides the necessary permissions.)

NOTE: Users must also be able to create a database on the database server. The default database server is the local machine.

2. Right-click the **SetupJAMS** executable and select **Run as Administrator** to start the JAMS installer.
3. On the Welcome page, verify this installation is the proper edition and version.
4. Click **Next**.



5. On the License Agreement page, review the JAMS license agreement.
6. Select the checkbox next to **I accept this license agreement**.

7. Click Next.

JAMS License Agreement

Welcome

Please read and accept the JAMS license agreement.

Help/Systems, LLC Software License and Services Agreement

This Software License and Services Agreement (the "Agreement") is between Help/Systems, LLC, a Delaware limited liability company with offices located at 6455 City West Parkway, Eden Prairie, MN 55344 USA ("HelpSystems") and the individual or type of Person intending to download, install or otherwise use the Software (the "Customer" and together with HelpSystems, the "Parties").

HELPSYSTEMS PROVIDES THE SOFTWARE SOLELY ON THE TERMS AND CONDITIONS SET FORTH IN THIS AGREEMENT AND ON THE CONDITION THAT CUSTOMER ACCEPTS AND COMPLIES WITH THEM. BY DOWNLOADING, INSTALLING OR OTHERWISE USING THE SOFTWARE, CUSTOMER (A) ACCEPTS THIS AGREEMENT AND AGREES THAT CUSTOMER IS LEGALLY BOUND BY ITS TERMS; AND (B) REPRESENTS AND WARRANTS THAT: (I) IF AN INDIVIDUAL, CUSTOMER IS 18 YEARS OF AGE OR OLDER; AND (II) CUSTOMER HAS THE RIGHT, POWER AND AUTHORITY TO ENTER INTO THIS AGREEMENT AND BIND CUSTOMER TO ITS TERMS. IF CUSTOMER DOES NOT AGREE TO THE TERMS OF THIS AGREEMENT, HELPSYSTEMS WILL NOT AND DOES NOT LICENSE THE SOFTWARE TO CUSTOMER AND CUSTOMER MUST NOT DOWNLOAD, INSTALL OR USE THE SOFTWARE OR SOFTWARE DOCUMENTATION.

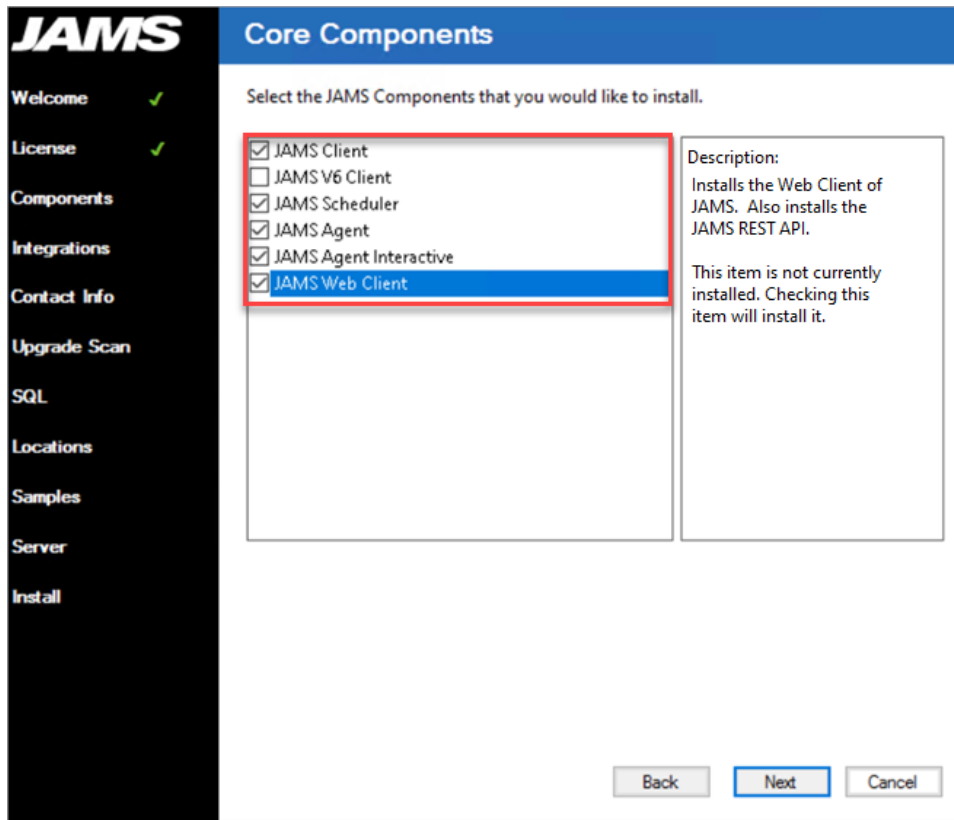
SECTION 1. Definitions.

☒ I accept this license agreement

Back Next Cancel

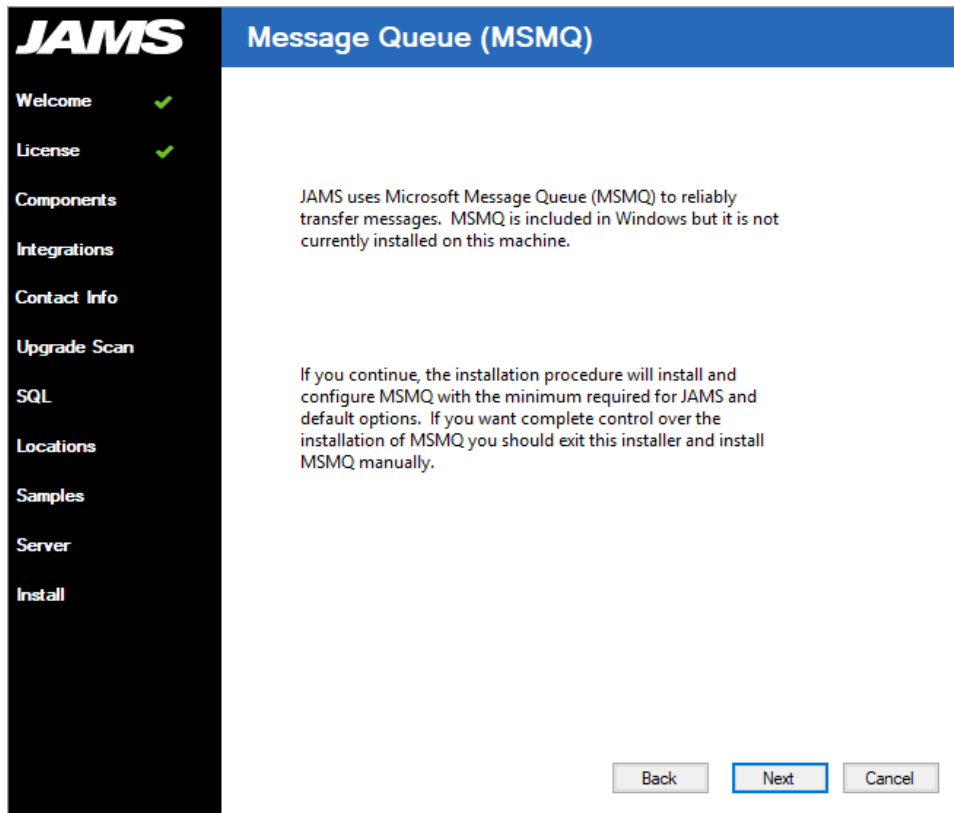
8. On the Core Components page, select the JAMS Components to install. If this is a new installation where no JAMS Scheduler exists, the JAMS Client and JAMS Scheduler should be installed.

9. Click **Next**.

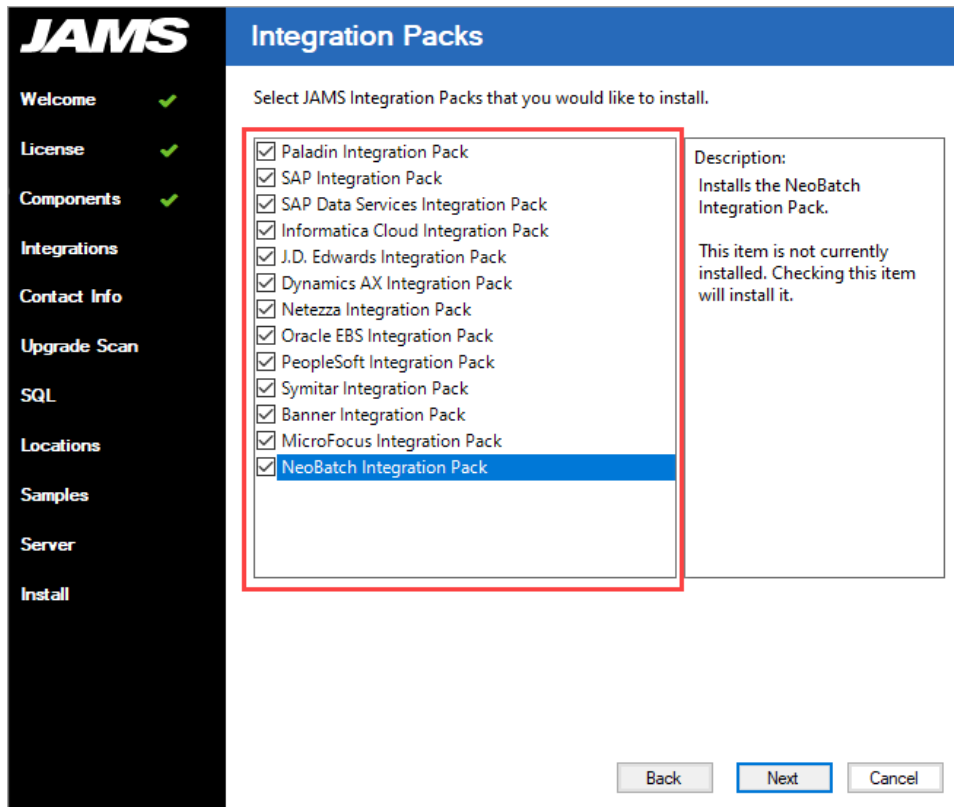


10. On the Message Queue (MSMQ) page, view the message regarding installing MSMQ.

11. Click **Next**.



12. On the Integration Packs page, select the packs to install, if needed. Note that Integration Packs are not required to install JAMS.

13. Click **Next**.

14. On the Contact Information step, enter the information that will be used when working with Technical Support.

Completing this page is optional but entering accurate information can help Technical Support improve its support capabilities. If you are unsure about the information, leave it blank and enter it at a later date.

15. Click **Next**.

NOTE: The installer includes an Upgrade Scan page to review the installation prior to upgrading from a previous version of JAMS. For a new installation of JAMS, the installer skips this step and displays the SQL page.

JAMS

Contact Information

Please supply the contact information below to facilitate communication with JAMS technical support if need be.

Company Name:
Your Company Name Here

Contact Person:
JAMS Admin Name

Phone Number:
(555)555-5555

Contact Person's Email Address:
JAMSAdmin@yourcompanyname.com

Default From Address for Sent Messages:
JAMS@yourcompanyname.com

SMTP Server for Sent Email Messages:
SMTP Server Name Here

Back Next Cancel

16. On the SQL page, select or clear the check box to install SQL Express locally to create the JAMS Database. If another SQL Instance will be used, do not select the check box. (The check box is disabled if SQL Express is already installed.)

NOTE: The JAMS Installer will create the database, unless specified otherwise. The Database should always be created by JAMS. If the Database was not created during the initial installation, you can manually start the JAMS Database Creation Wizard. See [Creating the JAMS Database](#) for more information.

17. Enter the information to create the JAMS Database as well as the authentication method used to connect to it.

NOTE: Windows Authentication uses the Machine Account to connect to SQL Server after the initial install. The user performing the install needs to also have dbcreator rights for the JAMS Database to be created. The rights can be removed after the install is completed and the database is created.

If you select SQL Server Authentication, JAMS connects to SQL with these credentials. The credentials are saved in the Common.config file located at C:\Program Files\MVPSI\JAMS\Scheduler.

NOTE: (The credentials will be visible to anyone that has access to this file.)

18. Click Next.

JAMS

SQL

Cancel the creation of the JAMS database? ☐
 (Note: The JAMS Scheduler Service will not start until this database is created.)

Install SQL Express to create the JAMS database? ☐
 (Deselect this to use a preexisting SQL instance)

Enter the server name or address of your SQL server\instance:

Enter the name for your JAMS database:

SQL Authentication:

Back Next Cancel

NOTE: The JAMS Scheduler Service will not start until the JAMS Database is created.

NOTE: Users should NEVER attempt to manually create the JAMS Database.

19. On the Locations page, accept or change the target directory where JAMS will be installed. The location for the JAMS Database files is a location on the SQL Server. If you change the location, verify the new location exists.

20. Click Next.

JAMS

Locations

Select the target directory where JAMS will be installed:

The full installation directory will be:
C:\Program Files\MVPS\JAMS

Location of JAMS Database Files:
 Primary Database File:

Database Logs:

Back Next Cancel

21. On the JAMS Sample Jobs page, select or clear the check box to install sample jobs.

22. If you select to install the sample jobs, define a JAMS user to run the sample jobs.

NOTE: To run jobs in JAMS, an Execute As user must be specified.

23. Click Next.

JAMS

JAMS Sample Jobs

Install the JAMS sample jobs? ☒

Please enter a Windows username and password for the account that you want to use for the sample jobs.

This information is optional and can be easily changed from the JAMS Client.

Username (include the domain unless it is a local user):
domain\username

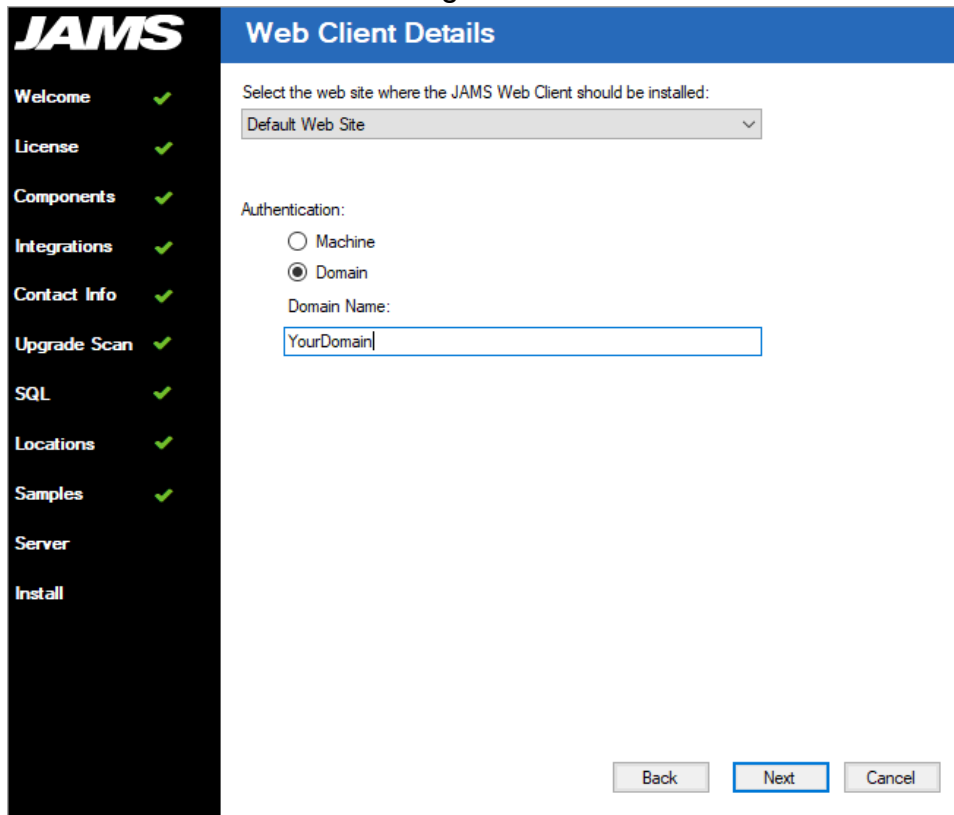
Password:

Verify Password:

Back Next Cancel

24. On the Web Client Details page, select where the JAMS Web Client should be installed. (This page is displayed only if the JAMS Web Client was selected on the Core Components page.)

25. Click **Next**. The installation begins.



The image shows the 'Web Client Details' window of the JAMS installation process. On the left is a dark sidebar with the JAMS logo at the top and a list of installation steps: Welcome, License, Components, Integrations, Contact Info, Upgrade Scan, SQL, Locations, Samples, Server, and Install. The first six steps are marked with green checkmarks. The main area has a blue header 'Web Client Details'. Below it, the text 'Select the web site where the JAMS Web Client should be installed:' is followed by a dropdown menu showing 'Default Web Site'. Under the 'Authentication:' section, there are two radio buttons: 'Machine' and 'Domain', with 'Domain' selected. Below this is a text field for 'Domain Name:' containing 'YourDomain'. At the bottom right are three buttons: 'Back', 'Next' (highlighted with a blue border), and 'Cancel'.

JAMS

Welcome ✓
License ✓
Components ✓
Integrations ✓
Contact Info ✓
Upgrade Scan ✓
SQL ✓
Locations ✓
Samples ✓
Server
Install

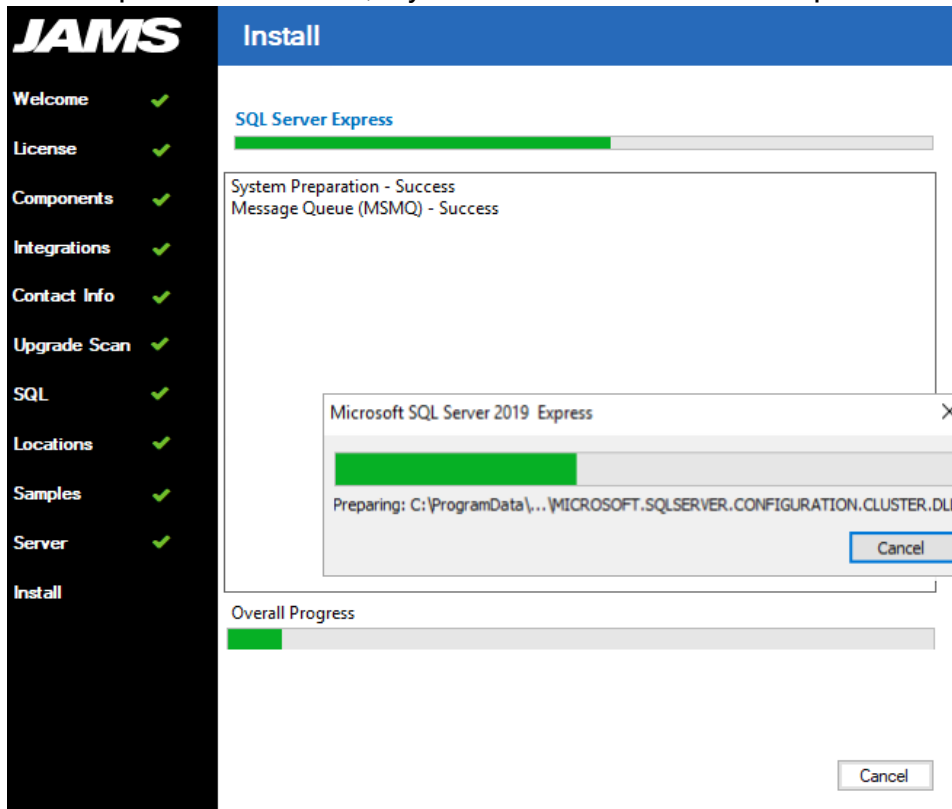
Web Client Details

Select the web site where the JAMS Web Client should be installed:
Default Web Site ▼

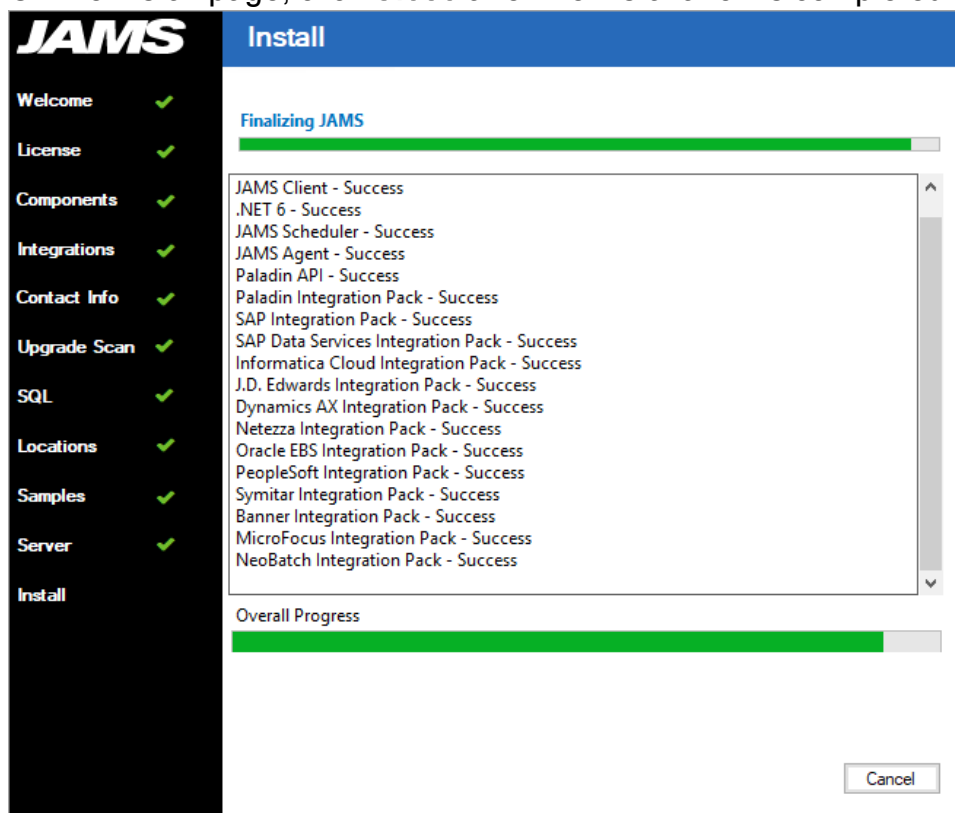
Authentication:
☐ Machine
☒ Domain
Domain Name:
YourDomain

Back Next Cancel

26. View the progress of the installation. You may also see a progress bar displaying the SQL Express installation, if you selected to install it in a previous step.



27. On the Install page, click **Close** after the installation is completed.



Applying the JAMS License

JAMS licenses are sent out as XML files. To apply a JAMS license, copy the license key file (s) into the C:\Program Files\MVPS\JAMS\Scheduler directory. No other actions should be necessary.

If a license is properly placed in the C:\Program Files\MVPS\JAMS\Scheduler> directory and it is not recognized, the license file may be blocked by the File Blocking feature in Windows. To check the status and unblock files:

1. Right-click the license file and select **Properties**.
2. In the Properties dialog, ensure the General tab is selected.
3. Check the lower right corner to see if an **Unblock** button is displayed. If it is displayed, the file is blocked. Otherwise, the file is not blocked.
4. Click **Unblock**.
5. Click **OK** or **Apply** to save the changes.

If the license file is not recognized and is not blocked, restart the JAMS Scheduler Service:

1. Close the JAMS Client.
2. Open Windows Services on the JAMS Server.
3. Restart the JAMS Scheduler Service.
4. Open the JAMS Client. The license should be recognized. If problems persist, contact Technical Support.

For more information about the JAMS license, see [Licenses in JAMS](#).

After Installing JAMS

Here are some additional details you may need to know to continue using or evaluating JAMS.

Access Control

By default, NT AUTHORITY\Authenticated Users have full access to JAMS Objects in new installations. It is recommended to set up JAMS Security as described in [JAMS Security: Setting Access Control Lists](#). However, to work with all JAMS functions, you must be part of the administrators group because administrators, by default, have full access into JAMS.

NOTE: In JAMS V7.0.1367 and later, removing all ACEs on an object behaves the same as in Windows. When all ACEs are removed from an object, only the GrantAdministratorsByPass group have access to the object. Previously, removing all ACEs from an object would give all authenticated users access to that object.

The Server ACL controls who may connect to a specific server. The administrators group retains full access while authenticated users have limited access. When you open the JAMS Client, you will not be in the administrators group. To open the JAMS Client with administrator rights, right-click the **JAMS Client** icon in the Windows Start menu and select **Run as administrator**. This gives you permissions to adjust the ACLs for other users.

Setting an Execute As User

For security purposes, JAMS will not run Jobs without an Execute As user defined. An Execute As user may be defined at the Job or Folder level. To allow all Jobs in the environment to run, users may set an Execute As user on the root folder.

To add the Execute As user:

1. Go to the Properties tab of the Job or Folder definition.
2. Click **Add Property**.

3. Select **Execute As** from the property list.
4. Click **OK**.

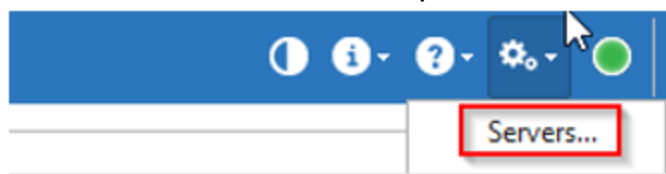
With the Execute As property added, users are able to select a JAMS Credential from the drop-down list. To create a Credential in JAMS, see [JAMS Security: Working with User Credential Definitions](#) for more information.

NOTE: JAMS Properties that are disabled must be added to the object before their value can be defined. To add a Property to a Folder or Job Definition, see [Working with Jobs](#).

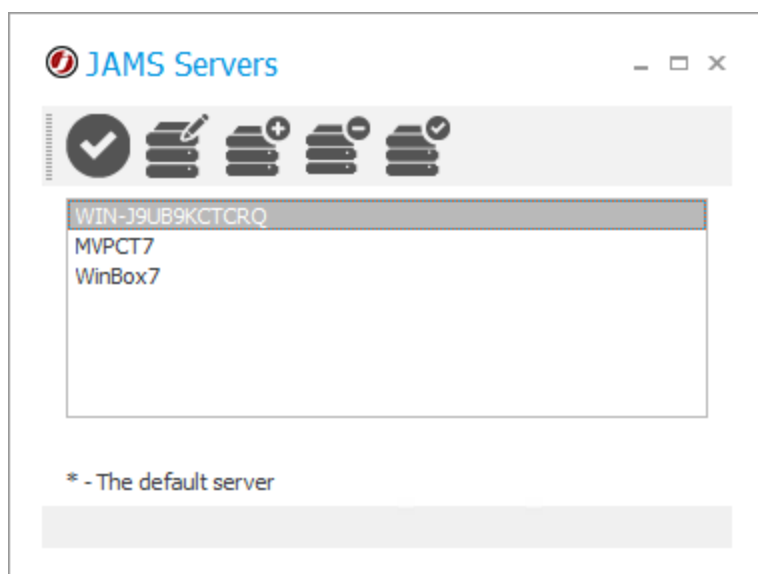
Adding a Server Definition

If you are running the JAMS Client on the same machine as the JAMS Scheduler, the JAMS Client can automatically locate the Scheduler. However, if the Scheduler is running on a different machine, you must include a server definition to tell the JAMS Client where to find the Scheduler.

To add a Server definition, click on **Client Settings** in the upper right corner of the JAMS Client and select **Servers** to open the JAMS Servers dialog.



NOTE: You can have many servers defined within the JAMS Client.



Creating the JAMS Database (If Not Selected During Install)

The JAMS Installer will create the JAMS Database, unless you specify otherwise. The Database should always be created by JAMS. If it was not created during the initial installation, you can manually start the JAMS Database Creation Wizard to create the JAMS Database.

NOTE: Installing JAMS without a database won't affect the JAMS installation, but JAMS cannot function without it. To reinstall the JAMS Database, start the wizard as described in the section below.

Starting the JAMS Database Creation Wizard

To install the JAMS Database after JAMS has been installed, do the following:

1. Go to the JAMS Scheduler default directory (C:\Program Files\MVPSI\JAMS\Scheduler).
2. Right-click **JAMSDBA.exe** and select **Run as administrator**.
3. If the Common.config file exists in C:\Program Files\MVPSI\JAMS\Scheduler, delete it.
4. In the JAMSDBA> prompt, enter **Install** and press **Enter**.
5. Follow the prompts in the wizard to create the database.

Installing the JAMS Web Client/REST API (If Not Selected During Install)

If the JAMS Web Client was not installed during the initial installation of JAMS, you can install it by following the steps below. By default, it will be installed on the JAMS Server, but the JAMS Web Client can be installed on any Windows IIS Server.

Prerequisites

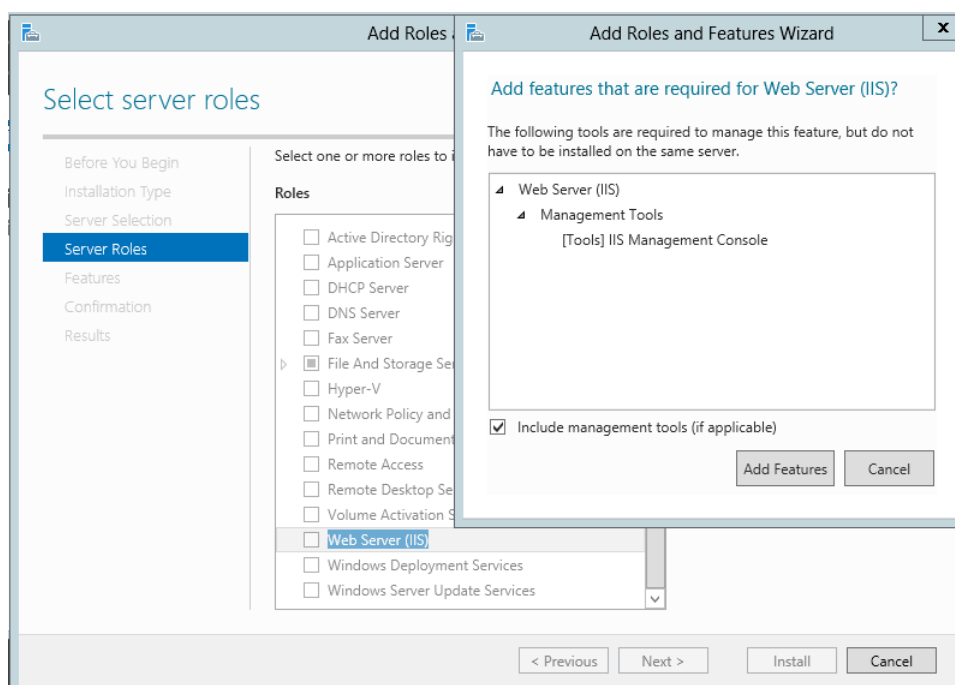
- JAMS V7.0 or higher
- Microsoft Windows Server 2022, 2019, 2016, or 2012R2
- .NET Framework 4.8
- IIS 8 or higher
- Edge, Chrome, or Firefox for the best experience

- If the REST API is installed on a separate server from the Scheduler, the following ports are required to be open on the Scheduler and Web Server:
 - 4773 - Used to communicate from the Web Server to the Scheduler.
 - 1801 - Used by the Web Server to communicate with the jamsrequests and jamsrequestssubmitcancel MSMQ queues on the Scheduler.

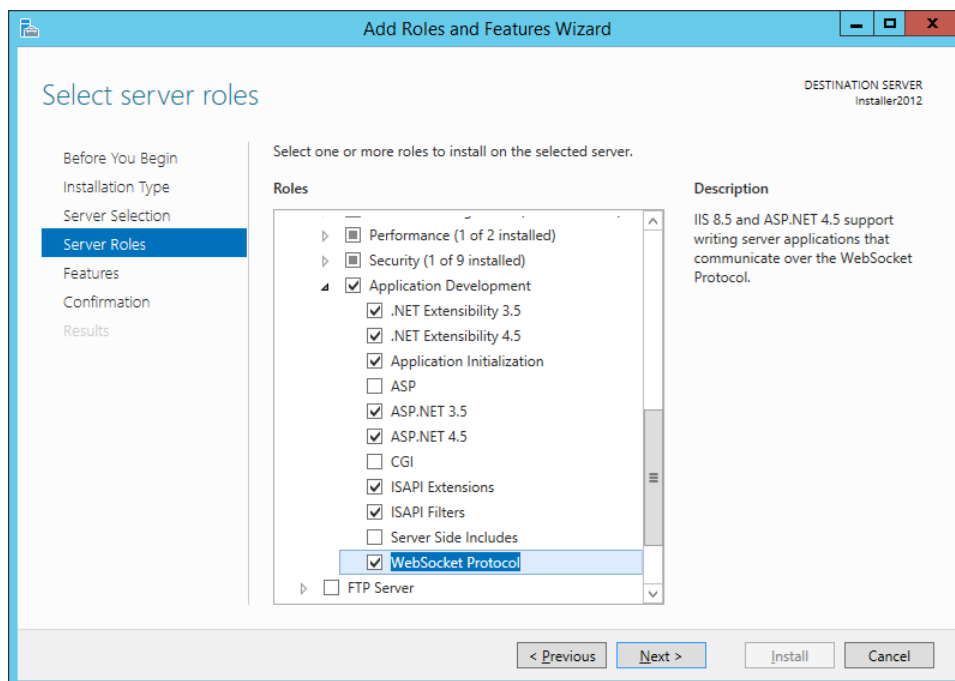
Installing IIS

The installer attempt to enable the roles and features shown below within JAMS if the user performing the install has administrator rights. When installing roles and features, it may be necessary to supply a path to the Windows Source files (SxS). If this is true, manually install the roles and features shown below before installing the JAMS Web Client.

1. Enable the Web Server IIS Role.



2. Ensure the components listed below are enabled.



Installing the JAMS Web Client

1. Run the JAMS installer.
2. Follow the prompts until you see the Core Components page.
3. On the Core Components page, ensure the **JAMS Web Client** component is selected.

IMPORTANT: If you install the JAMS Web Client on a server where the JAMS Scheduler has already been installed, do not clear any Components that are already selected. This will uninstall those Components. If you install the JAMS Web Client on a server where the JAMS Scheduler is not installed, select only the **JAMS Web Client** Component.

4. Click Next.

JAMS

Core Components

Select the JAMS Components that you would like to install.

<input checked="" type="checkbox"/> JAMS Client	<p>Description:</p> <p>Installs the Web Client of JAMS. Also installs the JAMS REST API.</p> <p>This item is not currently installed. Checking this item will install it.</p>
<input type="checkbox"/> JAMS V6 Client	
<input checked="" type="checkbox"/> JAMS Scheduler	
<input checked="" type="checkbox"/> JAMS Agent	
<input checked="" type="checkbox"/> JAMS Agent Interactive	
<input checked="" type="checkbox"/> JAMS Web Client	

Back Next Cancel

5. Follow the prompts until you see the Web Client Details page.
6. On the Web Client Details page, select the web site where the JAMS Web Client should be installed. **Default Web Site** is the default.

7. Click **Next**.

JAMS

Web Client Details

Select the web site where the JAMS Web Client should be installed:

Default Web Site

Authentication:

☐ Machine

☒ Domain

Domain Name:

YourDomain

Back Next Cancel

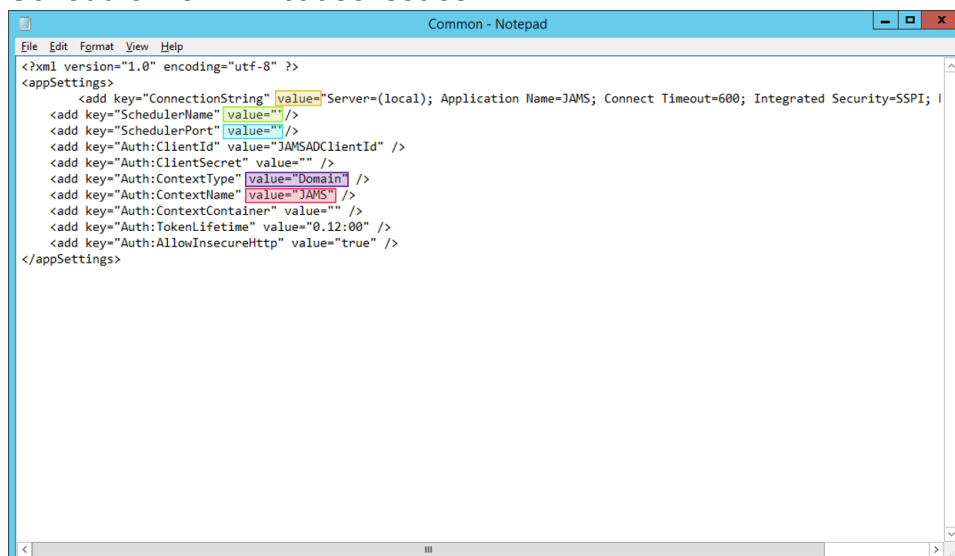
8. Follow the remaining prompts to complete the installation.

After JAMS is installed, open the Common.config file for the JAMS Web Client from C:\Program Files\MVPS\JAMS\WebClient\.

Edit the Common.config file as necessary for the environment.

1. Verify the connection string value matches the desired server.
2. If the JAMS Site is installed within a domain, set the ContextType value to **Domain** and ContextName value to the domain itself. (In the image below, the domain is "JAMS".)
3. If the JAMS Site is not installed within a domain, set the ContextType value to **Machine** and ContextName value to the name of the machine.
4. If the JAMS REST API/Web Client is installed on a different server than the JAMS Scheduler Server, set the SchedulerName value and SchedulerPort value accordingly.
If the Scheduler is not on another server, setting values for SchedulerName and

SchedulerPort will cause issues.



5. If the JAMS REST API/Web Client is installed on a different server than the JAMS Scheduler Server AND a failover is not configured, add the key below to the Common.config file (C:\Program Files\MVPS\JAMS\Scheduler\Common.config) on the JAMS Server. This opens port 4773, allowing the JAMS Web Client to communicate with the Scheduler. In an environment with a Failover configured, port 4773 will already be open. After configuring the JAMS Server Common.config file, restart the JAMS Server service.

```
<add key="EnableRemoteAccess" value="true" />
```

6. If the JAMS REST API/Web Client is installed on a different server than the JAMS Scheduler Server, the JAMS fat client must be installed on the web server. The client can be downloaded [here](#). The Web Server and Scheduler Server must be running the same version of the fat client.
7. If the JAMS REST API/Web Client is installed on a different server than the JAMS Scheduler Server, users must grant the Web Server full permissions to the jamsrequests and jamsrequestssubmitcancel queues on the Scheduler Server. For more information on granting access to the MSMQ using the naming format MyWebServer\$, see [Granting Permissions to the MSMQ Service from a Secondary JAMS Scheduler](#).
8. If the JAMS Web Client is configured for HTTPS (the default is HTTP), the following additional key must be added to the Common.config file, where **JAMSServerName** is the actual name of the JAMS Scheduler.


```
<add key="Auth:TokenUrl" value="https://JAMSServerName/JAMS/">
```

App Pool Settings

The user running the AppPool will be the user running the JAMS Services, which by default is LocalSystems.

NOTE: It may be necessary to give the JAMSSiteAppPool the JAMSApp Role in SQL. Users are recommended to run JAMSSiteAppPool under LocalSystem for local machine installations. For domain installations, it may be necessary to run the JAMSSiteAppPool as NetworkService.

If your JAMS Services are running as a domain account, it is also recommended to run the JAMSSiteAppPool as that same domain account.

Restart IIS services.

Documentation

The REST API includes built-in documentation displaying the endpoints and corresponding Headers that can be passed. There are two variants of Documentation that can be accessed based upon your preference:

<http://localhost/JAMS>

<http://localhost/JAMS/swagger>

Installing JAMS Integration Packs

JAMS natively supports automation using many leading business applications. “Integration Packs” for these preferred applications are selected using a check box during the JAMS installation process. Once an Integration Pack for a product is enabled, you can create, manage, deploy and monitor almost any kind of JAMS Job using the following products:

- SAP Integration Pack
- SAP Data Services Integration Pack
- Informatica Cloud Integration Pack
- J.D. Edwards Integration Pack
- Dynamics AX Integration Pack

- Netezza Integration Pack
- Oracle EBS Integration Pack
- PeopleSoft Integration Pack
- Symitar Integration Pack
- Banner Integration Pack
- MicroFocus Integration Pack
- NeoBatch Integration Pack

When installing JAMS, select the Product Integration Pack you want to include. If you have already installed JAMS without the necessary Integration Pack, re-run the JAMS installer, and select the Integration Packs to install.

NOTE: In JAMS upgrades, clearing the component check boxes will uninstall those components.

The JAMSDBA Utility

The JAMSDBA.exe utility is used to manage the JAMS Database and to perform other installation and management tasks. JAMSDBA is a command line utility located in the Scheduler installation directory (C:\Program Files\MVPSI\JAMS\Scheduler by default).

NOTE: The JAMSDBA.exe utility must be run as an Administrator. (Right-click the JAMSDBA.exe and select **Run as Administrator**.)

When starting the JAMSDBA, you will see a JAMSDBA> prompt. You can enter the command you want to execute or enter **HELP** to access online help. You can also start JAMSDBA with a command appending the command you want to execute; for example:

```
JAMSDBA INSTALL/DBLOG
```

Microsoft Messaging Queueing (MSMQ)

JAMS uses Microsoft Message Queue (MSMQ) to reliably pass messages between the JAMS Services. It does this by creating private queues named jamsrequests and jamsrequestssubmitcancel. MSMQ is included with Windows, but it is not installed by default. If MSMQ is not installed before installing JAMS, the JAMS installer will install MSMQ using the default minimum settings.

NOTE: If you want to control how MSMQ is installed, it is best to install MSMQ before installing JAMS.

To change the way that MSMQ is installed after installing JAMS:

1. Stop the JAMS Services.
2. Reinstall MSMQ.
3. Restart the JAMS Services.

Configuration Settings

JAMS contains a number of configuration settings that are listed below. You can change these settings using the Configuration shortcut on the JAMS Menu.

Contact

Contact	Description
Company Name	The company name to include in error reports.
Contact Name	The person in your organization responsible for JAMS deployment.
Contact Phone	The responsible person's phone number.
Contact Email	The responsible person's email address.
Automatically Report Exceptions	When set to true, failures of the JAMSServices will be automatically sent to JAMS technical support.
E-Mail	Description
Default Notify Email Address	A comma separated list of e-mail addresses that will receive all notifications. You can remove inherited addresses by prefixing the address with a minus (-) sign. You can remove all inherited addresses by entering -*.
From Address	The from email address used when sending the e-mail from JAMS.

Maximum Log Size	Log files that are larger than this value will not be attached to notification e-mails.
SMTPServer	The name of the SMTP server used for JAMS. This must be configured to send error reports to JAMS Support.
SMTP Retry Count	The retry count used to resend SMTP emails if an attempt to send one is unsuccessful.
SMTP Retry Delay	The time to wait to resend SMTP emails if an attempt to send one is unsuccessful.

System

ACL Security Bypass	Description
Grant Administrators Bypass	Option to allow individuals in the Administrator group to bypass ACL security.
Grant Bypass Group	Individuals listed in a specified group to bypass ACL security.
ACL Import/Export	Description
Restrict ACL on Export	<p>Select the check box to prevent ACL information from being included in the XML when exporting any object, such as a Job or an Agent, from JAMS. To restrict the ACL for a specific object, use the Export-JAMSXML PowerShell cmdlet with -IgnoreACL. When selected, the Restrict ACL On Export option overrides the cmdlet.</p> <div> <p>NOTE: Ensure the JAMS Client and JAMS Scheduler have been updated to the same version to use this option.</p> </div>

Restrict ACL on Import	Select the check box to prevent ACL information from being included in the XML when importing any object, such as a Job or an Agent, into JAMS. Inherited ACL information from within JAMS will still be applied. To restrict the ACL for a specific object, use the Import-JAMXML PowerShell cmdlet with -IgnoreACL. When selected, the Restrict ACL On Import option overrides the cmdlet.
Options	Description
Schedule Examples	Examples of natural language date specifications displayed in date editors.
Base URL	The base URL for this JAMS Scheduler web site.
Encoding	Description
Default input Encoding	The default input encoding for routine jobs.
Default Output Encoding	The default output encoding for routine jobs.
File Locations	Description
Default Log Location*	The default location for Job log files.
Common Log Location	Used in a failover configuration to define a location that is common to both servers.
Temporary File Location	The location for temporary files.
History	Description
History Lookback Period	The number of days in the past to query history inside a detail dialog.
History Query Limit	Maximum number of history records to return in a single query? (zero means unlimited)
History Query Time Limit	The maximum time to wait (in seconds) for a response to a history query.
Projected Schedule	Description
Projected Schedule Limit	The maximum number of jobs to return in a projected schedule.

Projected Schedule Time Limit	The maximum time to wait (in seconds) for a response to a projected schedule.
Monitor	Description
Keep Completed	Keep completed Jobs in the Monitor for a specified period of time.
Workdays	Description
Work Monday	Is Monday usually a workday?
Work Tuesday	Is Tuesday usually a workday?
Work Wednesday	Is Wednesday usually a workday?
Work Thursday	Is Thursday usually a workday?
Work Friday	Is Friday usually a workday?
Work Saturday	Is Saturday usually a workday?
Work Sunday	Is Sunday usually a workday?

Presentation

Theme	Description
Light Theme	The theme shown to end-users when they select Light Theme.
Dark Theme	The theme shown to end-users when they select the Dark Theme.
Documentation	Description
Header	This HTML fragment is placed before all documentation content configured within this server.
Footer	This HTML fragment is placed after all documentation content configured within this server.
E-Mail	Description
E-Mail Header	This HTML fragment is placed before all e-mail content configured within this server.
E-Mail Footer	This HTML fragment is placed after all e-mail content configured within this server.

*The DefaultLogLocation is a .NET format string used to construct the full file specification for a Job's log file. The data values passed in the formatting operation are:

0. Directory specification
1. Filename
2. File Extension
3. Folder Name
4. Job or Sequence Name
5. Run occurrence number
6. JAMS Entry Number
7. Timestamp date & time
8. Timestamp year
9. Timestamp month
10. Timestamp day
11. Timestamp hour
12. Timestamp minute
13. Timestamp second
14. Timestamp millisecond

Working with JAMS Services

JAMS Scheduler installation includes the following Windows Services:

- JAMS Scheduler Services
- JAMS Server Service
- JAMS Executor Service
- JAMS Agent Service

NOTE: Installing JAMS Agents on Windows also includes the JAMS Agent Windows Service.

JAMS Scheduler Service

The JAMS Scheduler Service is responsible for automatically scheduling jobs, firing triggers, and checking dependencies. The JAMS Scheduler must include access to the JAMS Database or the service will fail. However, JAMS is designed to be resilient. Job execution is handled by the JAMS Executor service, so if the JAMS Scheduler service fails, no job execution information is lost.

JAMS Server Service

The JAMS Server provides middle-tier services to all JAMS Client components (GUI, PowerShell, .NET Class Library, and Web services). While the JAMS Server service is not involved in the execution of Jobs, many jobs can use the JAMS PowerShell client, which does utilize this service.

JAMS Executor and Agent Services

The JAMS Executor is responsible for executing and monitoring Jobs. This service does not access the JAMS Database.

The JAMS Agent service is an extension of the JAMS Executor service. When the JAMS Executor needs to execute a Job on a different machine, it does so using the JAMS Agent running on that machine.

Troubleshooting JAMS Services

Each JAMS Service generates a ServiceName.log, such as JAMSScheduler.log, in the installation directory. These logs are reset every Sunday with the previous weeks log files renamed to ServiceNameArchive.log, such as JAMSSchedulerArchive.log. The default location is C:\Program Files\MVPSI\JAMS\Scheduler.

NOTE: JAMS Services writes any serious errors to the Windows Event log. When troubleshooting JAMS, always check the event log and the .log files.

If you suspect there are problems with JAMS, shutting down one of the three services may resolve your issues. The list below provides some guidelines from the least to most disruptive option.

Restart the JAMS Scheduler

While the JAMS Scheduler Service does the most work, restarting it is the least disruptive. Restarting the JAMS Scheduler Service does not cause any Jobs failures and all job completion information remains intact. While the JAMS Scheduler service is stopped, no new Jobs can enter into the Schedule.

Stop the JAMS Server Service

The JAMS Server service may be stopped without losing any job execution information. The JAMS Client will not function while the JAMS Server service is shut down.

Shut down the JAMS Executor Service

Stopping the JAMS Executor service should be a last resort, as it will not typically resolve JAMS issues. This service executes and monitors Jobs. When this service is halted, completion information for any executing Jobs will be lost and some executing Jobs may fail. The JAMS Executor service does not access the JAMS database, so stopping this service won't be necessary during SQL service maintenance.

Service Accounts

By default, JAMS Services are set to run under the LocalSystem account, but this can be modified to run on a Windows Domain based account. See [Change the Account Running JAMS](#) for more information.

In general, it is recommended that you leave the JAMS Executor and JAMS Agent services running under LocalSystem. These services require access to the database or network and require the privileges associated with the LocalSystem account.

Use the Service Control application to change the account for the JAMS Scheduler and JAMS Server services in order to control network and database access.

When modifying the account, you may need to adjust the security settings on:

- C:\Program Files\MVPSI\JAMS\Scheduler folder
- C:\Program Files\MVPSI\JAMS\Scheduler\JAMSScheduler.log
- MSMQ jamsrequests and jamsrequestssubmitcancel private queues
- JAMS Database

You need to modify the security on the MSMQ private queues to grant the domain account full access to the queue. This may require you to take ownership of the MSMQ queue.

The following local security policies should also be granted for the domain-based account:

- Log on as a Batch job
- Log on as a Service
- Adjust memory quotas for a process
- Bypass traverse checking
- Replace a process level token

If the domain-based user account is not in the administrators group, create an Active Directory Group, add the user to the group, and make the following changes in the Common.config file located at: Program Files\MVPSI\JAMS\Scheduler<directory>

```
<add key=AuthorizedGroup " value="domain\YourGroup"/>
```

Managing the JAMS Database

The JAMS Database provides critical features to the JAMS Scheduler and stores JAMS definitions. The following section contains information on managing the JAMS SQL database back end.

Manually Creating the JAMS Database

Typically, the JAMS Database is created with the JAMS Scheduler during the installation process. However, in some cases, when installing the JAMS Scheduler you may prefer not to create the JAMS Database. See Creating the JAMS Database for more information.

NOTE: The JAMS Database should always be created using the JAMS Installer or JAMSDBA utility. Manually creating the JAMS Database outside of these tools will cause various issues in JAMS.

Security

JAMS supports either Windows Integrated Authentication or SQL Server Authentication. This can be modified after installation by changing the connection string found in the Common.config file. The default installation requires the JAMS Services to be running under the LocalSystem account. To facilitate this, the installation executes the following SQL commands:

```
exec sp_grantlogin @loginame='BUILTIN\Administrators'
```

```
exec sp_grantdbaccess @loginame='BUILTIN\Administrators', @name_in_db='JAMSService'  
exec sp_addrolemember @rolename='JAMSApp', @membername='JAMSService'
```

The effect of using Windows Integrated authentication is that anyone in the administrators group can map the JAMSService Database user and become a part of the JAMSApp Database role.

NOTE: You can modify the security to fit your needs, but the JAMS Services must be included in the JAMSApp Database role.

Backing up the JAMS Database

It is critical to back up the JAMS Database. Since the JAMS Database is a standard SQL Server database, simply add it to your existing SQL Server backup procedures. You can also use the JAMS Job named JAMSBackup, located in the JAMS folder, to back up the JAMS Database.

Choosing the Recovery Model

Before implementing a backup plan, select the recovery model to use with the JAMS database. The default installation uses the Simple Recovery Model. The other option is the Full Recovery Model. The main difference between the two is in the amount of journaling performed by the database.

Simple Recovery Model

The Simple Recovery Model is the easiest to use. You periodically backup the JAMS database only, not the journal file. The downside to this model is a lack of recovery options. If the JAMS Database becomes corrupted, you can recover it from a backup but all database changes since the last backup will be lost. This includes job execution history, so Job dependencies may not work as expected.

Full Recovery Model

This model contains more robust recovery features. However, the disadvantage of this model is that you must manage the journal file for the JAMS Database. Each time a change is made to the JAMS Database it is also written to the journal file.

For example, if a hardware failure occurs or you have otherwise corrupted the JAMS Database, you can restore a backup and then recover it using the journal files up to a specific point in time.

Restoring the JAMS Database from a Backup

Restoring a JAMS Database from a backup requires a standard SQL Server restore operation, but there are some details to consider before attempting the restoration process.

The JAMS Database includes the current schedule. Often, you may not want the current schedule to be restored. For example, if the database was backed up on Monday and you restore it on Friday, you probably don't want to start running Monday's Jobs along with the rest of the week's processing.

You can use JAMSDBA.exe to reset the Schedule.

```
CD C:\Program Files\MVPSI\JAMS\Scheduler " JAMSDBA RESET SCHEDULE
CD C:\Program Files\MVPSI\JAMS\Scheduler" JAMSDBA UPDATE/LOG
```

Moving or Restoring the JAMS SQL Database

The process for moving the SQL database used by JAMS does not require reinstalling JAMS.

1. Ensure there are no Jobs currently running or are scheduled to run in the near future. You can manually put all Jobs on hold by using the Monitor view or by using a Maintenance Resource as described in the [Creating a Maintenance Window](#) article. When restoring the Database and activating the JAMS Server, you may not want jobs to run at that time.
2. Use the JAMSBakup Job in the JAMS folder to back up the SQL database and restore to the target server.
3. Edit the connection string property of the Common.config file located at C:\Program Files\MVPSI\JAMS\Scheduler\ by default.
4. Update the server location and the database name, if the name of the database is being changed.
5. Update the permissions to the JAMS Scheduler Server to connect to the restored JAMS Database.
6. Replace **YourDomain\YourMachineName** with your own name in both lines of the query:

```
exec_sp_grantlogin @loginame='YourDomain\YourMachineName$'
exec_spgrantdbaccess @loginame='YourDomain\YourMachineName$', @name_in_
db='JAMSMachine2'
exec_sp_addrolemember @rolename='JAMSApp', @membername='JAMSMachine2'
```

NOTE: If you have a failover server, change the **JAMSMachine2** to **JAMSMachine1** in the query and run the query again. You need to truncate the Failover table of the JAMS Database regardless of the failover configuration.

7. Restart the JAMS Scheduler Service from the Windows Control Panel.
8. If you are running JAMS V6.4X or older, run JAMS_DB_ERRORS.sql, located in the JAMS Scheduler directory, on the master database within the same SQL instance where the JAMS Database resides.

NOTE: Moving a SQL/JAMS database does not require reinstalling JAMS.

The Common.config file contains the connection string used by JAMS services to link to the SQL database. Below are two examples of connection strings: one for Windows Authentication and another for SQL Authentication.

Windows Authentication

```
<?xml version="1.0" encoding="utf-8" ?>
<appSettings>
<add key="ConnectionString" value="Server=SQLA\INST1; Failover Partner=SQLB\INST1;
Database=JAMS; Application
Name=JAMS; Connect Timeout=600; Integrated Security=SSPI"/>
</appSettings>
```

SQL Account Authentication

```
<?xml version="1.0" encoding="utf-8" ?>
```

```
<appSettings>
<add key="ConnectionString" value="Server=SQLA\INST1; Failover Partner=SQLB\INST1;
Database=JAMS; Application
Name=JAMS; Connect Timeout=600;Trusted_
Connection=False;uid=YOURSQLACCOUNT;pwd=YOURPASSWORD" />
</appSettings>
```

Settings Path for the JAMS Client

JAMS can override all saved client settings at any installed locations. This can be useful when you want multiple JAMS Clients to default to the same shortcuts and themes. It also simplifies how new layouts and shortcuts changes can be standardized across-the-network with only one location needed as a template.

Configure the **SettingsPath** for each JAMS Client by modifying the user.config file as described below:

1. Open the configuration file that is by default located at: C:\Program Files\MVPS\JAMS\Client\User.config
2. Add a new entry within the <appSettings> tags called "CommonSettingsPath" with a value of the location where you want to save the client settings. For example:

```
<appSettings file="User.config" >
<add key="ClientSettingsProvider.ServiceUri" value="" />
<add key="CommonSettingsPath" value="\\AppServer\ClientSettings\" />
</appSettings>
```









3. Optionally, if you want the JAMS Client to save changes back to this common location simply add an entry for **WriteToCommonSettings** with a value of "true". By default, any new changes will be saved to the user-specific location.

```
add key="WriteToCommonSettings" value="true" />
```

- Copy the desired client settings into the new common location, including the same X.X.X version number subfolder as found in the user's folder. By default, JAMS stores the client settings by version number for each user and stores it in the .XML files found in:

```
%LocalAppData%\HelpSystems\JAMS\X.X.X.
```

You can replicate settings from one client to the next by copying these flat files.

Name	Size	Type
 AgentsTree16F3AF1085.xml	26 KB	XML Document
 ConfigList0E26F7EE68.xml	10 KB	XML Document
 ConfigurationViews00DB75825F.xml	4 KB	XML Document
 ConfigurationViews00DCD09FD8.xml	4 KB	XML Document
 ConfigurationViews02DB75A970.xml	4 KB	XML Document
 ConfigurationViews02108EC4D0.xml	4 KB	XML Document
 ControlViews004D8300D4.xml	4 KB	XML Document
 ControlViews024D8300D4.xml	4 KB	XML Document

- Inside your Common Settings folder, a subfolder will be created with the version number of the Client in the same manner as the user's folder. For example, if the Common Settings folder is defined as \\AppServer\ClientSettings and your Client runs version 7.5.1, create the \ClientSettings\7.5.1\ folder.
- Two critical files in this group include: Servers.xml and JAMSPage.xml. The former stores the JAMS Server definitions and the latter defines shortcut settings on the JAMS page. In most cases, these are the only files that are required, with other files controlling more user specific setting such as columns and column widths.
- Or you can copy the entire folder to the new location. These files contain the current client settings, but make sure you select the folder that corresponds to the current installed version of JAMS.
- JAMS will attempt to load files from the common location that it does not find in the user-specific location. To force an existing JAMS Client to load from the common location, you need to delete the user- specific settings.

These user settings can be helpful if you want to have varying client layouts for different user groups such as administrators, developers, and users. For example, you could set up three directories that contain the appropriate settings for each group. After configuring the Client

config file for the members of these groups, you could then manage the settings independently.

- If the **WriteToCommonSettings** option is “true” for multiple JAMS Clients, they can overwrite the common settings.
- JAMS will only load from the common location if it doesn’t find the files in the user-specific location.
- You can prevent users from making changes to the server definitions by adding:

```
<add Key="LockServer" value="true" />
```

- Prevent users from changing shortcuts by adding:

```
<add Key="LockShortcutbar" value="true" />
```

- When Configuration files exist in the Common location and the user's local settings, ensure the Client loads Configuration files from the Common location by adding:

```
<add Key="PrioritizeCommonSettings" value="true" />
```

The following is a sample of a Configuration file using **PrioritizeCommonSettings**:

```
<?xml version="1.0" encoding="utf-8" ?>
<appSettings>
  <add key="PrioritizeCommonSettings"
    value="True" />
  <add key="CommonSettingsPath"
    value="\\az-win019-test\share\ClientSettings" />
  <add key="LockShortcutBar"
    value="False" />
  <add key="LockServer"
    value="False" />
  <add key="LockQuery"
    value="False" />
  <add key="LockUsername"
    value="False" />
```



```
<add key="PromptForSearchKeys"
      value="False" />
<add key="DefaultServer"
      value="" />
<add key="DefaultPort"
      value="773" />
</appSettings>
```

High Availability

The Benefits of a JAMS Failover Scheduler

In many instances, it is beneficial to ensure a secondary failover environment has been configured for the JAMS installation. The failover setup ensures there is a completely redundant instance of JAMS residing on a secondary server, often offsite, relying on a heartbeat connection between the two.

By default, this heartbeat is set to a sixty-second interval. If the secondary JAMS Scheduler server does not get a response from the primary JAMS Scheduler server within 3 consecutive beats, the secondary will take over as the primary JAMS Scheduler to ensure the schedule continues as expected.

JAMS Failover Architecture

The JAMS Failover Architecture should consist of at least three servers:

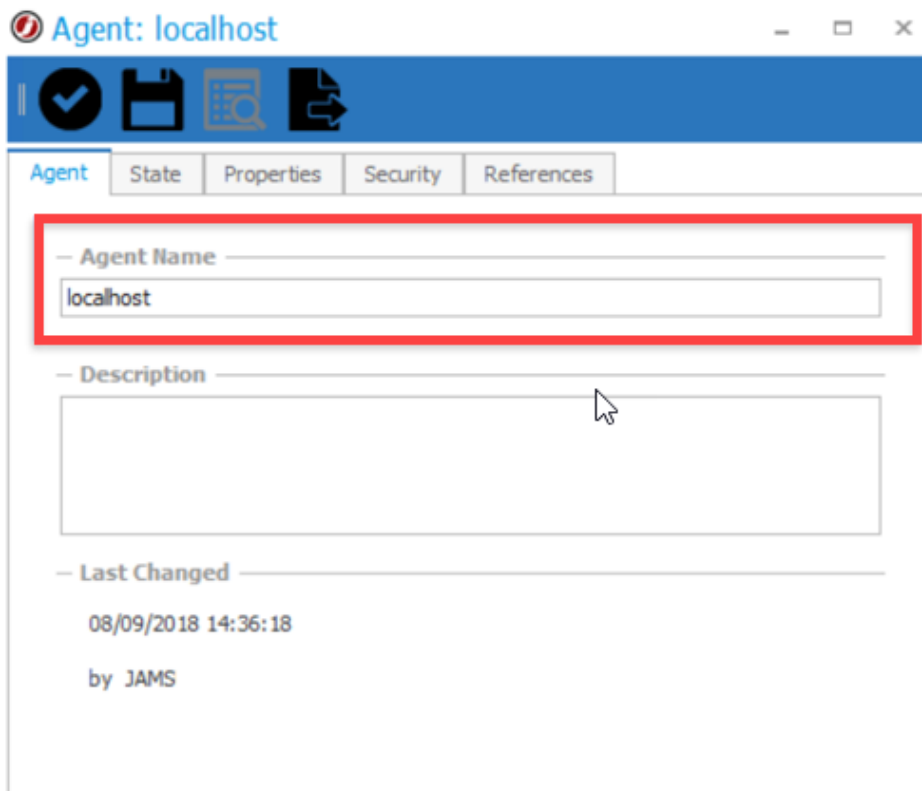
- JAMS Primary Scheduler Server
- JAMS Failover Scheduler Server
- JAMS Agent Server(s), where Jobs are run

If a failure occurs, the secondary JAMS Scheduler will take over for the primary JAMS Scheduler to ensure the schedule continues as expected. If Jobs are executing locally on the primary JAMS Scheduler server, a failure will cause all Jobs executing on the JAMS Scheduler server to fail. Running Jobs on JAMS Agent server(s) protects those Jobs from a primary JAMS Scheduler server failure.

Installing and Configuring the JAMS Failover

NOTE: An additional license needs to be purchased to install a Failover Server. Contact your Account Manager for more details.

1. Install the JAMS Primary Engine.
2. The Local Agent Name defaults to the name of the Primary Scheduler. Update the name of the Secondary Scheduler to localhost.



3. For redundancy of the JAMS Database, you can configure AlwaysOn Availability after you complete these steps. AlwaysOn Availability is a Microsoft project that replaced Database Mirroring in SQL Server 2012.
4. Stop the **JAMS Scheduler** service on the JAMS Scheduler Server.
5. If the primary and secondary JAMS Schedulers are to share a database, add a user to the JAMS database to allow the secondary JAMS Scheduler to connect. This can be done by running the SQL statements below on the JAMS database. Replace the login name with the machine name.

```
exec sp_grantlogin @loginame='YourDomain\YourMachineName$'
exec sp_grantdbaccess @loginame='YourDomain\YourMachineName$', @name_in_
db='JAMSMachine2'
exec sp_addrolemember @rolename='JAMSApp', @membername='JAMSMachine2'
```

6. Install the JAMS Scheduler on the secondary server.
 1. When prompted for the database server, enter the same SQL Server, Instance, and Database Name as the primary JAMS Scheduler server.
 2. Click Next. On the dialog stating "Database already exists", select Use Database.
7. Stop the JAMS Scheduler service on the secondary server.
8. Create and edit a Failover.config file in the C:\Program Files\MVPSI\JAMS\Scheduler\directory on the primary and secondary servers. A sample Failover file is shown below.
9. Start the JAMS Scheduler on both the primary and secondary servers.
10. Grant permissions to the MSMQ Service from the secondary JAMS Scheduler as described in [How to Grant Permissions to the MSMQ Service from a Secondary JAMS Scheduler](#).

A Sample Failover config

```
<FailoverConfig>
<Primary>Server1</Primary>
<Secondary>Server2</Secondary>
<Port>4773</Port>
<Interval>60</Interval>
</FailoverConfig>
```

- The primary and secondary server names are the hostnames of the respective machines.
- The port is the port used by the primary and secondary scheduler to communicate the heartbeat connection.
- The interval value is the length of time that must elapse before a heartbeat is missed. If 3 heartbeats are missed, then the failover process will start.
- The interval value can be modified to suit the users needs. It is recommended to not have this value lower than 15 seconds to ensure that the schedulers are not constantly trying to take control of the database during intermittent network issues.

Common Log Output Location

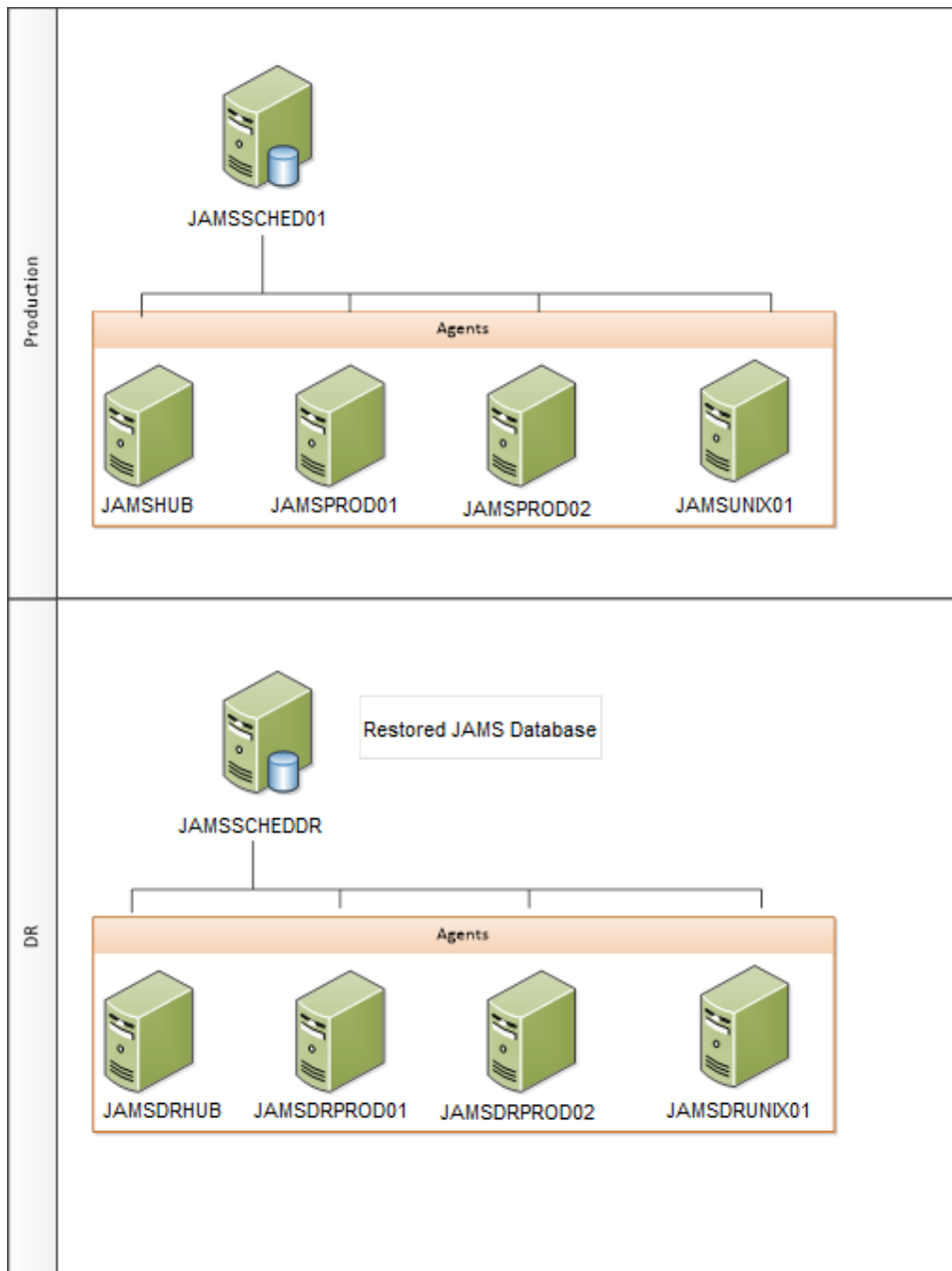
To have a common log output location on network share for both Primary and Secondary schedulers, see [Setting a common log output location for JAMS in a Failover Environment](#) for more information.

NOTE: After you configure the location for the common log, ensure the default log location exists on the Primary and Secondary schedulers.

Disaster Recovery Configuration

Recommended DR Model

The JAMS Server and Agents should have DR equivalents, named differently than the production servers. This will ensure the boxes are not seen on the network with the same name as their production equivalents.



With the JAMS DR architecture set up as outlined above, the DR Process includes:

- Restoring the Production JAMS database to the DR Server and running any necessary database SQL scripts
- Replacing references to the JAMS Production nodes with DR equivalents

Disaster Recovery Process

Before Beginning the DR Process

On the DR Server, ensure the following are true:

- The JAMS Services are disabled (Includes the JAMS Executor, JAMS Scheduler, and JAMS Server)
- Host files on the JAMS Server and Agent Nodes are pointing to the DR Servers
- Common.config is pointing to the local SQL Server (located at C:\Program Files\MVPSI\JAMS\Scheduler\Common.config by default)

Restore the SQL Database

Create a backup of the JAMS Production Database and then restore the JAMS Production Database in DR.

Truncate the Failover Table

Truncate the failover table on the DR Server.

NOTE: The failover table details the GUID of the JAMS Server that is running against this database. This GUID should match the Installed GUID found in the Server Configuration file (located at C:\Program Files\MVPSI\JAMS\Scheduler\Server.config by default).

Back up the Current Monitor View state

On the JAMS DR Server:

- Start the JAMS Server Service.
- Within the JAMS DR Client, go to the **Monitor** View and take screenshots of the current Jobs within the Monitor View.
- Stop the JAMS Server service.

NOTE: The Failover table on the DR Server must be truncated before the JAMS Server is started. The JAMS Server will then automatically add the required entries to the table.

Update the JAMS Server Agents/Nodes to Point to the DR Servers

The JAMS Nodes must be updated to point to the DR Servers instead of Production Servers. In this example, all Sequences and Jobs within JAMS refer to a JAMS Queue, rather than point to an Agent Node directly. With this Queue configuration, re-directing the Nodes will be a simple case of updating the Production Queues to point to the DR Nodes.

JAMS will not allow existing nodes to be deleted if any running or pending Jobs that point to those Queues/Nodes are within the Schedule/Monitor view. Therefore, in order to allow the nodes to be updated, all Jobs will need to be cleared from the Monitor View using the Reset Schedule process detailed below.

Reset the JAMS Schedule

On the JAMS DR Server

- Ensure the JAMS Scheduler Service is stopped. The JAMS Scheduler service should be stopped so the Reset Schedule command (explained later) does not re-schedule Jobs before the queue nodes have been updated.
- Run the JAMSDBA executable as an administrator.
- Enter **RESET SCHEDULE** and press **Enter**.

When JAMS starts up against a restored database, it checks to see if the Schedule is more than 48 hours old. If it is, it automatically rebuilds the schedule from the current time. Otherwise, it will continue with the current schedule as-is. As the JAMS Database will likely be less than 48 hours old on DR, the 'Reset Schedule' process will have to be manually run using the JAMSDBA utility.

Once the Scheduler Service starts, the Schedule will be built based on the current time. To rebuild the Schedule against a different time, the /Restart switch can be specified.

Enable Access to the JAMS Client

Start the JAMS Server Service on the DR Server. To successfully connect to the Server from a JAMS Client, the JAMS Server service needs to be enabled and started. View the status of the JAMS Server at the bottom left corner of the JAMS client.

Update the JAMS Server Nodes

- Ensure the JAMS Client is opening against the DR Server and that all Jobs have been cleared from the Monitor View.
- Run the **Update Queues** PowerShell script to update the JAMS Queues to point to DR Nodes.

NOTE: Because the JAMS Scheduler is still disabled, run the Update Queues script outside JAMS.

- Ensure the Queues have been successfully re-pointed in the front end by viewing the Agents and Queue Nodes.

Start the JAMS Services

On the DR Server, manually start the JAMS Scheduler Service and the JAMS Executor Service. Starting the JAMS Scheduler results in the Schedule being rebuilt within the Monitor view, as defined by the JAMSDBA utility.

NOTE: Sequences or Jobs that were scheduled before DR that were still pending when production went offline may need manual attention.

Run Test Processes

On the DR Server, run a test process in JAMS to validate that the Scheduler is working as expected.

Licensing DR

The JAMS DR Server requires a unique JAMS License.

Agents will be consumed as follows:

The JAMS License is on the JAMS Server and it will detail the number of available Agent licenses. When the JAMS Server is restarted, the number of allocated Agent licenses is set to zero. As Jobs consume Agents, the allocated License count is incremented.

Agents in JAMS

There are many different Agents available for JAMS, and each one is designed and configured for a specific platform and purpose.

Most JAMS Agents have installers on the JAMS Server along with the JAMS Scheduler (C:\Program Files\MVPSI\JAMS\Scheduler\Agents). The JAMS Client can be used to deploy and upgrade Agents, after those Agents have been defined.

NOTE: For more information on working with specific JAMS Agents, refer to the Agent topics.

Agents in JAMS are initially defined using four fundamental properties:

- **Name** - The reference name for the Agent definition in JAMS.
- **Description** - An optional description of the defined JAMS Agent.
- **Type of Agent** - The type of Agent to be created, based on the Agent's intended use. The type defines the protocol and direction of the communication between the Agent and the JAMS Scheduler. The default Agent Type is Outgoing. JAMS has the following types of Agents:
 - **Local** - This Agent type is used for the JAMS Scheduler machine because the Scheduler acts as its own Agent.
 - **Incoming** - This Agent type is used the JAMS Agent Interactive. The Agent initiates communication with the JAMS Scheduler. To stop an Agent from connecting, the connection needs to be disabled at the Agent.
 - **Outgoing** - This Agent type is the most common type and is used for most Agents. The JAMS Scheduler initiates communication with the Agent. This is used for Windows and AgentD. Windows Agents can maintain the connection between Jobs and run multiple Jobs simultaneously without a new connection.
 - **SSH** - This Agent type can connect to a remote system through the SSH protocol. You can enter one command at a time just like you were if you were typing at a terminal.
 - **SSHAgentX** - This Agent type is used only for when the remote machine has the SSHAgentX Agent installed.
 - **Passive** - This Agent type is mainly used for the z/OS integration. For Passive Agents, the Execution Method handles the communication.
- **Type of Platform** - The operating system platform for the Agent. Options include Windows, Linux, UNIX, OpenVMS, and more.

Agent Definitions screen

The Agent Definitions screen lists all Agents that have been defined. The following information is displayed:

- **Agent Name** - The name assigned to the Agent. The name of a JAMS Agent is a friendly name for the Agent in JAMS. The JAMS name of an Agent is not required to match the system name of the Agent machine.
- **Description** - An optional description of the Agent.
- **Platform** - The operating system platform for the Agent. Options include Windows, Linux, UNIX, OpenVMS, and more.
- **Agent Type** - The type of Agent based on its intended use.
- **Enabled** - An Agent can run Jobs when it is enabled. If it is disabled, the Agent will not run Jobs.
- **Agent State** - The current state of the Agent.
- **Job Count** - The number of Jobs running on the Agent.
- **Job Limit** - The maximum number of Jobs that can run at one time on the Agent. You can adjust this to optimize performance.
- **License Allocated** - A read-only field that indicates if JAMS has allocated a license to the Agent. The license is allocated the first time a Job is submitted to the Agent, if a license is available.
- **Port** - The network port that JAMS is using to communicate with the Agent. The default for outgoing Agents is 77.
- **Version** - The version of JAMS the Agent is running.

Defining an Agent in JAMS

1. Click **Agents** from the Shortcuts menu.
2. In the Agent Definitions screen, select **Add**.
3. In the Name field, enter a name for the Agent. The name of a JAMS Agent is a friendly name for the Agent in JAMS. The JAMS name of an Agent is not required to match the system name of the Agent machine. You can enter the HostName or IP address in the Address property of the Agent definition, if it is different from the Display Name.
4. In the Type of Agent field, select the type of Agent.
5. In the Type of Platform field, select the platform for the Agent.
6. Click **Ok**. By default, JAMS automatically opens the full Agent Definition dialog, so you can configure it.

Configuring an Agent in JAMS

After an Agent has been defined in JAMS, you can configure its properties, including the Job Count/Limit, connection settings, security, and access controls.

1. Click **Agents** from the Shortcuts menu.
2. Right-click an Agent, and select **Properties**.
3. Select the **State** tab to view Agent information, or set a Job Limit for the Agent.
4. In the **Properties** tab, you will see a list of Agent Properties tailored to the Type and Platform of the Agent. You can also change the Type of Agent and the Type of Platform by using the drop-down boxes. Some Agent Type and Agent Platform combinations are not supported. If you select an unsupported combination, an Invalid Agent error message is displayed.
5. Use the available Agent Properties to configure the Agent. For example, on an Outgoing Windows Agent, the Address would be defined.
6. Select the **Security** tab to modify group permissions to the Agent.

NOTE: Ensure the **Submit** permission is selected for user accounts that will use this Agent.

7. Click **Save and Close**.

NOTE: After the Agent is created, it is recommended that you set up TLS communication if SSPI is not available. See [Setting up Secure Communication with Outgoing JAMS Agents](#). This will be required for communication to occur. You will see the Agent state listed as Unauthenticated until this security is configured.

NOTE: If you change the security settings for Allow No Remote Host Security or Allow SSPI Remote Host Security, the changes will be applied when the Agent has a new connection. This may occur when the Agent is restarted or when the next connection is made, if the Connect on Demand option is used.

Deploying an Agent in JAMS

After you define an Agent in JAMS, it can be deployed from the Agent Definitions view. The JAMS Scheduler offers built-in capabilities to install a JAMS Agent on machines running Windows or System I (AS400).

NOTE: See [Manual Installation of JAMSAgentX](#) for information on deploying on Linux.

NOTE: OpenVMS and Linux Agents cannot be push deployed from the JAMS Server.

If the Agent already exists, follow the steps below to deploy or upgrade the Agent.

1. Click **Agents** from the Shortcuts menu.
2. Right-click the Agent to be deployed, and select **Deploy**.
3. In the dialog, select the JAMS User to be used to deploy the JAMS Agent. This user should have all necessary permissions to the defined JAMS Agent.
4. Click **OK**. The Deploy Agent Job runs in the Monitor view.

Configuring the Connection for Outgoing Agents

For an Outgoing Agent, you can control when the Agent is connected to the JAMS Scheduler by using the Connect On Demand option. This is useful if you want to limit the number of open and active connections to Agents that are rarely used.

By default, this option is disabled so the Agent connection remains open if no Jobs are running on it. If you enable this option, the Agent connects to the JAMS Scheduler only when a Job is running on the Agent. The Agent state changes to Idle when a Job is not running on the Agent. The version may also be Unknown if the JAMS Scheduler has not yet been connected to the Agent.

1. Click **Agents** from the Shortcuts menu.
2. Right-click the Outgoing Agent, and select **Properties**.
3. Click the **Properties** tab.
4. Select or clear the **Connect On Demand** checkbox.
5. Click **Save and Close**.

Canceling Jobs on Agents

When you cancel a Job, the JAMS Scheduler will no longer try to run it. If you cancel a Job that is waiting for reconnection, the Job will stay in the Canceling state until the Agent is reconnected.

If you cancel a Job on an Agent and Connect on Demand is enabled, the JAMS Scheduler will stop trying to connect to the Agent until another Job is run. Any status or changes will be delayed.

1. Click **Monitor** from the Shortcuts menu.
2. Right-click a Job that is running on an Agent.

3. Select **Cancel**.
4. When prompted to confirm the cancellation, click **Yes**.

Jobs on Agents can be force canceled by selecting the **Force** checkbox in the Cancel dialog or by using the `-force` parameter on the command line. This will cause the JAMS Scheduler to treat the job as completed, but it will try to stop the Job. If the Agent is not connected, the Job may not receive the cancel and may continue to run.

Agents in Batch Queues

Selecting an Agent in a Batch Queue is optional. If no Agent is specified in the Batch Queue and the Job Definition does not have/inherit an Agent, the Job executes locally on the Scheduler. If multiple Agent Names are specified, Jobs are distributed (load balanced) between the available Agent Names.

See the [Using Queues](#) article for information on adding an Agent to a Queue or deleting an Agent from a Queue.

Enabling or Disabling an Agent

By default, new Agents are set to Enabled so they can run Jobs. If you need to prevent new Jobs from running for a period of time, you can disable the Agent. Any Jobs that are currently running will continue, but no other Jobs will be submitted to run on that Agent.

1. Click **Agents** from the Shortcuts menu.
2. Right-click the Agent, and select **Enable** or **Disable**. The Enable column is updated.

Deleting an Agent in JAMS

Any JAMS Agent can be deleted from the Agent Definitions view. Removing the Agent from the JAMS Client will not remove the JAMS Agent software from the Agent machine.

1. Click **Agents** from the Shortcuts menu.
2. Right-click the Agent to be deleted, and select **Delete**.
3. When prompted to confirm the deletion, click **Yes**.

JAMS Agent for System i

The JAMS Agent for System i is designed to support the IBM System i operating system (aka i5/OS and OS/400). JAMS connects to the IBM System iServer using TCP/IP protocols.

Deploying the JAMS Agent for System i

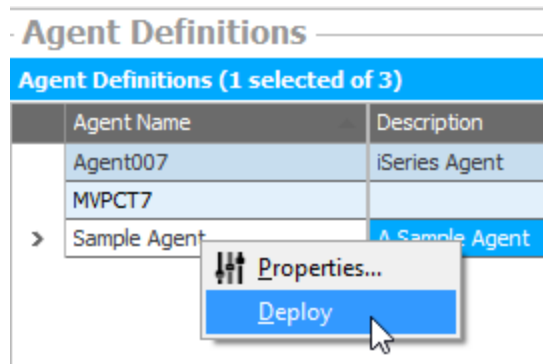
The JAMS Agent for IBM System i is installed along with the JAMS Scheduler, but must be pushed out to the agent machines. Specifically, deployment is controlled from the Agents view in the JAMS Client.

Before attempting to deploy, make sure you have provided each JAMS user the appropriate System i access along with the privileges required to deploy the agent.

NOTE: If the library JAMSLIB is present in both the system library list and user library list, remove JAMSLIB from the system library list.

Upgrading the JAMS Agent for System i

Installing an updated version of JAMS may already include the JAMS Agent. To deploy the update, select one or more agents from the Agent Definitions view, right-click, and select **Deploy**.



Troubleshooting the JAMS Agent for System i Deployment

The most common issues with the JAMS Agent deployment are:

- The user selected for the deployment job does not have sufficient privileges on the target machine.
- FTP settings are disabled or blocked.

To gain a more detailed view of the deployment, review the Job Log of the DeploySeriesi in Job History.

If the JAMS DeploySeriesi job does not work because of access restrictions, such as the FTP is blocked, you can manually deploy the agent by following steps described below:

1. Create a JAMS user profile:

```
CRTUSRPRF USRPRF(JAMS) PASSWORD(*NONE) SPCAUT(*JOBCTL)
```

2. Create a library:

```
CRTLIB LIB(JAMSLIB)
```

3. Create a save file:

```
CRTSAVF FILE(JAMSLIB/JAMSSAVF)
```

4. Copy the JAMSSAVF.SAVF from the JAMS Scheduler to the AS/400 server.
(JAMSSAVF.SAVF is located at C:\Program Files\MVPS\JAMS\Scheduler\Agents\IBMSeriesi by default.)

5. Restore the objects:

```
RSTOBJ OBJ(*ALL) SAVLIB(JAMSLIB) DEV(*SAVF) SAVF  
(JAMSLIB/JAMSSAVF) MBROPT(*ALL) RSTLIB(JAMSLIB)
```

6. Start the JAMS subsystem:

```
STRSBS SBSD(JAMSLIB/JAMS)
```

JAMS Agent for Windows

The JAMS Agent for Windows runs as a background service that executes JAMS Jobs under the direction of a JAMS Scheduler on a different machine. It is a best practice to use Agents to offload resources to another server other than the JAMS Scheduler. Agents can also be used when JAMS is configured for high availability, so Job execution is not interrupted during a failover event.

You have two options to install the JAMS Agent for Windows:

- Manually install the JAMS Agent for Windows
- Push deploy the JAMS Agent for Windows

Before You Begin

- Ensure the Agent meets the system requirements for the JAMS version. See the JAMS Installation Guide for more information.
- Ensure the user installing the JAMS Agent for Windows has administrator rights.
- Ensure the following ports are open and accessible through the firewall:
 - Port 77 - The JAMS Agent service listens on port 77 to communicate with the JAMS Scheduler.
 - Port 773 - To run Sequence, Workflow, or SQL Jobs on a Windows Agent, this port needs to be open on both the JAMS Scheduler Server and the Agent Server.
- For only the Push Deploy option, Windows RemoteAdmin needs to be enabled and accessible from the JAMS Scheduler server.
- For only the Push Deploy option, the administrator share on the remote server must be accessible from the JAMS Scheduler server.

Manually Installing the JAMS Agent for Windows

The installer package (.msi) sets up an Agent service on the Agent machine called 'JAMS Agent'. A new directory will contain the executable files in C:\program files\MVPSI\JAMS\Agent by default.

1. On the JAMS Scheduler machine, go to Program Files\MVPSI\JAMS\Scheduler\Agents\Windows.
2. Copy the SetupAgent.msi or SetupAgentx64.msi file and paste it onto the remote Agent machine.
3. Double-click the .msi file to run the installer and follow the prompts. When the installation is complete, you will see a Windows service called 'JAMS Agent' running.
4. Define the JAMS Agent for Windows as described below.

Defining a JAMS Agent for Windows

After you install the Agent software, define the Agent within the JAMS Client so it can run Jobs.

1. Click **Agents** from the Shortcuts menu.
2. Click **+**.
3. In the Name field, enter a name for the Agent.
4. Click **Ok**.
5. Click the **Properties** tab.
6. Verify the Address field is correct.

7. Optional - If you are going to push deploy the Agent, you can set the Deploy Credential on the Agent. If you do not set it on the Agent, you will be prompted to select a Credential when you deploy the Agent.
8. It is recommended that the JAMS Agent Service run as 'Local System'. This is the default option and does not require additional configuration.

NOTE: After the Agent is created, it is recommended that you set up TLS communication if SSPI is not available. See [Setting up Secure Communication with Outgoing JAMS Agents](#). This will be required for communication to occur. You will see the Agent state listed as Unauthenticated until this security is configured.

Push Deploy the Windows Agent using Local User Accounts

When deploying a remote JAMS Agent to a Windows server, the standard practice is to use a domain account user with Administrator privileges on both the JAMS Scheduler and the remote JAMS Agent machine.

However, if you don't have an environment that uses domain accounts, there are two alternative options for setting up the remote JAMS Agent software for a Windows server:

- Manually install the JAMS Agent for Windows (see the preceding section).
 - To allow the push deployment of the Windows JAMS Agent to a remote server when not using domain user accounts, ensure the JAMS Scheduler and remote Agent server use the same administration level user account and password.
1. Click **Agents** from the Shortcuts menu.
 2. Ensure the Agent has been defined as described above.
 3. Right-click the Agent to be deployed, and select **Deploy**.
 4. In the dialog, select the JAMS User to deploy the JAMS Agent.
 5. This user should have all necessary permissions to the defined JAMS Agent.
 6. Click **OK**. The Deploy Agent Job runs in the Monitor view.

JAMS Agent Interactive for Windows

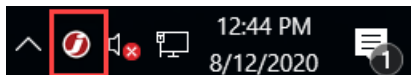
Overview

The JAMS Agent Interactive is an incoming Agent that is used to run Jobs that require interactive access to the Windows Desktop. This can include the integration for JAMS with Automate Desktop, if the Task requires interactive desktop access. If the Task requiring interactive access runs on a local/remote Agent that is not JAMS Agent Interactive, the Job may stay in the "executing" state.

NOTE: The JAMS Agent Interactive was created to automate applications. As a best practice, use a dedicated PC for this automation because Jobs can open and manipulate application windows. When a Job is running, you should avoid using this PC because it may disrupt the automated Job.

All Jobs running on this Agent run as the logged in or specified user, regardless of the ExecuteAs Credential for the Job.

The JAMS Agent Interactive runs as an application in the Windows taskbar notification area. You can right-click the icon to see the options listed below.



Option	Description
Scheduler Connections	View, add, or manage connections to a JAMS Scheduler. When you add a connection, an incoming Agent is also created on the specified JAMS Scheduler.
About	View version information about the JAMS Agent Interactive.
JAMS Support Website	Access the JAMS Support website site for more information.
Agent Documentation	Access the documentation for the JAMS Agent Interactive within JAMS Help.
Agent Log	View the log file for this Agent.
Disable application start on login/Automatically start application on login	<p>Enable or disable the JAMS Agent Interactive from starting when you log in to the machine. The text that is displayed varies based on the current state of this option.</p> <p>By default, the application automatically starts on login after you create a connection to a JAMS Scheduler as described in the section below.</p>
Exit	Close the JAMS Agent Interactive.

Installing the JAMS Agent Interactive

If you are integrating JAMS with Automate Desktop, install the JAMS Agent Interactive on the same machine as Automate Desktop.

1. On the JAMS Agent machine, right-click the JAMS installer and select **Run as Administrator**.
2. Follow the prompts until you see the Integration Pack page.

3. Select **JAMS Agent Interactive**.
4. Click **Next**.
5. Follow the remaining prompts in the installer to complete the installation.

Opening and Closing the JAMS Agent Interactive

You can open the JAMS Agent Interactive by right-clicking the JAMS Agent Interactive icon and selecting **Scheduler Connections**.

To close the JAMS Agent Interactive, right-click the JAMS Agent icon and select **Exit**.

Setting up the JAMS Agent Interactive

After the JAMS Agent Interactive is installed, you can set up the connection to a JAMS Scheduler. The initial setup of the JAMS Agent Interactive includes configuring secure communication and adding connections to one or more JAMS Schedulers.

Follow the steps below for environments that do not use primary/secondary servers in a failover environment.

Setting up Secure Communication

The procedure needs to be done only once during this initial setup. Ensure you run the commands below as a local administrator.

NOTE: If you already have a root certificate, do not run the **create root ca** command below. Otherwise, you will need to regenerate all of your certificates. You can check if you have a root CA by opening JAMSRegister.exe and running the **list root ca /default** command. This command also lists the thumbprint for the root CA.

1. On the JAMS Scheduler, go to \Program Files\MVPSI\JAMS\Scheduler.
2. Right-click **JAMSRegister.exe** and select **Run as administrator**. A command prompt window opens.
3. Run the command shown below.

```
create root ca /storekey /makedefault
```

4. Run the command shown below.

```
ADD EXECUTOR LISTENER /NEWCERT /FW
```

5. Close the JAMSRegister.

Adding a Connection to a JAMS Scheduler

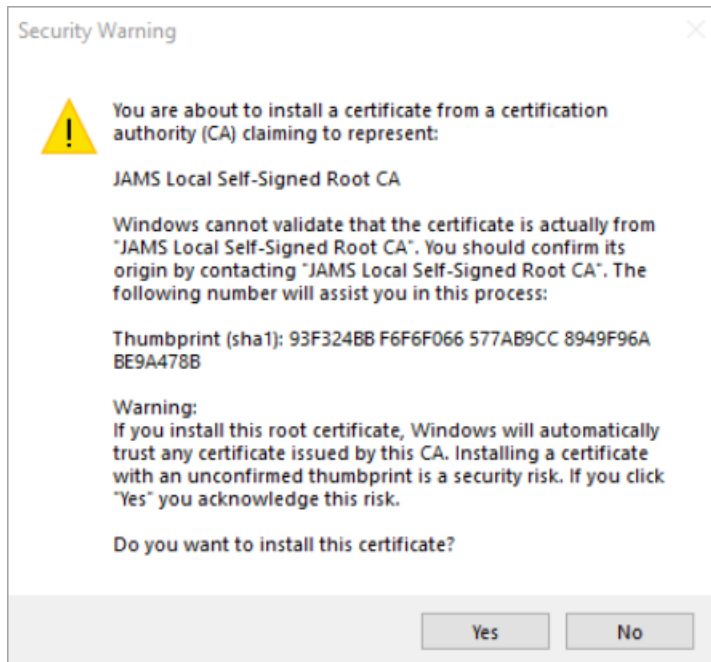
After the JAMS Agent Interactive has been installed, add a connection to the JAMS Scheduler. This process creates an Agent definition on the selected JAMS Scheduler with the ACL restricted to the user account specified in the Username field below. The Agent is named "MachineName-UserName" where the MachineName is the name of the machine where JAMS Agent Interactive is installed and the UserName is the user adding the connection.

NOTE: If you do not specify a username and password, the JAMS Agent Interactive runs as the currently logged in user.

Ensure your credentials can connect to the JAMS Server as well as create and modify Agents within JAMS.

1. Go to the Agent machine.
2. In the Windows taskbar notification area, right-click the JAMS Agent Interactive icon and select **Scheduler Connections**.
3. Click **Add**. The Add New Scheduler Connection dialog is displayed.
4. In the Server Name field, enter the name of the JAMS Server. If you enter a valid server name, you do not need to enter an IP address in the Address field.
5. In the Address field, enter the IP address or name of the JAMS Server, if necessary.
6. In the Port field, keep the value as 773.
7. In the Username field, enter the user name to connect to the JAMS Server. This account will also be used to create the incoming Agent.
8. In the Password field, enter the password for the user name that you entered.
9. Click **Test Connection** to verify the Agent can communicate with the JAMS Scheduler. A message is displayed to indicate if the connection was successful.
10. Click **Ok**.

11. If the server that you specified is on a different machine, you will see the message below. Click **Yes** to install the certificate from the specified server.



The JAMS Agent Interactive is now listed as an incoming Agent within the Agents option in the JAMS Client. You can repeat the steps above to add additional connections, if necessary. You can now set Jobs to run on this Agent.

Setting up JAMS Agent Interactive in a High Availability Environment

To configure JAMS Agent Interactive in a high availability environment, ensure the primary and secondary servers use a shared name that is specified in the `/EXECNAME=` command to create the certificate. This name is not the actual machine name. If a shared name is not specified, the machine name of the primary server is used.

For example, if you have two servers named PROD-JAMS-PRM and PROD-JAMS-SCN, you could use PROD-JAMS for the shared name.

Primary JAMS Scheduler

1. On the primary JAMS Scheduler, go to `\Program Files\MVPSI\JAMS\Scheduler`.
2. Right-click **JAMSRegister.exe** and select **Run as administrator**. A command prompt window opens.

3. If you did not run the create root ca command from the previous section, run the command shown below. Otherwise, skip this step.

```
create root ca /storekey /makedefault
```

4. Note the thumbprint that is listed because you need this in the following step.
5. Run the command shown below to export the certificate. You need to copy this to the Secondary JAMS Scheduler.

```
Export root ca /filename="C:\root.pfx" /prompt  
/thumbprint=<Insert the thumbprint of the certificate>
```

6. Enter a password to store the private key, which is referred to as the root key password and will be used in another step.
7. Enter the password again when prompted to verify it.
8. Run the command shown below:

```
Add Exec Listener /NEWCERT /EXECNAME=<Enter the shared name for  
the primary and secondary JAMS Scheduler> /ADDRESS=<Enter the  
network address or name for the primary server> /FW
```

NOTE: The /ADDRESS is only needed if the machine name alone cannot specify the servers on the network. For example, you may need to enter the address if the servers are not within a company network.

9. Close the JAMSRegister application.
10. Copy the root.pfx file that you exported to the secondary JAMS Scheduler.

NOTE: If you need to redo the **Add Exec Listener** command due to incorrect information, run the **Remove Exec Listener** command in JAMSRegister to remove the current listener. You can then run the **Add Exec Listener** command again.

Secondary JAMS Scheduler

1. On the secondary JAMS Scheduler, go to \Program Files\MVPS\JAMS\Scheduler.
2. Right-click **JAMSRegister.exe** and select **Run as administrator**. A command prompt window opens.
3. Run the command shown below to import the certificate. Ensure you update the /filename command to specify the location where the file is saved.

```
Import root ca /filename=root.pfx /prompt /makedefault  
/storekey
```

4. Enter the root key password.

5. Run the command shown below:

```
Add Exec Listener /NEWCERT /ADDRESS=<Enter the network address  
or name for the secondary server> /FW
```

JAMS Agent Interactive Machine

Add the primary JAMS Scheduler as a connection using the steps in the “Adding a Connection to a JAMS Scheduler” section.

Managing Scheduler Connections

Viewing Information for a Connection to a JAMS Scheduler

1. In the Windows taskbar notification area, right-click the JAMS Agent Interactive icon and select **Scheduler Connections**.
2. On the Scheduler Connections tab, select the connection that you want to edit.
3. Click **Edit**. The Edit Scheduler Connection dialog is displayed.
4. View the information in the Agent Information section. This section is read-only.
 - Server Name is the name of the server that you entered when the connection was created.
 - Address is the name or address of the selected server followed by the port number.
 - Agent Name is the name of the Agent that includes the machine name and the user account that created the connection.
 - Agent Thumbprint is the Agent thumbprint of the existing certificate that is used for communication. (The root certificate has a different thumbprint.)
5. Click **Ok**.

Requesting a New Certificate

You can create a new certificate from the JAMS Scheduler if the current certificate authority has changed or the certificate has expired.

1. In the Windows taskbar notification area, right-click the JAMS Agent Interactive icon and select **Scheduler Connections**.
2. On the Scheduler Connections tab, select the connection that you want to edit.
3. Click **Edit**. The Edit Scheduler Connection dialog is displayed.
4. View the information in the Agent Information section. This section is read-only.

5. In the Request New Certificate section, do the following:
 1. In the Server Name, view or change the JAMS Server that this Agent will use to request the new certificate. If you entered a server name and an address when the connection was initially created, the address is displayed in this field. If you entered only a server name, the name is displayed.
 2. In the Port field, keep the value as 773.
 3. In the Username field, enter the user name to connect to the JAMS Server.
 4. In the Password field, enter the password for the user name that you entered.
6. Click **Request Certificate**.
7. Click **OK** on the Request New Certificate dialog.
8. Click **Ok**.

A new certificate is created and stored for this Agent. The Agent Thumbprint on the Edit Scheduler Connection dialog is updated.

Removing a Connection to a JAMS Scheduler

1. In the Windows taskbar notification area, right-click the JAMS Agent Interactive icon and select **Scheduler Connections**.
2. On the Scheduler Connections tab, select the connection that you want to remove.
3. Click **Remove**.
4. Click **Yes** when prompted. The connection is removed.

As a best practice, you may want to remove the Agent from within the JAMS Client and review any existing Job Definitions that may specify this Agent.

Viewing Information about the JAMS Agent Interactive

Viewing Version Information for the JAMS Agent Interactive

1. In the Windows taskbar notification area, right-click the JAMS Agent Interactive icon and select **About**.
2. View the version numbers on the About screen.

Viewing the Log File for the JAMS Agent Interactive

If you need to view detailed information about the JAMS Agent Interactive, you can view the log file.

1. In the Windows taskbar notification area, right-click the JAMS Agent Interactive icon and select **Agent Log**. The JAMSAgentInteractive log file is opened as a .txt file.
2. Review the information listed in the log file.

JAMS Agentd for Linux/Unix

Agents Introduction

Agents allow users to define specific actions that can run locally or remotely throughout a JAMS network. By default, JAMS can only run Jobs and Sequences on the Primary Engine. Agents allow users to extend that capability to run Jobs on a variety of different environments. For example, by using JAMS Agents, a single JAMS Scheduler can execute tasks on many systems.

The JAMS Agentd is a version of the JAMS Agent that runs as a background process on Linux/Unix machines. This topic details how to install this Agent to your primary machine.

NOTE: Jobs running on an AgentD Agent must use or be based on either the **UnixShell** or **UnixScript** execution methods.

Installing the Agent

There are two ways to install the Agent:

- **Install from a package (.rpm or .deb):** You need to use appropriate package management tools (rpm, yum, apt-get, yast). The .rpm and .deb packages are preferred if you need to install in the standard location as root. The package can also be installed in a non-standard location, but it only handles installation as root. It also has the advantage of allowing the package system to check for updated versions automatically. (See the notes on the system distribution specific Repo page.)
- **Install from the tarball (.tgz):** This is an archive of the files necessary for the install, which needs to be manually installed by following the instructions in the README file included with the .tgz file. This is the preferred method if you need a non-root install.

Installing the Agent from a Package

1. Go to the JAMS Scheduler Repository at: [JAMS Scheduler Repository](#).
2. Select the repository for your system.

- Download the appropriate jams-agentd package (.rpm [Red Hat packages], .deb [Debian packages], or .rpm [SUSE packages]). For Red Hat, there are different builds for each release.

JAMS Scheduler Repository

Debian, Red Hat, and SUSE repositories are available for packages by Fortra with separate Red Hat release directories available for RedHat 5, 6, and 7.

These repositories use [0x0744742A](#) as signing key.

Repositories

Name	Description
Debian	Debian Repo for use with Debian, Ubuntu, Mint, and others.
Red Hat	RPM Repo for use with Red Hat Enterprise Linux, Fedora, CentOS, Oracle Linux, and others.
SUSE	SUSE Repo.

JAMS AgentD

[Release Notes](#)

JAMS AgentX

[Release Notes](#)

- Copy the package to the target machine and become root.
- Use the package manager to install the package.
- It may be necessary to open the firewall for the JAMS Server to connect to the Agent. In this case, the default port 77.

For example, use the following for Red Hat rpm. You can use the --prefix PATH option to relocate the RPM package. (This is not available for the Debian packages.)

```
# rpm -i jams-agentd-6.5.31-1.el7.x86_64.rpm
```

NOTE: JAMS AgentD must be installed as root.

The installer adds the following files:

Root Installation Directory	File Name	Description
/usr/sbin	jams-agentd	Service daemon and management command.
	jams-register	Connection security management command.

Root Installation Directory	File Name	Description
/usr/share/doc/jams-agentd	CHANGES	The list of version changes.
	COPYRIGHT	The copyright statement.
	NEWS	Additional details for the list of changes.
	README	Additional information about installation and management.
/usr/share/doc/jams-agentd/openssl	License	The OpenSSL license.
/usr/share/man/man8	jams-agentd.8.gz	The manual page for jams-agentd.
	jams-register.8.gz	The manual page for jams-register.

The installer also creates a standard configuration, if one is not present. This includes the following certificates and directories:

Root Installation Directory	File Name	Description
/etc/init.d	jams-agentd	The service control script.
/etc/jams	agent-client.config	The Connection configuration file.
	jams-agentd.config	The Service configuration file.
/etc/jams/certs		The Secure SSL certificate storage.

Root Installation Directory	File Name	Description
/etc/pam.d	jams-agentd	The PAM authentication file for the Agent.
/var/log/jams		The JAMS related log files.
/var/run	jams-agentd.pid	The executing Agent process id file.
/var/tmp/jams		JAMS Job temporary files (Job scripts).

The jams-agentd daemon has the following built-in commands for managing the daemon service:

Command	Description
jams-agentd -E	Stop the service.
jams-agentd -F	Restart the service.
jams-agentd -G	Start the service.
jams-agentd -T	Report service status.

The jams-agentd daemon can also be managed directly, if the service interface is not being used. Use the following commands:

Command	Description
jams-agentd -d	Start the jams-agentd daemon in the background.
jams-agentd -K	Stop the jams-agentd daemon.
jams-agentd -Q	Test if the daemon is running. Exit code 1 if not.

NOTE: Making any modifications to the configuration file will require a daemon restart. The configuration file includes comments describing the various available options.

Avoid stopping and restarting the daemon directly because it may cause problems with the service management software.

Testing the Connection and Connecting from the JAMS Scheduler

Ensure the correct ports are open/listening on the agentd machine. Port 77 is the default port that jams-agentd uses. You will want to test this connection from the JAMS Scheduler machine by using Telnet or PowerShell. See the example below:

```
telnet YourAgent 77
test-netconnection YourAgent -port 77
```

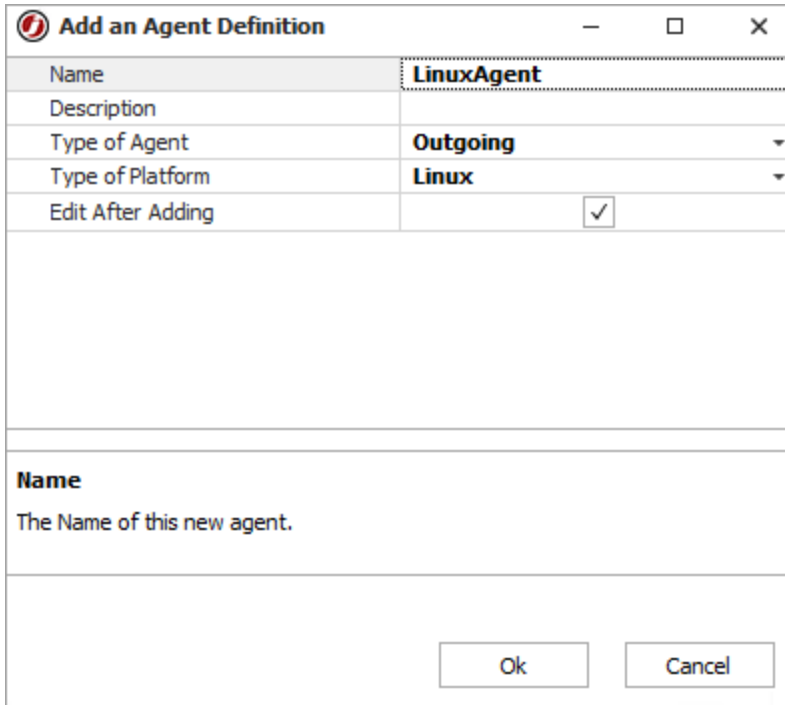
Creating the Agent Definition

After you verify the connection from the JAMS Scheduler machine to the agentd machine, you can create the Agent Definition in the JAMS Client.

NOTE: After the Agent is created, it is recommended that you set up TLS communication. See [Setting up Secure Communication with Outgoing JAMS Agents](#). This will be required for communication to occur. You will see the Agent state listed as Unauthenticated until this security is configured.

1. In the JAMS Client, click **Agents** from the Shortcut menu.
2. Click the **Add (+)** button to add an Agent.
3. Enter the Name, Description, Type of Agent, and Type of Platform.

4. Click OK.

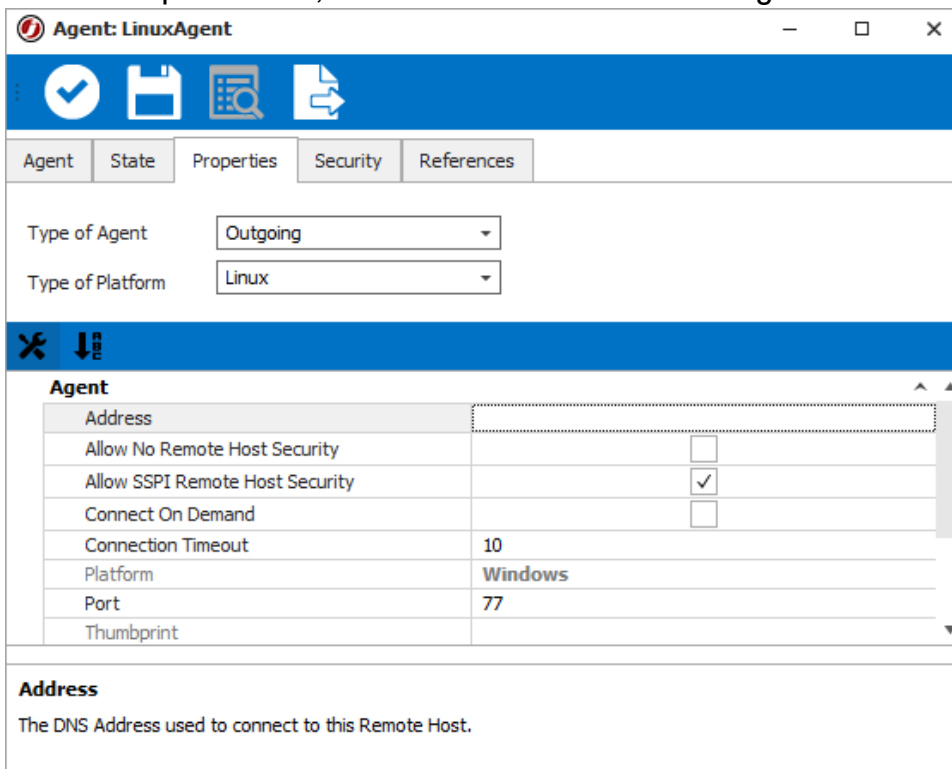


The dialog box titled "Add an Agent Definition" contains the following fields and controls:

Name	LinuxAgent
Description	
Type of Agent	Outgoing
Type of Platform	Linux
Edit After Adding	<input checked="" type="checkbox"/>

Below the table is a large empty text area. At the bottom right are "Ok" and "Cancel" buttons.

5. On the Properties tab, enter the IP address for the Agent.



The "Agent: LinuxAgent" window shows the "Properties" tab. It includes a toolbar with icons for check, save, search, and print. Below the tabs are dropdowns for "Type of Agent" (Outgoing) and "Type of Platform" (Linux). A blue bar with a refresh icon and a list icon is below the dropdowns. The "Agent" section contains a table with the following properties:

Address	
Allow No Remote Host Security	<input type="checkbox"/>
Allow SSPI Remote Host Security	<input checked="" type="checkbox"/>
Connect On Demand	<input type="checkbox"/>
Connection Timeout	10
Platform	Windows
Port	77
Thumbprint	

Below the table is a section titled "Address" with the text: "The DNS Address used to connect to this Remote Host."

6. Click the **State** tab.

7. View the License Allocated section. The License Allocated section is blank until you execute the first Job on the Agent. The State section displays "Idle" when the JAMS AgentD is currently not executing Jobs.
8. Click **OK**.

Creating a Credential

Create a Credential for a user on the Linux machine that has the ability to log in and execute the process that is defined in the JAMS Job.

NOTE: It is not necessary to create a root credential to run Jobs. The Credential should give the Job the minimum permissions necessary to run the required Job. If a root Credential is required, it should be very carefully protected against unauthorized use.

1. In the JAMS Client, click **Credentials** from the Shortcut menu.
2. Click the **Add (+)** button to add a new Credential.
3. On the Add a New JAMS Credential dialog, enter the Logon As and Password fields.
4. Click **OK**.

Add a New JAMS Credential	
Credential Name	LinuxAdmin
Logon As	User
Enter Password	*****
Re-Enter Password	*****
Edit credential after adding	<input checked="" type="checkbox"/>
<p>Re-Enter Password</p> <p>Re-enter the same password to help avoid mistakes.</p>	
<p>Ok Cancel</p>	

Creating a Job Definition

1. In the JAMS Client, click **Definitions** from the Shortcut menu.
2. Click the **Add (+)** button to add a new Definition.
3. On the Add a New JAMS Job Definition dialog, select an **Execution Method**.

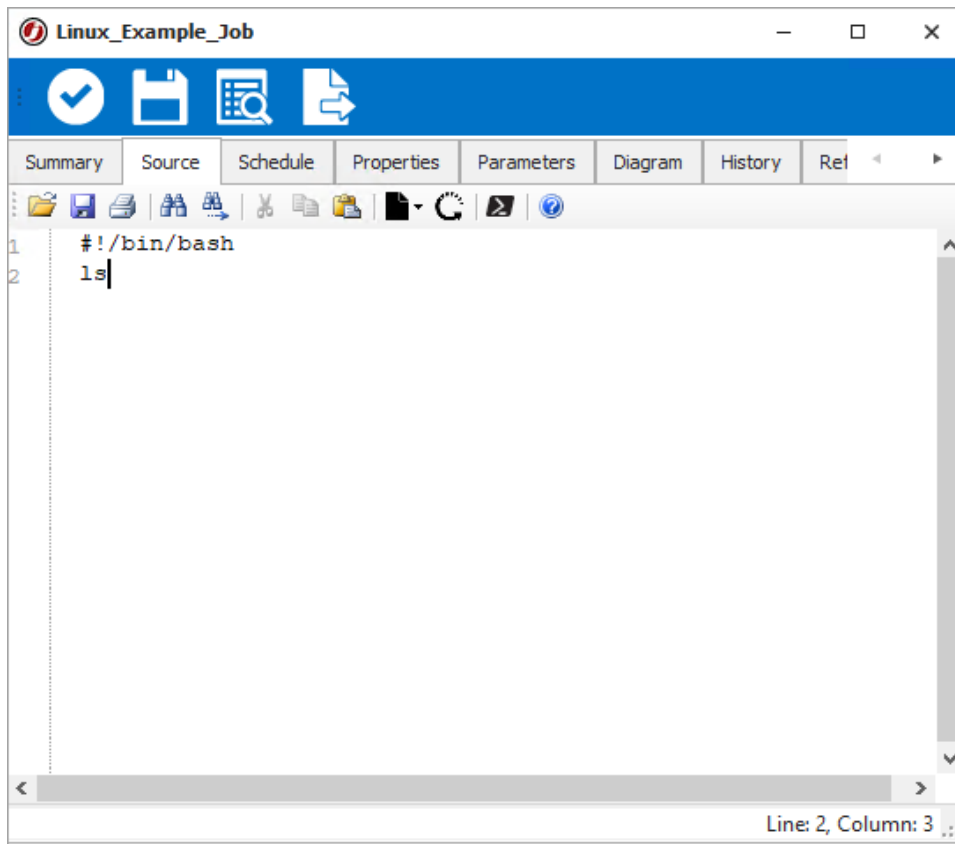
Folder	\JAMSJobs
Name	Linux_Example_Job
Description (optional)	
Execution Method	UnixShell
Edit this job definition after adding	<input checked="" type="checkbox"/>
Scheduled Date (optional)	
Calendar (optional)	[Select a Calendar]
Scheduled Time (optional)	12:00 AM

Execution Method
Identifies the execution method used to execute this job.

Ok Cancel

4. Click the **Source** tab.
5. Enter the following:

```
#!/bin/bash  
ls
```

6. Click the **Properties** tab.
7. Under Schedule, set the Agent.
8. Under Execute, select **LinuxAdmin**.
9. Click **OK**.

Public Key Authentication

PUBLIC-KEY is a valid user_authentication option in the config file. It uses the same public-private key in JAMS as SSH.

PAM Support

Currently, the PAM support needs to have a PAM configuration file for jams-agentd. This is expected to be in the /etc/pam.d directory. When installing jams-agentd, the installer will attempt to create this file by copying the file for sshd with certain edits.

The PAM support uses only auth and account parts. No password updates are possible and a session is not used. This may cause problems with some types of PAM plug ins.

jams-agentD Commands

Commands

The following table lists the commands available for the jams-agentd command.

Short Command	Long Command	Argument Optional	Valid Options	Description
-d	--daemon-mode		-c -p	Fork/exit the program to run in the background.
-h	--host-mode		-c -j	Job hosting mode, used internally to run Jobs.
-s	--shared-mode		-c -u	Shared
-C	--create-config	file name	-p -u	Create config_file with default settings or send to standard out.
-E	--stop-service			Stop jams-agentd service.
-F	--restart-service			Stop and start jams-agentd service.
-G	--start-service			Start jams-agentd service.
-I	--install-jams		-c -p -r -u	Install jams-agentd files and configurations with prompts.
-J	--install-quietly		-c -p -r -u	Install jams-agentd files and configurations without prompts.
-K	--kill-daemon	signal number		Stop jams-agentd daemon, using the optional signal.
-P	--pam-config	file name		Create standard PAM config file or send to standard out.
-Q	--query-status			Check for running daemon.
-R	--remove-service			Remove the jams-agentd service.
-S	--service-file	file name	-c -u	Create the standard Service control file or send to standard out.

Short Command	Long Command	Argument Optional	Valid Options	Description
-T	--service-status			Report the status of jams-agentd service.
-U	--update-config			Update the config file with latest settings (planned).
-V	--version			Report the current version and configuration locations.
-?	--help			List of commands and options.

Options

The following table lists options available for the jams-agentd command.

Short Option	Long Option	Argument	Description
-c	--config-file	file	Use an alternate configuration file.
-j	--job-id	job-id	The ID of the running Job.
-l	--log-level	log-level	Set the logging level to ERROR, WARN, INFO, DEBUG, or TRACE.
-p	--port-number	port	Listen on alternate port.
-r	--root path	install-dir	Directory to use as the root for installation.
-u	--user-name	user name	User environment.

Integrations

JAMS provides integrations with a variety of other programs and tools. You can install the integrations by using the JAMS installer. The articles in this section describe how to set up and configuration these integrations.

Automate

JAMS provides two integrations with Automate. You can integrate with Automate Desktop to run Tasks that require interactive desktop access or Automate Plus/Ultimate to run Workflows in Automate from JAMS. See the articles in this section for more information.

Automate Desktop

JAMS can be integrated with Automate Desktop to run Jobs that require interactive desktop access. The Integration with Automate Desktop is included with the JAMS installer. If the tasks from Automate require interactive desktop access, you need to install and configure the JAMS Agent Interactive, which is an option in the JAMS installer. If the Task requiring interactive access runs on a local/remote Agent that is not JAMS Agent Interactive, the Job may stay in the "executing" state.

Before You Begin

- Ensure you have access to the AML files.
- If the Task requires interactive desktop access, install and configure the JAMS Agent Interactive on the same machine as Automate Desktop. See [JAMS Agent Interactive for Windows](#) for more information.
- Ensure the JAMS Scheduler, JAMS Agent, and JAMS Client are running JAMS V7.2.X or higher.
- Ensure Automate Desktop 11.6 is used. JAMS V7.4 and higher supports only Automate 11.6.

NOTE: If you are using a version earlier than JAMS V7.4 or Automate 11.6, contact JAMS Support for assistance configuring the integration.

Creating a Sequence Job to use the Automate Sequence Task

To run an Automate Task using JAMS, you can import an existing AML file into the Automate Task within a JAMS Sequence.

1. From the JAMS Home screen, click **Create a Job**.
2. Select **Create a Sequence (multi-step) Job**.
3. In the Add a New JAMS Job Definition dialog, do the following:
 1. Select the Folder where the Job should be created.
 2. In the Name field, define a name for the new Job.
 3. In the Description field, enter a description for the Job, if necessary.
 4. In the Execution Method field, ensure **Sequence** is selected.
 5. Click **OK**. The Source tab for the Job is displayed.
4. Drag and drop the Automate Task onto the Design area (middle panel) on the screen.
5. In the Properties Panel, do the following:
 1. In the Display Title field, enter a name for the Automate Task, if necessary. (This field is updated after the AML file is selected for the Automate Task field.) By default, it is the name of the Automate Task from within the AML file or it is the name of the AML file, without the file extension.
 2. In the Automate Task field, click in the field to browse to the location of the AML file and then click **Open**. After the AML file is selected and imported into the Automate Sequence Task, its contents are saved within JAMS and the AML file is no longer needed.
 3. In the Automate Variables field, verify the variables were correctly imported from the AML file. The variable names are read-only. You can edit the value for a variable only if the ISPARAMETER attribute in the AML file is set. This is displayed as the Is Parameter checkbox under the variable value.

NOTE: Ensure the variables in AML files are not named "path", which is a reserved term. Using this term may cause issues when running the Automate Task in JAMS. If a variable is named "path", update the variable name within Automate before creating the AML file.

4. In the Output Names field, enter a list of comma-separated names that will be used as variables to receive values from Automate Desktop, if the variable names match the names sent from Automate. Successor Tasks can use these values by using the specified name surrounded by brackets. For example, if you enter "output1" in the Output Names field, enter "{output1}" in a field in a successor Task to use that value.

NOTE: If a variable has the ISPARAMETER attribute set in the AML file, the variable cannot be used in the Output Names field. When this attribute is set, the variable value is not accessible to other Sequence Tasks in JAMS.

6. If the Task requires interactive desktop access, click the Properties tab to set the Agent to the JAMS Interactive Agent.
7. In the Agent field, select the name of the JAMS Interactive Agent.
8. Click **Save and Close**.

After the Sequence is set up and saved, you can run the Job and view it in the JAMS Monitor.

Automate

The integration with Automate Plus/Ultimate lets you run Workflows in Automate from JAMS. The Job can then be monitored from the Monitor view within JAMS. This integration does not require enabling it in the JAMS installer. The Workflow in Automate can contain variables. Any available input variables for the Workflow appear as Parameters on the Job.

NOTE: When JAMS matches its Parameters to the Automate Workflow Variables, only the Variables at the Workflow level (Shared Variables) are matched. JAMS does not display Variables on the individual Tasks inside the Workflow.

NOTE: The Variables from Automate will overwrite all parameters on the JAMS Job. If you want to include parameters in JAMS, it is recommended that you add them at the folder-level so the Job inherits them.

Before You Begin

- Ensure the Automate REST API has been enabled within the Automate application. See Automate API documentation in the [Community Portal](#) for more information.
- Ensure you have the correct port number for accessing the REST API. Otherwise, the default port, 9708, is used.
- Ensure you have the name or Id of the Automate Workflow.
- Install the JAMS Windows Agent if the JAMS Scheduler will not be running the Workflow. See [JAMS Agent for Windows](#).

Creating a Credential

Create a Credential in JAMS that will be used to connect to the Automate server. This Credential contains the Username and Password for authenticating to the Automate REST API. Users must have Inquire and Get Password permissions to this Credential to retrieve Workflows from Automate.

See [Working with User Credential Definitions](#) for more information.

Creating an Automate Job in JAMS

Ensure the user creating the definition has permissions to the Credential that is referenced in the Job. If you change the Automate Server or Port fields, the Automate User and Workflow fields will be cleared.

NOTE: If you change the Automate Server or Port fields, the Automate User and Workflow fields as well as the Job parameters will be cleared. Any inherited parameters from the folder-level are still displayed.

1. Click **Definitions** from the JAMS Shortcuts menu.
2. Click a folder where the Job will be saved.
3. Click **+**.
4. In the Name field, enter a name for the Job.
5. In the Execution Method field, select **Automate**.
6. Click **Ok**.
7. Click the **Source** tab.
8. In the Automate Server field, enter the IP address or server name of the Automate server.
9. In the Port field, enter the port for the Automate server. The default is 9708.
10. In the Automate User field, select the JAMS Credential that you previously created.
11. In the Workflow Name, select or type the Workflow name or Id in Automate that you want to run. If there are Workflows with the same name in different folders, enter the Workflow Id.
12. Click the **Schedule** tab to add schedule items to set when the Job will run.
13. Click the **Parameters** tab to view any variables that are defined on the Automate Workflow. You can override these values, if needed.
14. Click the **Properties** tab.
15. In the Execute As field, select a credential to run the Job.
16. In the Agent field, select the Agent if the Job will not run on the Scheduler.
17. Click **Save and Close**.

Banner in JAMS

Introduction

JAMS offers two robust and user-friendly integrations with Ellucian Banner. The V2 Banner Procedure option comes with updates to the Execution Method to support direct integration with the Banner Oracle database, while the V1 Banner option may still be configured if desired.

Both Banner integrations allows JAMS users to take advantage of Banner's core functionality such as registration, grade distribution, human resource information, financial aid processing, and other financial information.

When the adapter runs, it initiates communication with the Banner application server and instructs executing Banner Jobs as configured in JAMS. The process can then be monitored, canceled, and restarted in either Banner or JAMS.

NOTE: A JAMS V7.1/7.2 Client or Agent cannot be used with a JAMS V7.3 or higher Scheduler. Upgrade the JAMS Client or JAMS Agent to V7.3 or higher to use the Banner Integration.

Advanced Queues

JAMS now supports Advanced Queues in Banner. No configuration is needed within JAMS for Advanced Queues because it is self-contained within the Banner Package. If Advanced Queuing is detected and enabled, JAMS uses Advanced Queues. If Advanced Queuing is not enabled, dbms pipes are used.

Setting up the V2 Banner Procedure Integration

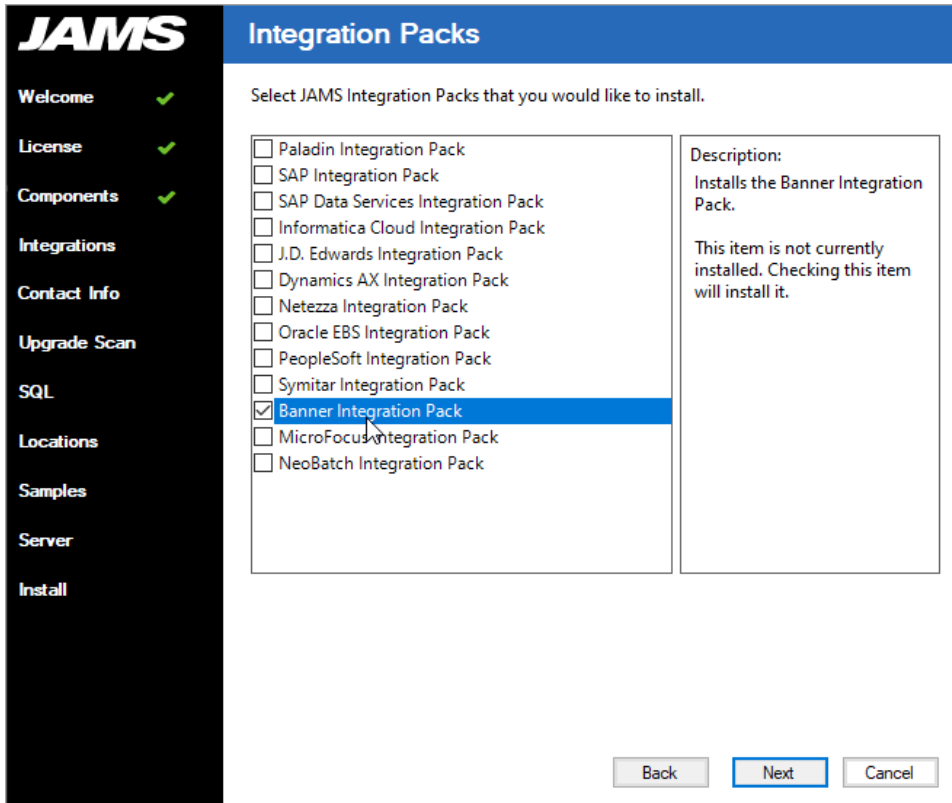
Load the Banner Integration Pack

You can load the Banner Integration Pack by using the JAMS installer.

NOTE: The V2 Banner Integration requires the Oracle Integration Pack to be installed. It also requires the Oracle Connection Store Objects, which are part of the Integration Pack.

1. Run the JAMS installer.
2. Follow the prompts until you see the Integration Packs page.
3. Select the **Banner Integration Pack**.

4. Click Next.



5. Follow the remaining prompts in the installer to complete the installation.

If needed, you can run the `LOAD/ASSEMBLY=JAMSBannerIntegration` command in JAMSDBA to reset the Execution Methods back to their default states, if you selected the Banner Integration Pack during install.

1. Go to <JAMS Install Folder>\MVPSI\JAMS\Scheduler.
2. Right-click **JAMSDBA.exe** and select **Run as administrator**.
3. At the JAMSDBA prompt, run the `LOAD/ASSEMBLY=JAMSBannerIntegration` command.

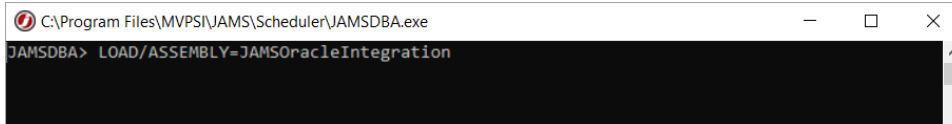
NOTE: Starting with JAMS V7.3, the `Load/Assembly=JAMSBannerIntegration` command is the preferred command over the `LOAD/BANNER` command. Previous versions of JAMS should run the `LOAD/BANNER` command.

Install the JAMS Oracle Integration the JAMS Server

After the Banner Integration Pack is installed, install the JAMS Oracle Integration.

It is recommended that you [create a maintenance window](#) if you are doing this during normal business hours due to restarting the JAMS Scheduler and JAMS Server services.

1. In the JAMSDBA, run the **LOAD/ASSEMBLY=JAMSOacleIntegration** command.

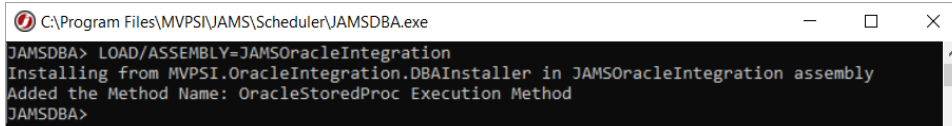


```

C:\Program Files\MVPSI\JAMS\Scheduler\JAMSDBA.exe
JAMSDBA> LOAD/ASSEMBLY=JAMSOacleIntegration

```

2. Verify the following screen is displayed after the assembly loads.



```

C:\Program Files\MVPSI\JAMS\Scheduler\JAMSDBA.exe
JAMSDBA> LOAD/ASSEMBLY=JAMSOacleIntegration
Installing from MVPSI.OracleIntegration.DBAInstaller in JAMSOacleIntegration assembly
Added the Method Name: OracleStoredProc Execution Method
JAMSDBA>

```

3. On the JAMS Scheduler, open the Windows Services application.
4. Click the JAMS Scheduler service and click **Restart**.
5. Click the JAMS Server service and click **Restart**.

Install the Oracle Stored Procedure on the Banner Database

1. Log in to the Banner Oracle database as "baninst1".
2. Compile the custom package Z_HS_JAMS_JOBUTIL. The package is located on the JAMS Scheduler in the <JAMS Install Folder>\MVPSI\JAMS\Integrations\Banner directory.

NOTE: The package contains functions and procedures for running Ellucian Banner Jobs through the JAMS Scheduler. Refer to the package definitions for additional details.

3. Create a public synonym for the package by running the command below.

```
CREATE PUBLIC SYNONYM Z_HS_JAMS_JOBUTIL FOR BANINST1.Z_HS_JAMS_JOBUTIL;
```

4. Create a custom role to be assigned to the database user that will be used by JAMS by running the command below.

```
CREATE ROLE HS_JAMS_ROLE;
```

5. Add the required grants to the role by running the commands below.

```
GRANT EXECUTE ON Z_HS_JAMS_JOBUTIL TO HS_JAMS_ROLE;
GRANT SELECT,INSERT,UPDATE ON GJBPRUN TO HS_JAMS_ROLE;
```

6. Create a database user that will be used by JAMS by running the commands below. Replace **<password>** with a valid password for the database user.

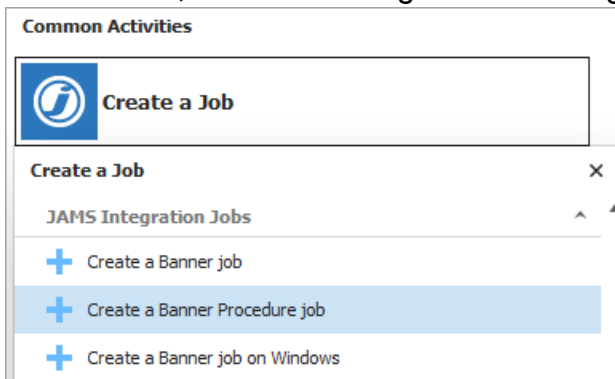
```
CREATE USER JAMSADMIN IDENTIFIED BY <password>;
GRANT CREATE SESSION TO JAMSADMIN;
GRANT HS_JAMS_ROLE TO JAMSADMIN;
```

Creating a JAMS Connection to the Banner Oracle Database

This option is available only if the Oracle Integration has been installed. See [Connections for Integrations](#) for more information.

Creating a Banner Procedure Job in JAMS

1. On the Home screen, click **Create a Job**.
2. Select **JAMS Integration Jobs**.
3. Select **Create a Banner Procedure job**. If you do not see the Jobs as shown below, close JAMS, install the Integration Pack again, and reopen JAMS.



4. On the Add a New JAMS Job Definition dialog, do the following:
 - In the Folder field, select the Folder where the Job will be saved.
 - In the Name field, enter a name for the Banner Job.
 - Click **OK**. The Banner Job is created. The Source tab is displayed.

Folder	\BannerProcedureJobs
Name	BannerJob
Description (optional)	
Execution Method	BannerProcedure
Edit this job definition after adding	<input checked="" type="checkbox"/>
Scheduled Date (optional)	
Calendar (optional)	[Select a Calendar]
Scheduled Time (optional)	12:00 AM

Name
The name of the new job.

Ok Cancel

5. On the Source tab of the Job, do the following:

- In the Banner Connection field, select your Banner Connection that was created in the Connection Store.
- In the Banner Job field, view the list of the available Jobs in Banner.
- In the Banner User field, select a JAMS Credential for the Banner user.
- In the Parameter Set field, select the Parameter Set that will be used when executing the Job. The Parameter Set will populate the Parameter tab within the JAMS Job.
- Review the **Include .log File** and **Include .lis File** check boxes. If there is sensitive information within the .log and .lis files and you do not want them in the JAMS Job log file, clear the check boxes for these two options.

NOTE: If multiple .log and .lis files are used in a Banner Job, they can be displayed in the JAMS Entry Log. If you do not see this information in the JAMS Log, increase the Log File Polling Interval value on the Properties tab to a value that slightly exceeds the expected run time of the Job. The default value is 30 seconds.

NOTE: If Banner Credentials have not yet been saved in JAMS, save them by clicking the **Credentials** Shortcut within the JAMS Shortcut menu in the JAMS Client. When adding the Banner Credential in JAMS, ensure the **Display Name** and **Logon Name** are the actual Banner user name. If a proxy user is used to submit Banner Jobs, the Logon Name should use the '[]' format, such as [banproxy], to match the format used by the GURJOBS process.

6. Click the **Properties** tab.

7. BannerProcedure Jobs need to run on the JAMS Scheduler server. This can be configured in the Properties tab by leaving the Agent field blank. If the default Agent inherited from the Folder is different than the JAMS Scheduler, select the JAMS Scheduler Server from the dropdown menu in the Agent field.

8. Click the **Save and Close** button.

NOTE: If Banner Credentials have not yet been saved in JAMS, save them by clicking the **Credentials** Shortcut within the JAMS Shortcut menu in the JAMS Client. When adding the Banner Credential in JAMS, ensure the **Display Name** and **Logon Name** are the actual Banner user name. If a proxy user is used to submit Banner Jobs, the Logon Name should use the '[]' format, such as [banproxy], to match the format used by the GURJOBS process.

Banner Properties

There are additional properties that you can modify. Properties can be inherited from the Job Properties tab for the Banner Procedure Execution Method or set within the Banner Procedure Sequence Task or Job Definition. (In the Banner Procedure Job, you can hover your cursor over a property to view a tooltip that indicates if the property is inherited.)

The following properties are available in a Banner Procedure Sequence Task and Job Definition:

Property	Description
Banner Connection	Sets the Connection Store for connecting to Banner.
Banner User	Sets the JAMS Credential for the Banner user. This Credential is used to run Jobs.
Printer Name	Sets the name of the printer to use.
Form Name	Sets the name of the Banner form.
Submit Time	Sets the submit time for the Banner Job.

Property	Description
Include .log file	Includes the contents of the .log file into the JAMS Entry Log. Multiple .log files can be included in the JAMS Log.
Include .lis file	Includes the contents of the .lis file into the JAMS Entry Log. Multiple .lis files can be included in the JAMS Log.
Log File Polling Interval	Sets the polling time for the Banner .log and .lis files. The default is 30 seconds. It is recommended that this value be set to a value that slightly exceeds the expected run time of the Job.
MIME Type	Sets the type of file used to format reports/output files. Select PDF or Plain Text.
Special Print	Sets the field that may be used to pass information to 3rd-party applications.
PDF Font	Sets the type of font to use for PDF reports.
PDF Font Size	Sets the size of the font to use for PDF reports.
Environment Type	Sets the type of environment for the Banner instance to either Unix or Windows. By default, this option is set to Unix . If the Oracle database instance is running on Windows rather than Unix, set this option to Windows .

Submitting a Banner Procedure Job

1. In the Shortcuts menu, click **Definitions**.
2. Expand the folder containing the Banner Procedure Job.
3. Right-click the Banner Procedure Job and select **Submit**. The Submit dialog is displayed. Any required parameters are also marked and need to be updated before the Job will run.

NOTE: If the required parameters are not marked, reload the parameter sets/parameter from within Banner.

4. Click **Submit Run Request**.

NOTE: If you cancel a Banner Job that is currently running, only the corresponding JAMS Job is canceled. The Job log file in JAMS displays a message indicating the Banner Job on the host may still be running. As a best practice, check the Job in Banner.

Setting up the V1 Banner Integration

1. Install AgentX on the Banner application server. This Agent can be automatically deployed or manually installed, specifically where the gjajobs.shl is stored.

Use the links at the end of this topic to learn more about deploying JAMS Agents.

NOTE: If the Banner server is running Windows, you may deploy a JAMS Windows Agent to the Banner server.

2. From the JAMS Client, create a JAMS user that has permission to access the Banner application server and can also connect to the remote Linux server. This can also be the same user that was used to deploy the Agent.
3. From the JAMS Client, select the **Execution Methods** Shortcut to access the list of available Execution Method options.
4. Double-click the **Banner** Execution Method to open its properties.
5. Select the **Template** tab.
6. Customize the template settings in the areas shown in the following screenshots.
Line 6 through 16- Set the source options for Oracle and Java.
Line 21 - Source the Banner environment.
Line 26 - Change the value to the Banner directory.
Line 28 through 33 - This section is used to increment the Banner One-Up-Number. This section should not need to be changed.
Line 35 through 40 - This section ensures the Parameters on the JAMS Job are passed to the Banner server. This section should not need to be changed.
Line 44 - Ensure gjajobs.shl is referenced properly. You may need to SSH into the Banner server and manually confirm this information.
Line 53 through 66 - This section pulls the .lis and .log files from Banner. This section should not need to be changed.

```

Method Name: Banner

Execution Method Parameters Template Properties Job Properties References

1  #!/bin/bash
2  #####
3  # JAMS Banner Execution Method
4  #####
5
6  #
7  # Set Oracle environment
8  #
9  ORACLE_SID=PROD;
10 export ORACLE_SID
11
12 JAVA_HOME=/u01/java17/JDK
13 export JAVA_HOME
14
15 PATH=${JAVA_HOME}/bin:${PATH}:${ORACLE_BASE}/product/11.2.0/client_64:${ORACLE_BASE}/product/11.2.0/client_x86/bin
16 export PATH
17
18 #
19 # Source the Banner environment
20 #
21 . /banvol/jams/bannerenv
22
23 #
24 # Change to Banner directory
25 #
26 cd /jobsub/ugdev8
27
28 echo "Getting next one up number"
29 one_up_number=$(sqlplus -S -R 3 <<JAMS.XM.Username>> /<<JAMS.XM.Password>> @$ORACLE_SID << END
30 SET ECHO OFF HEADING OFF FEEDBACK OFF PAGESIZE 0
31 select trim(general.gjbseq.nextval) from dual;
32 exit
33 END'
34
35 echo "Inserting parameter values"
36 sqlplus -S -R 3 <<JAMS.XM.Username>> /<<JAMS.XM.Password>> @$ORACLE_SID << END
37 exec dms_session.set_role(umw_getcmd('<<JAMS.XM.ScriptName>>'));
38 <<BannerParameter(JAMS.BannerParameters)%>
39 exit
40 END
41
42 echo "Calling Banner"
43
44 #
45 #
46 #
47 #
48 # Wait for the Banner job to complete
49 #
50 while [ -f /home/banjob/<<JAMS.XM.ScriptName>>_sone_up_number.shl ]
51 do
52     sleep 30
53 done
54
55 #
56 # Display the log files
57 #
58 if [ -f /home/banjob/<<JAMS.XM.ScriptName>>_sone_up_number.log ]
59 then
60     echo "*****.log file found*****"
61     cat /home/banjob/<<JAMS.XM.ScriptName>>_sone_up_number.log
62 fi
63
64 if [ -f /home/banjob/<<JAMS.XM.ScriptName>>_sone_up_number.lis ]
65 then
66     echo "*****.lis file found*****"
67     cat /home/banjob/<<JAMS.XM.ScriptName>>_sone_up_number.lis
68 fi
69
70 ##### END #####

```

NOTE: The Banner Environment may also be set using a bash profile (source ~/.bash_profile) if the Banner environment variables have been set on the default bash profile.

NOTE: The Banner directory and the directory containing the gjajobs.shl need to be configured.

NOTE: If sensitive data would be included in the Banner .log and .lis files, you may want to create a copy of the Banner Execution Method, and then remove the logging options from the copy's template. This new Banner Execution Method could be used in situations where Banner writes sensitive data to the Banner output files.

7. Make sure that **any** folder containing a JAMS Banner Job (including sub-folders) includes a path to the Banner Macro. This path must be populated in the Folders Properties (**Properties -> Source Options -> Template Library**).
By default, this path is C:\Program Files\MVPSI\JAMS\Scheduler\BannerMacros.xml
8. With the configuration complete, create a new JAMS Job and select the **Banner** Execution Method in the Job Definition dialog.
9. On the Job Definition Source page, set the Banner User, select a Job Type from the list, and then enter the Job Name.

NOTE: The Job name should not include the folder path or the .shl extension. The Job name in JAMS must be in the same case as what is stored on the Banner server.

10. Double-click the newly completed JAMS Job to access its properties.
11. Click the **Properties** tab and go to the Submit Options section.
12. Edit the Job by selecting the **Execute As** and **Agent** properties. The Execute As should be a UNIX account with access to JAMS AgentX.

Banner Parameters

Banner parameters always have an associated number, as seen in the left side of the example below.

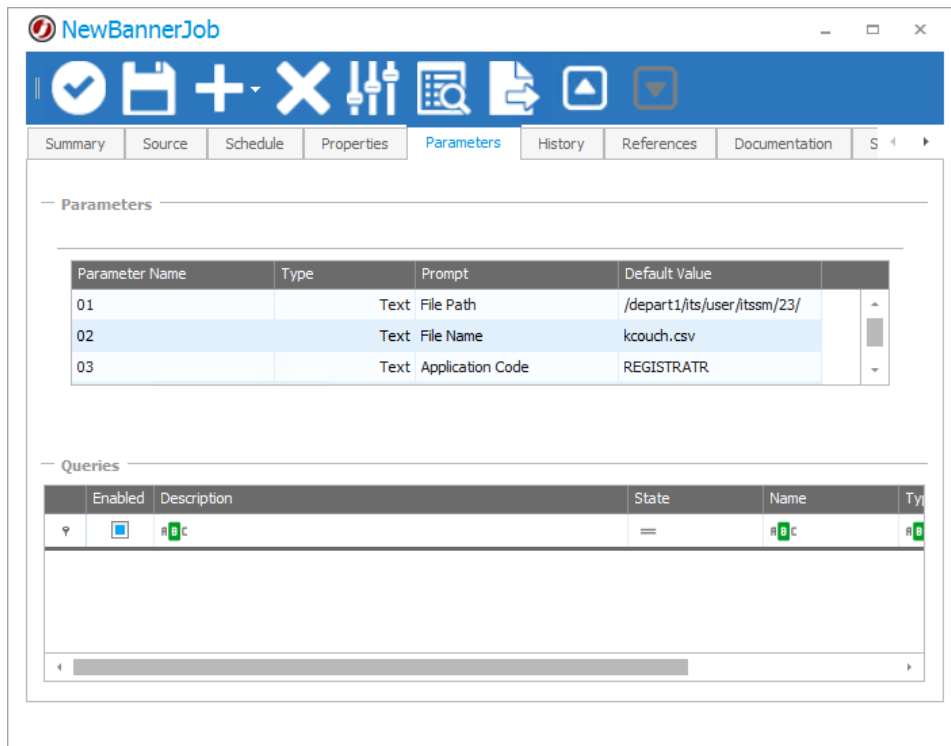
Parameter Values		
Number	Parameters	Values
01	File Path	/depart1/its/user/itssm23/
02	File Name	kcouch.csv
03	Application Code	REGISTRAR
04	Selection ID	SM_MANUEL
05	Creator ID	ITSSM23
06	Input Type (S, I or P)	I
07	Delete/Merge POPSEL Data	D

LENGTH: 30 TYPE: Character O/R: Required M/S: Single
Enter the complete file path and name (Example:/transfer/)

To pass parameters from JAMS to Banner, create JAMS Parameters with names that match the Banner parameter numbers on the given Job. Set the default value of the JAMS

Parameter to what the value would be in Banner. Finally, you may set the Parameter Prompt in JAMS to match the Banner parameter name.

The example below shows how the above Banner Parameters would look when set on a JAMS Job:



You may also specify multiple values for a single parameter by separating the values with pipe characters "|".

For example, if parameter 01 had a value of

```
|Val1|Val2|Val3|
```

JAMS would send the following to Banner:

01 = Val1

01 = Val2

01 = Val3

V2 Banner Parameter 88

The Parameter 88 setup was updated with the V2 Banner Integration to resemble other Banner parameters.

For example, a value of |One,Federal|Two,Grant is passed to Banner as the following:

88 = |One,Federal|Two,Grant|

You can define the parameter as 88 and then the value will contain a | delimited name,value pair.

V1 Banner Parameter 88

Parameter 88 is special as it can include a "label" before each value. JAMS removes the value and then inserts it into the 88 parameter. A value of "This,That" is passed to Banner as "88This = That".

For example, a value of |One,Federal|Two,Grant is passed to Banner as the following:

88One = Federal

88Two = Grant

You can also define the parameter as "88This" and then the value only needs to contain "That".

Dynamics AX Integration Pack

Configuring the .NET Business Connector

To ensure JAMS will run Dynamics AX Jobs, a Local Client (Where JAMS will retrieve the Job List) and a Business Connector (Where JAMS will run Jobs) must be properly configured.

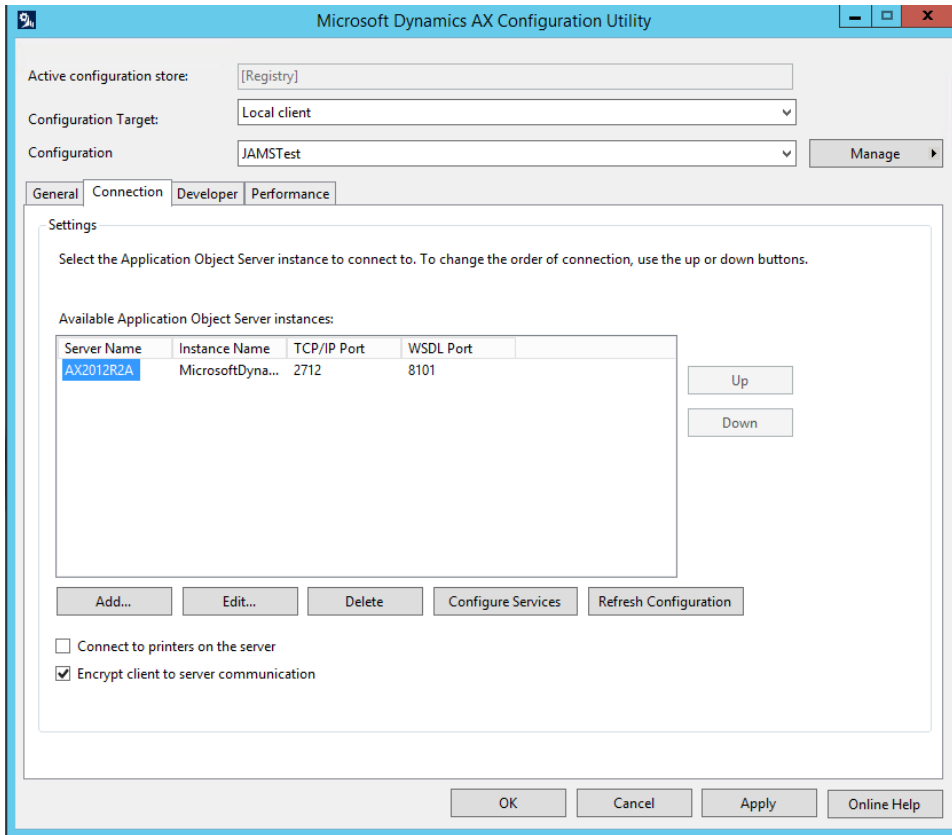
Begin by opening Microsoft Dynamics

- Open the Start Menu
- Navigate to Administrative Tools
- Open Microsoft Dynamics AX Configuration

NOTE: To properly configure Dynamics AX with JAMS, ensure Dynamics AX and the installed JAMS Add-on are for the same MS Dynamics AX version. Installers for Dynamics AX versions can be found on the [JAMS Support](#) site.

Configure the Local Client

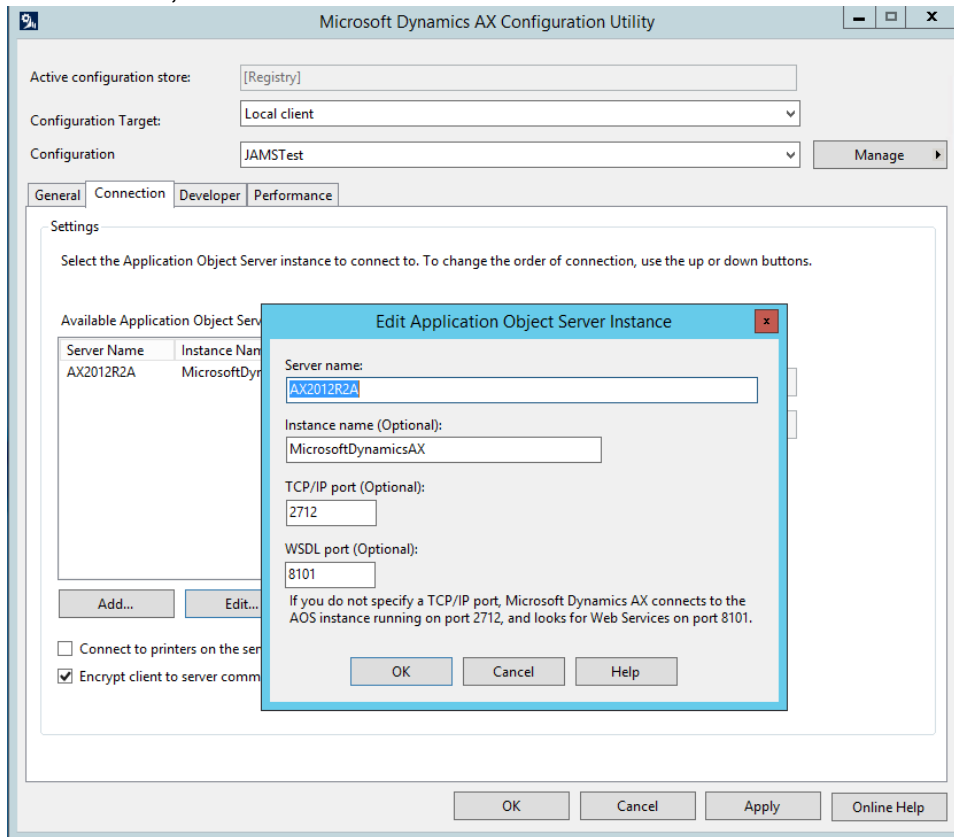
- Set the Configuration Target to "Local Client" using the drop-down menu.
- In the Connection tab, verify the Application Object Server and Instance Name are correct.



If the server name cannot be changed, a new configuration must be created.

- Click the **Manage** button, then select **Create Configuration**.
- In the dialog box, give the configuration a name, select **Active Configuration**, then click **OK**.
- Highlight the Server Name in the list of available servers and click **Edit**.

- In the dialog box that appears, fill in the Server Name, Instance Name, TCP Port and WSDL Port, then click **OK**.



- Click **Apply**.
- Click **Refresh Configuration**.
- Click **OK** to save the Configuration.

NOTE: You will need the TCP/IP and WSDL ports open from the machine running the JAMS client to the AOS server in order to create Dynamics AX jobs.

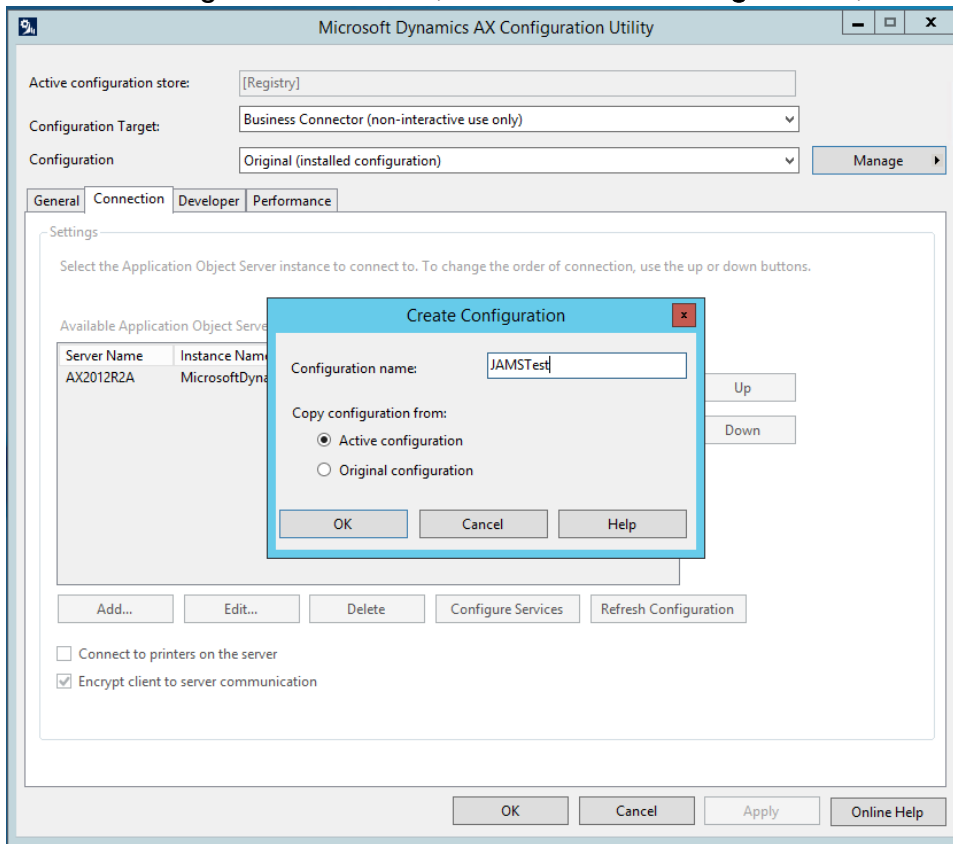
Configure the Business Connector

The Business Connector Configuration Target is the AOS where the JAMS job will connect to when the job actually runs.

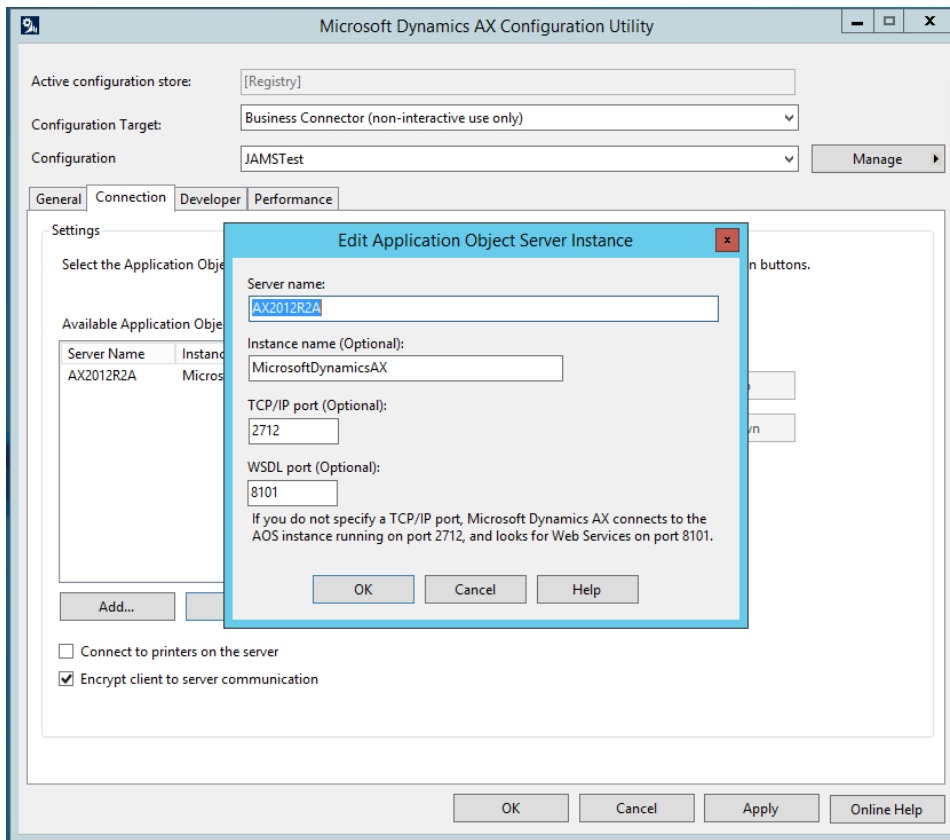
- Set the Configuration target to "Business Connector" using the drop down menu.
- In the Connection tab, verify the Application Object Server is correct and that the Instance Name is correct.

If the server name cannot be changed, a new configuration must be created.

- Click the **Manage** button, then select **Create configuration**
- Give the configuration a name, select **Active Configuration**, then click **OK**



- Highlight the **Server Name** in the list of available servers and click **Edit**.
- In the dialog box that appears, fill in the **Server Name**, **Instance Name**, **TCP Port** and **WSDL Port**, then click **OK**.



- Click **Apply**.
- Click **Refresh Configuration**.
- Click **OK** to save the configuration.

NOTE: You will need these ports open from the machine running the JAMS client to the AOS server in order to create Dynamics AX jobs.

Creating a User Credential for Dynamics AX Job Execution

JAMS natively supports automation using many leading business applications, including Microsoft Dynamics AX solutions. Integration Packs for these and other preferred applications are selected during the JAMS installation process. Once the integration is enabled, users can create, manage, deploy and monitor almost any kind of Dynamics AX job.

The JAMS Integration Pack for Microsoft Dynamics AX allows users to initiate Dynamics AX processes and take advantage of monitoring and report distribution capabilities using JAMS scheduling software.

When the integration runs, it initiates communication with and instructs Dynamics AX to schedule, execute AX jobs, batch jobs and reports as configured in JAMS. The process can then be monitored, canceled and restarted in either Dynamics AX or JAMS.

Follow the steps below to quickly add new JAMS user credentials.

1. Open the Credential Definitions view from the JAMS Menu.
2. Click the **Add** button on the Control Bar.
3. The JAMS Credential definition dialog opens.
4. Set the Credential Name, Logon As, Password, and Password Confirmation values.

NOTE: The user defined here must have the authority to use the .NET Business Connector.

A Microsoft Dynamics AX User is only allowed to use the .NET Business Connector if they satisfy one of the following Criteria:

- User is assigned to System Administrator Role
- User is assigned to the Business Connector Role
- User is assigned to a role with the privileges to execute .NET Business Connector Operations

5. Click **OK**. By default, JAMS will open the full Credential Definition dialog after the Credential definition is initially saved.
6. On the Security tab of the Credential dialog, Add the **ACE** for any user that is already set up in AX.
This user must have login rights as well as rights to execute items that will be added in later JAMS jobs. This may require setting up additional users based on their access rights in the AX product.
7. **Save and Close** the Credential definition.

Setting up the AX Job

Users must define and enter specific attributes in order to successfully run a job in JAMS.

Once a job has been properly defined it can either:

1. Be submitted to run on demand or . . .
2. Set up to run automatically using a schedule.

NOTE: A single JAMS Job can execute many other jobs. A schedule can exist on different machines to allow for separate management tasks and eliminate single points of failure.

Creating a New Folder

The first step is to create a new folder or alternatively use an existing folder directory to locate similar JAMS jobs.

NOTE: Creating new or using existing folders in JAMS is optional.

To create a new JAMS definitions folder follow the steps below:

1. Open the Definitions view from the JAMS Menu.
2. Right-click the desired Parent Folder in the Folder Navigator, then select **Add Folder** from the drop-down list.
3. The Add a new Folder dialog will appear.
4. Give the new folder a Name, then click **OK** to save the Folder definition.
5. Once the folder is defined, right-click on the folder and select **Properties** from the drop-down list to access the full Folder Definition dialog.
6. Define additional Elements, Parameters, Security, and Properties on the folder as desired, then click **Save and Close**.

Adding a new Dynamics AX Job using the JAMS Definition Wizard

After creating a folder, you can quickly set up a Dynamics AX job using the Add a New JAMS Job Definition dialog.

Follow the steps below to create and define a JAMS/AX job:

1. Select the desired folder for the Dynamics AX Job, then ensure the Job Definitions tab is selected.
2. Click the **Add** button from the Control Bar to open the Add a New JAMS Job Definition dialog.
3. In the dialog, give the new Job a Name, Description (optional), and Execution Method. In this case, select MSDAX2012Job to create a Microsoft Dynamics AX Job.
4. By default, the full Job Definition dialog will open when the Job is initially saved. Click **OK**.
5. The Job Definition dialog will open.
6. Define Elements, Parameters, Security, Properties, and Documentation as desired. Ensure "Execute As" has been set to the Dynamics AX user Credential.
7. On the Source tab, enter the appropriate values for the given Dynamics Job.
Company: Select the AX company ID for this job.
Job Type: Choose the appropriate radio button.

Job: Choose from a list of defined jobs in the AX AOT

Report: Select from a list of reports that are on an AX menu. This type of job can have many defined parameters.

Batch Job: Choose from the list of batch Jobs (by name) that have been created in AX. A batch job contains all the pertinent information to execute on its own.

8. **Save and Close** the job definition.

Available AX Job Parameters

Generic to all Job Types

Include MS Dynamics Log

Debug

Language: Specify the language setting to be used when logging into AX Job.

ObjectServer: Specify the object server setting to be used when logging into AX Job.

Configuration: Specify the object server setting to be used when logging into AX Job.

Report Job Type Specific Parameters

Query: Specify the name of a query to be applied to the report before rendering it. This Query is set up in AX.

Destination: Specify where the report should be sent once rendered.

Possible Values

Value	Meaning
1	Printer
3	Email
4	File
Printer	Printer
Email	Email
File	File

Default value:

1. Printer

FileFormat: Specify the format of the file being exported by processing a report.

Possible Values

Value	Meaning
ASCII	Ascii
RTF	Rich Text Format
HTML	HTML
PDF	PDF
PDFEMBEDFONTS	PDF with embedded fonts
TEXTUTF8	Text in UTF8 encoding
1	Ascii
2	Rich Text Format
3	HTML
4	PDF
5	PDF with embedded fonts
6	Text in UTF8 encoding

Default value:

1. Ascii

FileName: Specify the name of the file the report will be exported to when the destination is set to File.

PrinterName: Specify the printer name report will be printed when the destination is set to Printer.

FirstPage:

Specify

Default

0

LastPage:

Specify

Default

0

Copies:

Specify

Default

0

Subject: Specify the subject of an Email. Used when the destination is set to Email.

To: Enter an email address to send the report,

Job Specific Parameters

Parm: Specify a single parameter to be sent to a Job. Note the job in AX must be designed in such a way to accept a 'parm'

GoAnywhere MFT

The GoAnywhere Execution Method lets you use JAMS to run and monitor projects in GoAnywhere MFT for secure file transfers. The GoAnywhere MFT project should be defined before attempting to use JAMS to run the process. GoAnywhere MFT projects cannot be created or modified from within JAMS. Any values from GoAnywhere MFT cannot be passed back into JAMS.

The GoAnywhere MFT project can contain variables. If you want to override the value of a variable on your GoAnywhere project, add a parameter on the JAMS Job with the same name and set the value. You can also use JAMS variables when you set up the Job. Any field except for the GoAnywhere User can use a JAMS variable with the format {variable}.

NOTE: Ensure the JAMS Scheduler and JAMS Client have been updated to at minimum JAMS 7.7 to use this functionality.

NOTE: Active GoAnywhere MFT projects cannot be canceled from JAMS. You would need to cancel the Job from within GoAnywhere MFT.

Authentication

Two types of authentication are available:

- Basic - Enter the username and password for the account that is used to log in to GoAnywhere MFT.
- Token - Enter a Display Name and a valid user API key as the password for the Credential. The password is expected to be a valid Bearer Token that can be generated in GoAnywhere MFT. The Display Name is used only to identify the Credential in JAMS and is not connected to a GoAnywhere login.

Before You Begin

- Ensure you are running GoAnywhere MFT version 7.4.0.
- Ensure you have a Credential in JAMS that will run the Job. See [Working with User Credential Definitions](#) for more information.
- Ensure an Agent has been created if you want the Job to run on an Agent rather than the JAMS Scheduler. See [Agents in JAMS on page 57](#) for more information.
- Ensure you have the following information about the project:
 - The IP address or server name of the GoAnywhere server. It should have a format similar to `http://<insert IP address>:8000/goanywhere/rest/gacmd/v1`. For https, the format will be similar to `https://<insert IP address>:8001/goanywhere/rest/gacmd/v1`
 - The username/password or user API key value for authenticating with the GoAnywhere REST API V1. Using the API key is recommended.
 - The domain where the project is located.
 - The project name.
 - The path where the project is located.

Creating a Credential in JAMS for Authenticating with the GoAnywhere MFT

1. Click **Credentials** in the Shortcuts menu.
2. Click **+**.
3. In the Credential Name field, enter a Credential Name. This may be the name used when logging on with these Credentials.
4. In the Logon As field, enter a Logon As name if the name used when logging on is different than the Credential Name.
5. In the Enter Password and Re-Enter password fields, enter a password. If you plan to use Token authentication, enter the user API key in these fields.

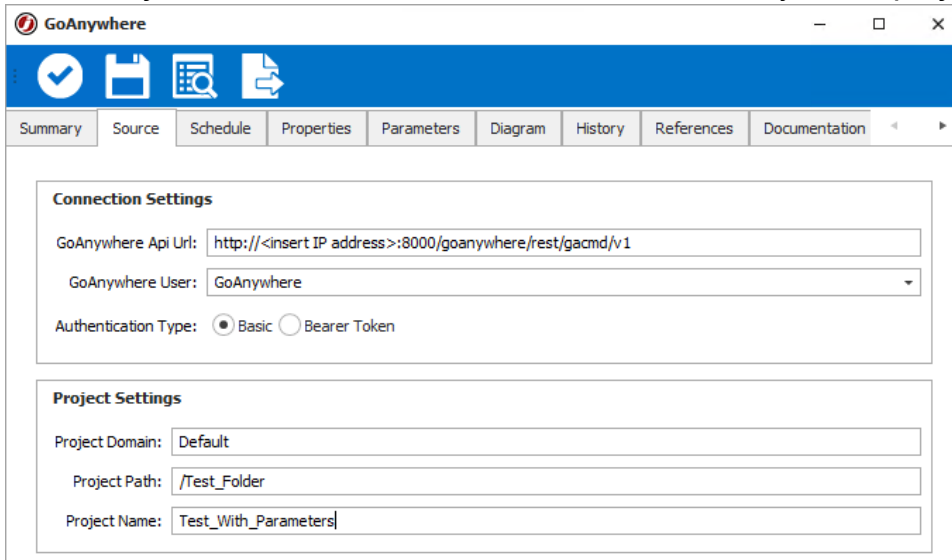
6. Click **Ok**.
7. If desired, enter a Description.
8. Click the **Security** tab.
9. Use the checkboxes to select the appropriate access types. Ensure the **Get Password** value is selected.
10. When you are finished, click **Save and Close**.

Creating a Job in JAMS to run GoAnywhere MFT Projects

To run the GoAnywhere MFT project, create a Job to connect to GoAnywhere MFT and run the specified project.

1. Click **Definitions** from the Shortcuts menu.
2. Click a Folder where the Job will be saved, or right-click a Folder and select Add Folder to add a new Folder.
3. Click **+** to add a new Job.
4. In the Name field, enter a name for the Job.
5. In the Execution Method field, select GoAnywhere Project.
6. Click **Ok**.
7. Click the **Source** tab.
8. In the GoAnywhere Server field, enter the IP address or server name of the GoAnywhere server. The port number may be different depending on your configuration, but it should have a format similar to http://<insert IP address>:8000/goanywhere/rest/gacmd/v1. For https, the format will be similar to https://<insert IP address>:8001/goanywhere/rest/gacmd/v1.
9. In the GoAnywhere User field, select the JAMS Credential that will be used to connect to GoAnywhere.
10. For the Authentication type, select Basic or Bearer Token based on how the JAMS Credential was created.
11. In the Project Domain field, enter the name of the domain. The default value is "Default".
12. In the Project Path field, enter the file path in the selected domain in GoAnywhere to the project that will run.

13. In the Project Name field, enter the name of the GoAnywhere project that will run.



The screenshot shows the GoAnywhere application window with the 'Project Settings' tab selected. The 'Connection Settings' section includes fields for 'GoAnywhere Api Url' (http://<insert IP address>:8000/goanywhere/rest/gacmd/v1), 'GoAnywhere User' (GoAnywhere), and 'Authentication Type' (Basic selected). The 'Project Settings' section includes fields for 'Project Domain' (Default), 'Project Path' (/Test_Folder), and 'Project Name' (Test_With_Parameters).

14. Click the **Schedule** tab and set any schedule information for when the Job will run. See [Schedules](#) for more information.
15. Click the **Properties** tab.
16. In the Execute As field, select the JAMS Credential that will be used to run the Job.
17. In the Agent field, select an Agent from the drop-down, if the Job will not run on the JAMS Scheduler.
18. Click the **Parameters** tab.
19. If you want to override the value of a variable on your GoAnywhere project, add a parameter on the JAMS Job with the same name and set the value.
20. Click **Save and Close**.

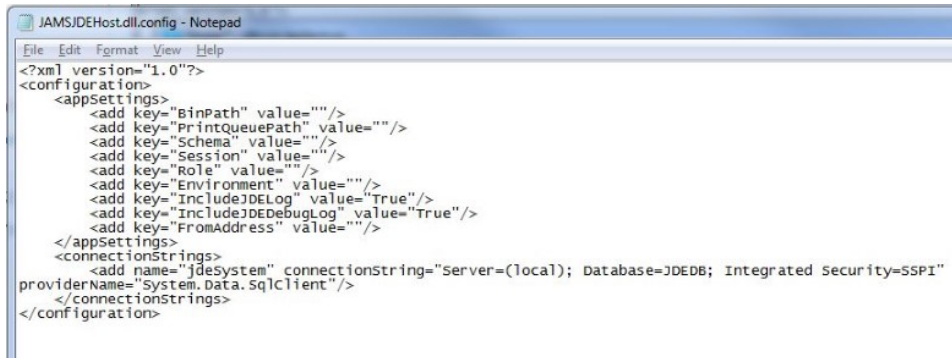
After you create the Job, you can manually submit it, if needed. Otherwise, it will be submitted based on the specified schedule. You can then view its progress by selecting Monitor from the Shortcuts menu.

Automating a JD Edwards Job in JAMS

There are a three prerequisite steps that must be undertaken before users can automate a JD Edwards job in JAMS.

- You must first install the JDE Integration Pack on the JAMS Scheduler and on the remote machine where runubexml.exe resides.
- Also, the JAMS Agent must also be installed on the remote machine where runubexml.exe resides.

- On the root installation directory (C:\Program Files\MVPSI\JAMS\Scheduler) of your JAMS installation, as well as the root installation folder of the JAMS agent (C:\Program Files\MVPSI\JAMS\Agent) locate the file named **JAMSJDEHost.dll.config.sample** and ensure the proper connection string for the JDE database has been defined. The following example utilizes a Microsoft SQL instance:



NOTE: If the JD Edwards system is configured with an alias for the JDE Batch Server in the application, a key must be added to the JAMSJDEHost.dll.config file.

```
<add key="ExeHost" value="MyJDEAlias" />
```

Additionally, if running JDE with an Oracle backend users can include an Oracle-based connection string as shown in the example below.

```
<connectionStrings>
  <add name="jdeSystem" connectionString="Data Source=E1PROD;Use
  Id=SQLSysUser;
  Password=Password;" providerName="system.data.oracleclient"/>
</connectionStrings>
```

Once the requirements for the connection string have been satisfied, make sure to rename the file upon saving to JAMSJDEHost.dll.config to allow JAMS to pick up and read the configuration settings.

The binPath value must be set which is the location of runubexml.exe, as shown in the example below: Example: E:\JDEdwards\E910\system\bin32

The Schema value must be set based on the given schema for the Master Control Table. For example, if the Master Control Table's Schema is SVM920.F986110, set the Schema value in JAMSJDEHost.dll.config to be SVM920.


```
<add key="Schema" value="SVM920"/>
```

NOTE: This file must be adjusted on the Remote Agent side if JD Edwards jobs are running on a JAMS Remote Agent.

For the next step, create a new job definition and set the Execution Method as **JDEjob**. The job's source will display a fill-in-the-blank form requiring various inputs necessary to automate the JDE jobs.

Finally, add **Home Directory** and **Execute As** properties to the Job. Modify the Job's Properties to define a Home Directory other than the default System32. Utilizing a directory such as C:\JAMS_TEMP\ as well as defining an Execute AS User Account that has permissions to run this job on Windows is recommended (See [Defining RunAs User Accounts in JAMS](#)).

This is the user accessing the C:\JAMS_TEMP directory on the remote JDE batch server, as well as accessing the Master Control Database from within JDE if the Connection String is set for Integrated Security.

Informatica Cloud In JAMS

The InformaticaCloud Execution Method allows users to automate any existing Informatica Cloud task, using the Informatica REST API.

NOTE: If you are using JAMS V7.2.632 or older for the InformaticaCloud Execution Method, it is recommended that you upgrade the JAMS Client and JAMS Scheduler to the latest version to use the new optimization improvements. To edit Informatica Cloud Jobs, the JAMS Client must be upgraded if it connects to a JAMS Scheduler that has been upgraded to the latest version. After you upgrade, you will need to edit each existing Informatica Cloud Job as follows:

- Reselect your task in the Task drop-down on the Source tab. (Select a different task, and then select the original task.)
- Save the Job.

Installing the Informatica Cloud Integration Pack

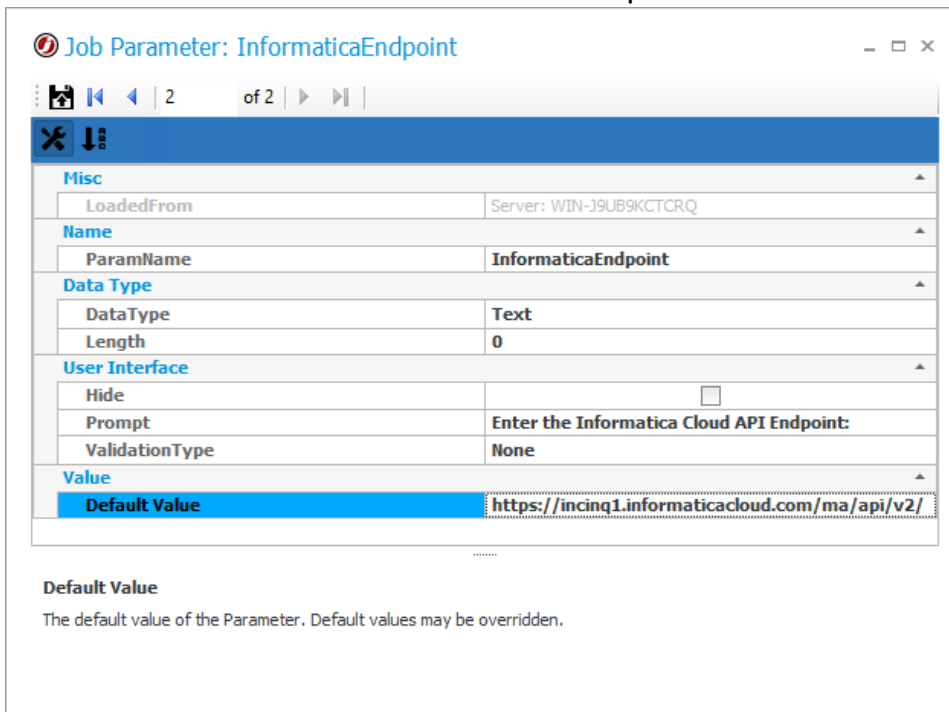
1. Run the JAMS installer.
2. Follow the prompts until you see the Integration Packs page.
3. Select the **Informatica Cloud Integration Pack**.

4. Click **Next**.
5. Follow the remaining prompts in the installer to complete the installation.

NOTE: Clearing the checkboxes for components will uninstall those components.

Configuring the Informatica Cloud Execution Method

1. Run the JAMS Client as an administrator.
2. Select the **Execution Methods** shortcut from the JAMS Shortcuts menu.
3. Double-click the **InformaticaCloud** Execution Method to open the Execution Method dialog.
4. Select the **Parameters** tab.
5. Double-click the **InformaticaEndpoint** parameter to open the Parameters dialog.
6. Set the Default Value to the InformaticaEndpoint value for the environment.



Job Parameter: InformaticaEndpoint

2 of 2

Misc

LoadedFrom: Server: WIN-J9UB9KCTCRQ

Name

ParamName: InformaticaEndpoint

Data Type

DataType: Text

Length: 0

User Interface

Hide: ☐

Prompt: Enter the Informatica Cloud API Endpoint:

ValidationType: None

Value

Default Value: <https://incinq1.informaticacloud.com/ma/api/v2/>

Default Value

The default value of the Parameter. Default values may be overridden.

7. Click **Save and Close** on the Parameters dialog.
8. Click **Save and Close** on the Execution Method dialog.

NOTE: The user running Informatica Cloud Jobs must have full access to the Informatica Cloud.

Creating an Informatica Cloud Job in JAMS

1. On the Home screen, click **Create a Job**.
2. Select **JAMS Integration Jobs**.
3. Click **Create an Informatica Cloud Task job**.
4. On the **Add a New JAMS Job Definition** screen, do the following:
 1. In the Folder field, select a folder where the Job will be saved.
 2. In the Name field, enter a name for the Job.
 3. Optional - In the Description field, select a date for the Job to start.
 4. In the Execution Method field, ensure InformaticaCloud is selected.
 5. In the Edit this job definition after adding field, ensure the checkbox is selected.
 6. Optional - In the Scheduled Date field, select a date for the Job to start.
 7. Optional - In the Calendar field, select the calendar to use.
 8. Optional - In the Scheduled Time field, select the time for the Job to start.
 9. Click **Ok**.
5. Click the **Source** tab.
6. Set the User and ensure it has access to the Informatica Cloud environment.
7. Set the Task Type for the relevant Informatica Cloud Task. The Task list will automatically populate with all of the Tasks of the selected Type.
8. Select the relevant Task using the drop-down list. The drop-down list is organized by folder and then by the tasks within that folder.
9. Define the Schedule, Properties, Parameters, Documentation, and Security Items as needed.
10. Click **Save and Close**.

Informatica Properties

Property	Description
Informatica Credential	Set the Informatica Cloud user to run the Job. This is a Credential that is created within JAMS.
Informatica Retry Count	Set the maximum number of automatic retry attempts for Informatica Cloud REST calls.
Informatica Retry Interval	Set the interval between retry attempts when an Informatica Job fails.

Property	Description
Informatica Polling Interval	Set the interval between automatic retry attempts to get information and status about Jobs running in Informatica. You may need to adjust the polling interval if you are running many long Jobs. If you cancel a Job in JAMS while the Job is still running in Informatica, the Job will be canceled after the polling interval time. The default value is 3 seconds.

Micro Focus in JAMS

Installing the Micro Focus Add-on

To work with Micro Focus in JAMS, users must install the JAMS Add-on for MicroFocus on the JAMS Scheduler Server, and must install a JAMS Agent and the JAMS Add-on for MicroFocus on the Micro Focus Server. If the JAMS Scheduler and Micro Focus are installed on the same server, an Agent is not required.

Install the Micro Focus Add-on on the JAMS Scheduler Server

- On the JAMS Scheduler Server, run the JAMS Installer.
- On the Feature Selection step, ensure the **JAMS Add-on for MicroFocus** component is checked in addition to the required components. Do not un-check any components unless those components should be uninstalled.
- Finish the installation.

NOTE: Un Checking components in the Feature Selection step will uninstall those components.

NOTE: The JAMS Server and Micro Focus Server should be running the same version of JAMS.

Install the JAMS Add-on for MicroFocus on the Micro Focus Server

- On the Micro Focus Server, run the JAMS Installer.
- On the Feature Selection step, ensure the **JAMS Agent** and **JAMS Add-on for MicroFocus** components are checked.
- Finish the installation.

Deploy a JAMS Agent to the Micro Focus Server

1. Start the JAMS Client as an Administrator.
2. Select the **Agents** shortcut from the menu.
3. In the Agent Definitions view, select **Add** from the Control Bar to open the Add an Agent Definition dialog.
4. In the dialog, give the new Agent a Name, select the Agent Type and Agent Platform, and if desired, a Description.
5. Click **OK** to save the initial Agent definition. By default, JAMS will automatically open the full Agent Definition dialog after the Agent is initially saved.
6. Set Agent Properties in the Agent Definition dialog, then **Save and Close** the Agent Definition dialog.
7. In the Agent Definition view, select the Agent to be deployed.
8. Select **Deploy** from the Control bar to open the Deploy JAMS Agent dialog.
9. In the dialog, select the **JAMS User** that will be used when deploying the JAMS Agent.
This user should have all necessary permissions to the defined Micro Focus Agent.
10. Click **OK**.

Configuring the Micro Focus Execution Method

1. Run the JAMS Client as an Administrator.
2. Select the **Execution Methods** shortcut from the Shortcuts Menu.
3. Double-click the **MFJob** Execution Method to open the Properties dialog, then select the Parameters Tab.

Method Name: MFJob

Execution Method Parameters Template Properties Job Properties

Parameter Name	Type	Prompt	Default Value
JAMSTraceLevel	Text	Select the JAMS Trace Level:	Off
MFServer	Text	Enter the MicroFocus server:	MyMFServer
MFBINPort	Integer	Enter the MFBINP listener port:	9122
MFSubmitType	Text	Select the submit type:	JES2

Move Up Move Down Add... Properties... Remove

- Use the **Properties** button to open the Parameter Properties dialog. Configure Micro Focus parameters as needed for the given Micro Focus environment.

Parameter	Description
MFBINPort	Enter the MFBINP listener port. The default in JAMS is 9122. Listener configurations are stored in the Micro Focus listener configuration file: seelistener.exe.config. If a listener port must be created for JAMS, reference the Micro Focus documentation .
MFServer	Enter the name of the Micro Focus Server.
MFSubmitType	Set the relevant Submit type from JES2 (Job Entry Subsystem 2), JES3, or VSE. The default is JES2.

- Save and Close** the Parameters dialog, then **Save and Close** the MFJob Execution Method.

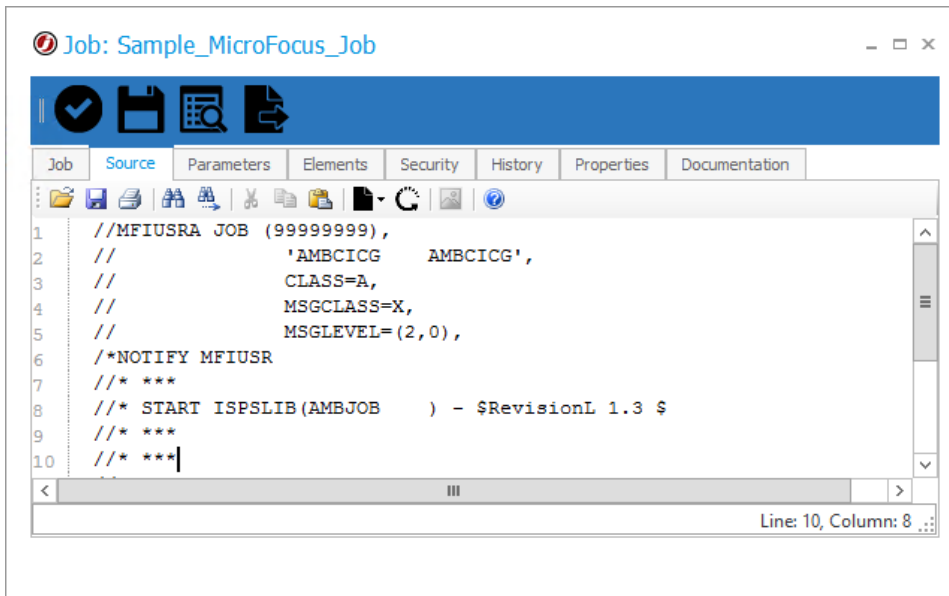
Creating a Micro Focus Job

- Select the desired folder for the MicroFocus Job, then ensure the Job Definitions tab is selected.
- Click the **Add** button from the Control Bar to open the Add a New JAMS Job Definition dialog.
- In the dialog, give the new Job a Name, Description (optional), and Execution Method. In this case, select **MFJob** to create a Micro Focus Job.
- By default, the full Job Definition dialog will open when the Job is initially saved. Click **OK**.

5. The Job Definition dialog will open.
6. Define Elements, Parameters, Security, Properties, and Documentation as desired.

NOTE: An Execute As property should be configured with a user that has access to the Micro Focus Server.

7. Paste or write the desired code in the Source.



8. **Save and Close** the Job definition.

NOTE: The JCL Code in JAMS should match the syntax of Micro Focus code.

For more information on Micro Focus, consult the [Official Micro Focus Documentation](#).

The JAMS Management Pack for MS SCOM

The JAMS Management Pack provides monitoring of JAMS components and services using Microsoft SCOM 2007R2, 2012 and 2012 R2. The monitoring includes availability, health and performance data collection. In addition to these monitoring capabilities, the Management Pack supports Agent tasks and Dashboard views. The Management Pack supports monitoring of JAMS on the Windows operating system only. Support for other operating systems may be added at a later date.

Preparing to Import the Management Pack

To ensure that target JAMS computers are managed by the SCOM Agents (preferably using LocalSystem account), the account used by the SCOM Agent must have adequate

permissions to access JAMS program directory, the system registry and access JAMS data via its PowerShell provider.

NOTE: You can use local accounts on managed computers and/or dedicated domain accounts to execute Management Pack rules and monitors; however, you will also need to configure the JAMS 'Run As' profile account using SCOM's operations console as outlined below.

It is recommended the target computer with the JAMS Scheduler role have the PowerShell feature enabled or installed. Please refer to the JAMS Installation Guide for additional information. It is also recommended that you install the JAMS Instrumentation Integration for each managed JAMS Scheduler machine in order to reduce overhead and improve functionality of the Management Pack.

Importing the Management Pack

The Management Pack consists of a single file: MVP.JAMS.mp

For general import instructions go to the article: <http://technet.microsoft.com/en-us/library/cc974494.aspx>

1. Log on to the computer using an account that is a member of the Operations Manager Administrators role within the Operations Manager management group.
2. From the Operations console, click **Administration**. When you run the Operations console on a computer that is not a management server, the 'Connect To Server' dialog box appears. In the Server name text box type the name of the management server that you want the Operations console to connect to.
3. Right-click the Management Packs node and then click Import Management Packs.
4. The Import Management Packs wizard opens. Click the **Add** button and then select Add from disk.
5. The Select Management Packs to import dialog box appears. Change to the directory that contains the JAMS Management Pack file. Select **MVP.JAMS** and then click **Open**.
6. On the Select Management Packs page the management pack that you selected for import should be listed. Choose **Import**.
7. The Import Management Packs page appears and displays the import progress. If there is a problem at any stage of the import process select the management pack to view the status details. Click **Close**.

Discovery

After importing the JAMS Management Pack, SCOM runs a discovery process on all managed Windows computers in order to locate the JAMS installations. By default SCOM discovers all JAMS installations on Windows. No additional configuration should be necessary for most customer environments.

Customizing the Management Pack

The JAMS Management Pack is 'sealed', preventing any setting changes in the original Management Pack file. However, you can still perform and save customizations such as overrides for monitors, alert rules and views, and save them to a different management pack.

By default SCOM saves all customizations to the Default Management Pack. As a best practice you should instead create a new management pack for each sealed management pack you want to customize.

For more information about management pack customizations and the Default Management Pack, link to the article: [About Management Packs in Operations Manager 2007](#)

The Management Pack includes the following customizations:

- JAMS Scheduler Performance Monitor. Note that there are 2 monitors targeting the JAMS Scheduler with and without the Instrumentation Integration installed (JAMS Instrumented Scheduler and JAMS Non-Instrumented Scheduler). Both monitors have identical thresholds for the number of currently executed or pending Job with a specific status.

You can disable these monitors and/or provide the overrides for the frequency as well as warning and error thresholds. The screenshot below illustrates the Scheduler Performance Monitor parameters that can be overridden using the SCOM Operations Console.

Override Properties

Monitor name: JAMS Scheduler Performance Monitor
 Category: Custom
 Overrides target: Class: JAMS Non Instrumented Scheduler

Override-controlled parameters:

	Override	Parameter Name	Parameter Type	Default Value	Override Value	Effective Value	Change Status
	<input type="checkbox"/>	Alert On State	Enumeration	The monitor ...	The monitor is...	The monitor is...	[No change]
	<input type="checkbox"/>	Alert Priority	Enumeration	Medium	Medium	Medium	[No change]
	<input type="checkbox"/>	Alert severity	Enumeration	Match monit...	Match monito...	Match monitor...	[No change]
	<input type="checkbox"/>	Auto-Resolve Alert	Boolean	True	True	True	[No change]
▶	<input checked="" type="checkbox"/>	Enabled	Boolean	True	True	True	[No change]
	<input type="checkbox"/>	ExecutingErrorThre...	Integer	65535	65535	65535	[No change]
	<input type="checkbox"/>	ExecutingWarningT...	Integer	32767	32767	32767	[No change]
	<input type="checkbox"/>	Generates Alert	Boolean	True	True	True	[No change]
	<input type="checkbox"/>	HeldErrorThreshold	Integer	65535	65535	65535	[No change]
	<input type="checkbox"/>	HeldWarningThres...	Integer	32767	32767	32767	[No change]
	<input checked="" type="checkbox"/>	IntervalSeconds	Integer	600	60	60	[No change]
	<input type="checkbox"/>	PendingErrorThresh...	Integer	1000	1000	1000	[No change]
	<input type="checkbox"/>	PendingWarningTh...	Integer	500	500	500	[No change]
	<input checked="" type="checkbox"/>	RunawayErrorThres...	Integer	50	2	2	[No change]
	<input checked="" type="checkbox"/>	RunawayWarningT...	Integer	10	1	1	[No change]
	<input type="checkbox"/>	ScheduledErrorThr...	Integer	65535	65535	65535	[No change]
	<input type="checkbox"/>	ScheduledWarning...	Integer	32767	32767	32767	[No change]
	<input type="checkbox"/>	StalledErrorThreshold	Integer	50	50	50	[No change]
	<input type="checkbox"/>	StalledWarningThre...	Integer	10	10	10	[No change]
	<input type="checkbox"/>	Sync Time	String				[No change]
	<input type="checkbox"/>	TimedErrorThreshold	Integer	65535	65535	65535	[No change]
	<input type="checkbox"/>	TimedWarningThre...	Integer	32767	32767	32767	[No change]

- There are 2 alert rules that are defined for the Instrumented Scheduler class:
 - JAMS Scheduler Execution Warning
 - JAMS Scheduler Execution Error

These rules can be enabled or overridden to produce Scheduler alerts based on the level of events logged to MVPSI JAMS/Operational by the JAMS Instrumentation Integrations.

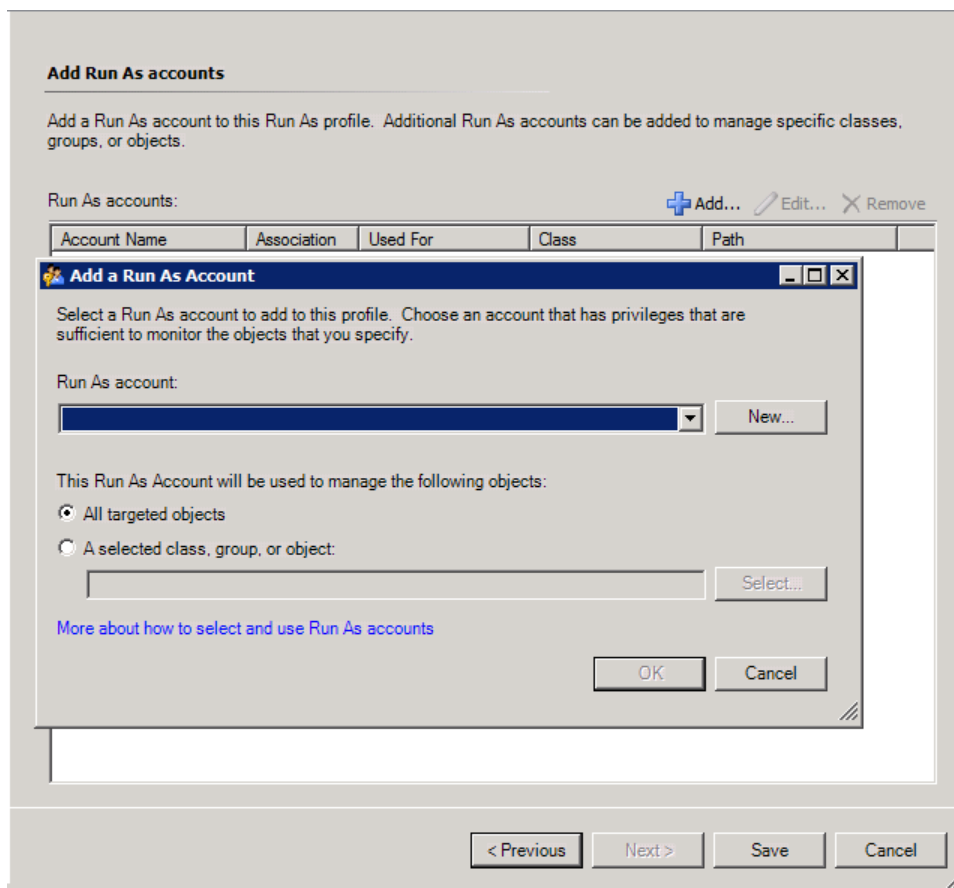
- JAMS Scheduler (performance) collection rules can be disabled or the frequency (IntervalSeconds) can be adjusted.

NOTE: There are separate set of rules for Instrumented and Non-Instrumented Scheduler classes.

- New monitors and rules can be easily added using new performance counters and/or ETW events implemented in the JAMS Instrumentation Integrations.

NOTE: Corresponding providers are available only on Scheduler machines that have Instrumentation Integrations installed. These JAMS Scheduler instances are discovered as instances of 'JAMS Instrumented Scheduler' class derived from 'JAMS Scheduler' class.

The JAMS Management Pack creates 'Run As' profile named 'Account for JAMS Collection Rules' as shown on the screenshot below. You can associate one or more SCOM accounts with this profile to define local or domain accounts that should be used by the management pack rules and monitors on each computer. Refer to SCOM documentation for further information on 'Run As' configuration.

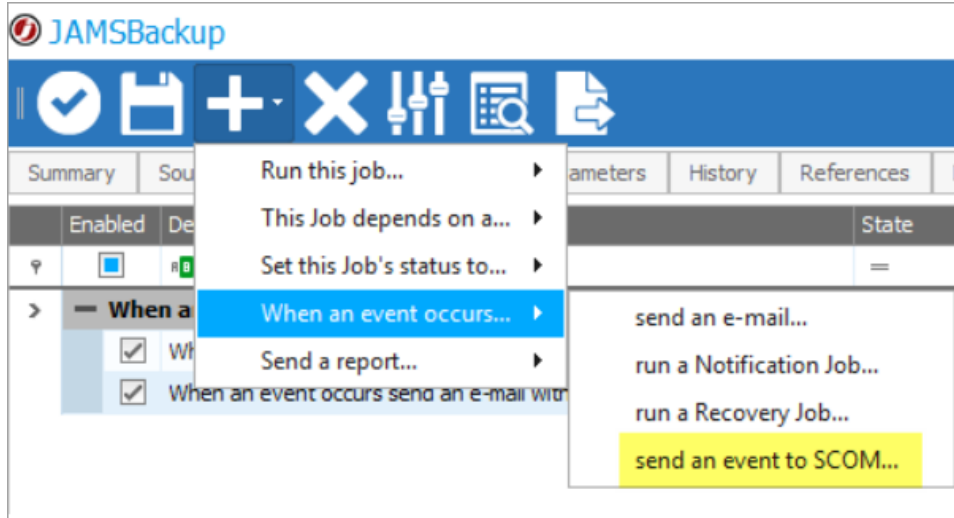


Removing the Management Pack

For additional information on removing the Management Pack, go to the article: <http://technet.microsoft.com/en-us/library/cc974489.aspx>

Configuring a JAMS Scheduler to Send Alerts

1. Open C:\Program Files\MVPSI\JAMS\Scheduler.
2. Run the JAMSDBA.exe.
3. Execute the following command: **Load/Assembly=JAMSSCOMEventHandler**
4. Restart the JAMS Scheduler service. The system adds the new SCOM Item as shown below.



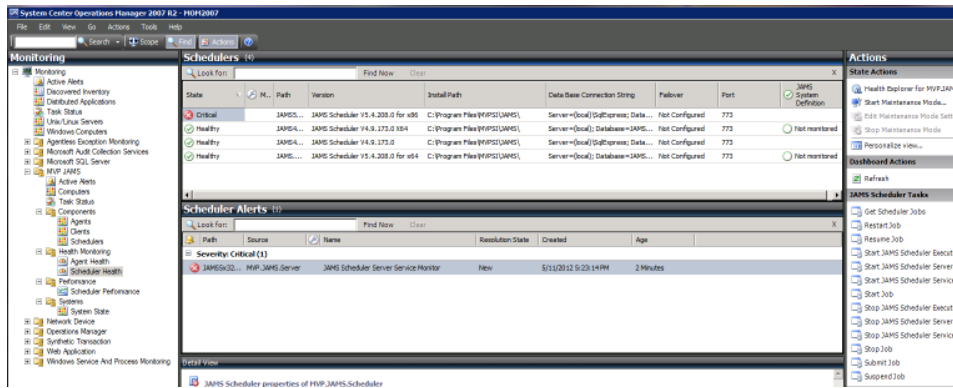
5. Create a Job.
6. Go to the Schedule tab.
7. Add the new SCOM Item.

JAMS looks for the following parameters before sending an alert to SCOM. These parameters can originate from the Job's execution method, the Job itself, or the folder the Job is located in.

- **SCOMPRIORITY:** Sets the priority of the alert. This parameter should be one of three integer values, with 0 = low, 1 = medium, and 2 = high.
- **SCOMSeverity:** Sets the severity of the alert. The parameter also should include one of three values, with 0 = informational, 1 = warning, and 2 = critical.
- **SCOMCustomField0** through **SCOMCustomField9:** These fields are passed to SCOM and intended for end-user customizations. The values are strings with JAMS performing symbol substitution, allowing you to reference any JAMS property using the standard <<JAMS.Name>> syntax.

View Information in the Operations Manager Console

Once imported, the JAMS Management Pack creates a root folder, MVP JAMS, that can be accessed via the SCOM Operations Console Monitoring pane using the following subfolders and views:



These Folders and Views display different aspects of monitoring data.

- **Root:** MVP JAMS folder
- **Alerts:** Alert View for all JAMS Alerts
- **Computer:** State view for all computers within the JAMS computer group.
- **Tasks:** Task view of all JAMS Scheduler and Agent tasks queued via the SCOM Operations console to run on SCOM Agent computer.
- **Components folder**
 - Scheduler State View
 - Agent State View
 - Client State View
- **Health Monitoring Folder**
 - Agent Health dashboard view displays JAMS Agent State View and Alerts.
 - Scheduler Health dashboard view displays JAMS Scheduler state views and alerts.
- **Performance Folder**
 - Scheduler Performance View

JAMS Management Pack Agent Tasks

The JAMS Management Pack provides a set of admin tasks that can be queued to run on a SCOM Agent machine; for example, to start/stop the JAMS component services, query or submit JAMS Jobs.

Some of these tasks are available via the Management Pack Alerts Knowledge links, while you can access JAMS Agent or Scheduler tasks via the Operations console Actions pane in the Management Pack Agents and Schedulers status views.

The following tasks are available for JAMS Agents:

- Start JAMS Agent Service
- Stop JAMS Agent Service

The following task are available for the JAMS Scheduler:

- Get Scheduler Jobs
- Restart Jobs
- Resume Jobs
- Start JAMS Scheduler Executor Service
- Start JAMS Scheduler Server Service
- Start JAMS Scheduler Service
- Start Job
- Stop JAMS Scheduler Executor Service
- Stop JAMS Scheduler Server Service
- Stop JAMS Scheduler Service
- Stop Job
- Submit Job
- Suspend Job

Troubleshooting the Management Pack

If the Scheduler Performance Monitor and/or collection and alert rules are not working it may indicate that PowerShell is not available or JAMS PowerShell module is not installed on the target system. Please check the SCOM Active Alerts view and the SCOM application event log on the management server or management computers for the alerts or logged event notifications indicating PowerShell is not properly installed on the target computer or script execution errors such as insufficient Windows or JAMS permissions assigned to accounts used to run the management pack workflows.

By default these workflows use the account assigned to the SCOM Agent (such as LocalSystem), but you can customize the 'Run As' settings as described in the topic: [*The JAMS Management Pack for MS SCOM on page 119.*](#)

Troubleshooting Alerts

If you are not receiving alerts from JAMS to SCOM, check the following:

- Review the four steps made in the first section of this topic: [Configuring a JAMS Scheduler to Send Alerts](#).
- Check the security permissions on the jamstoscom msmq private queue. Make sure that your SCOM activity user or Run As user has full control over the queue.
- If you don't have a Run As account configured for JAMS in SCOM, the default Activity user(s) needs to have full control over the **jamstoscom** queue.
- Restart the JAMS Server after installing the SCOM Monitoring Agent Service.

NeoBatch In JAMS

Installing the NeoBatch Integration Pack

To work with NeoBatch in JAMS, complete the following installations, then deploy an Agent to the NeoBatch Server.

On the NeoBatch Server:

- NeoBatch Add-on
- JAMS Agent

NOTE: The NeoBatch Server must be running NeoBatch Server Edition. Developer or Client editions of NeoBatch will not function with JAMS.

On the JAMS Clients that need to edit NeoBatch Jobs:

- NeoBatch Add-on
- JAMS Client

NOTE: If the JAMS Scheduler is installed on the NeoBatch Server, an Agent is not required.

Install the NeoBatch Integration Pack and JAMS Agent on the NeoBatch Server

- Run the JAMS Installer.
- On the Feature Selection step, check the **JAMS Agent** and **JAMS Add-on for NeoBatch** components.

NOTE: Un Checking components will uninstall those components.

NOTE: The JAMS Server and NeoBatch Server should be running the same version of JAMS.

Install the JAMS Add-on for NeoBatch on JAMS Clients

- Run the JAMS Installer.
- On the Feature Selection step, check the **JAMS Client** and **JAMS Add-on for NeoBatch** components.

NOTE: Un Checking components will uninstall those components.

Deploy a JAMS Agent to the NeoBatch Server

1. Start the JAMS Client as an Administrator.
2. Select the **Agents** shortcut from the menu.
3. In the Agent Definitions view, select **Add** from the Control Bar to open the Add an Agent Definition dialog.
4. In the dialog, give the new Agent a Name, select the Agent Type and Agent Platform, and if desired, a Description.
5. Click **OK** to save the initial Agent definition. By default, JAMS will automatically open the full Agent Definition dialog after the Agent is initially saved.
6. Set Agent Properties in the Agent Definition dialog, then **Save and Close** the Agent Definition dialog.
7. In the Agent Definition view, select the Agent to be deployed.
8. Select **Deploy** from the Control bar to open the Deploy JAMS Agent dialog.
9. In the dialog, select the **JAMS User** that will be used when deploying the JAMS Agent.
This user should have all necessary permissions to the defined NeoBatch Agent.
10. Click **OK**.

Configuring NeoBatch Execution Methods

Before creating Jobs with NeoBatch in JAMS, users must configure the NeoBatch Execution Methods. This will allow JAMS to connect to the NeoBatch Database when running Jobs.

1. Run the JAMS Client as an administrator.
2. Select the **Execution Methods** shortcut from the JAMS menu.
3. Double-click the **NeoBatchCatalog** execution method to open the Execution Method Definition dialog.

4. Select the **Parameters** tab, then double click on any parameter to open the Parameter Definition dialog.
5. Configure the NeoBatch parameters as needed for the given NeoBatch environment:

Parameter	Description
NeoBatchClassName	Any alphanumeric string as determined by NeoBatch classes - the name of the mainframe class name that NeoBatch Jobs will run as.
NeoBatchDbConnectionString	The connection string to the NeoBatch Database. Update this for the given environment.
NeoBatchSubmitAssembly	The assembly used by NeoBatch. For example, (Alchemy.NeoBatch.Submit, Version=3.2.1.30, Culture=neutral, PublicKeyToken=722c951c23d7c1ea)

The NeoBatch Submit Assembly Version and Key Token can be found by running the following powershell script. Note that the PowerShell script should be edited to reflect the path to Alchemy.NeoBatch.Submit.dll for the given environment.

```
([system.reflection.assembly]::loadfile
("c:\MyDLL.dll")).FullName
```

6. Click the **Save and Close** button to finalize the changes.
7. Repeat Steps 3-7 for the NeoBatchJcl execution method.

Creating NeoBatch Jobs in JAMS

With the Execution Methods properly configured in JAMS, NeoBatch Jobs may be run by:

- Using **NeoBatchJcl** to run JCL code directly in JAMS
- Using **NeoBatchCatalog** to reference and run an existing Job in the NeoBatch catalog

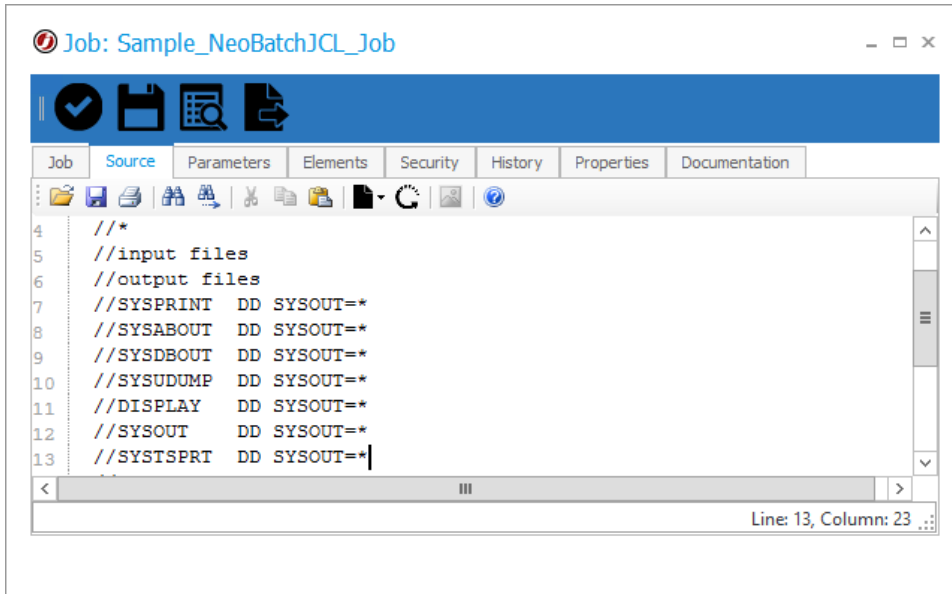
Run JCL directly in JAMS

1. Select the desired folder for the NeoBatch Job, then ensure the Job Definitions tab is selected.
2. Click the **Add** button from the Control Bar to open the Add a New JAMS Job Definition dialog.
3. In the dialog, give the new Job a Name, Description (optional), and Execution Method. In this case, select **NeoBatchJcl** to create a NeoBatch JCL Job.

4. By default, the full Job Definition dialog will open when the Job is initially saved. Click **OK**.
5. The Job Definition dialog will open.
6. Define Elements, Parameters, Security, Properties, and Documentation as desired.

NOTE: An Execute As property should be configured with a user that has access to the NeoBatch Server.

7. Paste or write the desired JCL code in the **Source**.



8. **Save and Close** the Job definition.

NOTE: Hosting JCL in JAMS allows users to use the JAMS Audit Trail to monitor JCL code changes.

Run a NeoBatch Job from the NeoBatch Catalog

1. Select the desired folder for the NeoBatch Job, then ensure the Job Definitions tab is selected.
2. Click the **Add** button from the Control Bar to open the Add a New JAMS Job Definition dialog.
3. In the dialog, give the new Job a Name, Description (optional), and Execution Method. In this case, select **NeoBatchCatalog** to create a NeoBatch Catalog Job.
4. By default, the full Job Definition dialog will open when the Job is initially saved. Click **OK**.
5. The Job Definition dialog will open.

6. Define Elements, Parameters, Security, Properties, and Documentation as desired.

NOTE: An Execute As property should be configured with a user that has access to the NeoBatch Server.

7. In the Job Source, set the path to a desired NeoBatch Job Member in the NeoBatch Catalog. This Job will be run when the JAMS Job Runs.
8. **Save and Close** the Job definition.

Netezza in JAMS

Installing the Netezza Integration Pack

To use the JAMS Add-on for Netezza, users must complete the following installations:

- Install the JAMS Scheduler and JAMS Add-on for Netezza on the JAMS Server.
- Install JAMS AgentX on the Netezza Server, or on a Linux Machine with access to the Netezza Server.

Installing Netezza for JAMS

1. On the JAMS Scheduler Server, run the JAMS Installer.
2. On the Feature Selection step, ensure the **JAMS Integration Pack for Netezza** Integration Pack is checked. Do not un-check any components unless those components should be uninstalled.
3. Finish the installation.
4. Deploy JAMS AgentX from the JAMS Scheduler Server to the Linux Machine, or install JAMS AgentX directly onto the Linux Machine. See [AgentX Documentation](#) for detailed instructions on deploying JAMS AgentX.

NOTE: Un Checking components in the Feature Selection step will uninstall those components.

NOTE: To deploy and use JAMS AgentX, a user must be configured in JAMS with permissions to the relevant Netezza or Linux box where AgentX will be used.

Creating Netezza Jobs in JAMS

JAMS Offers NetezzaLoad, NetezzaBackup, and NetezzaSQL Execution Methods to automate Netezza.

The **NetezzaLoad** Execution Method allows users to automate loads to the Netezza Server.

The **NetezzaBackup** Execution Method allows users to automate backups of Netezza Databases.

The **NetezzaSQL** Execution Method allows users to run a SQL Query on a Netezza system.

Creating NetezzaLoad Jobs in JAMS

1. Create a Job in the desired folder and choose the **NetezzaLoad** Execution Method.
2. Select the desired folder for the NetezzaLoad Job, then ensure the Job Definitions tab is selected.
3. Click the **Add** button from the Control Bar to open the Add a New JAMS Job Definition dialog.
4. In the dialog, give the new Job a Name, Description (optional), and Execution Method. In this case, select **NetezzaLoad** to create a Netezza Load Job.
5. By default, the full Job Definition dialog will open when the Job is initially saved. Click **OK**.
6. The Job Definition dialog will open.
7. Define Elements, Parameters, Security, Properties, and Documentation as desired.

NOTE: An Execute As property should be configured with a user that has access to Netezza.

8. On the Source tab, define the available properties as follows:

Property	Description
CA Certificate File	Specified the pathname of the root CA certificate file on the client system. This argument is used by Netezza clients who use peer authentication to verify the Netezza host system. The default value is NULL, which skips the peer authentication process.

Property	Description
Security Level	<p>Specifies the security level that should be used for the session. The argument has 4 available values:</p> <p>0 - preferredUnsecured - This is the default value. Specify this option when you would prefer an unsecured connection, but you will accept a secured connection if the Netezza system requires one.</p> <p>1 - OnlyUnsecured - Specify this option when you want an unsecured connection to the Netezza system. If the Netezza system requires a secured connection, the connection will be rejected.</p> <p>2 - preferredSecured - Specify this option when you want a secured connection to the Netezza system, but you will accept an unsecured connection if the Netezza system is configured to use only unsecured connections.</p> <p>3 - onlySecured - Specify this option when you want a secured connection to the Netezza system. If the Netezza system accepts only unsecured connections, or if you are attempting to connect to a Netezza system that is running a release prior to 4.5, the connection will be rejected.</p>
Table	Specifies the table name. You can specify a fully qualified name for this value.
Port	Specifies the port to use, allowing users to override the default. Leave 0 to use the default value.
Login Timeout	Allows you to enter a different value, expressed in seconds, for the login timeout. This overrides the default value of 30 seconds.
User	Database username. [NZ_USER] by default.
Password	Database password. [NZ_PASSWORD] by default.
Database	Database name. [NZ_DATABASE] by default.

Property	Description
Host	Netezza hostname or IP address. [NZ_HOST] by default.
Delimiter	Specifies field-delimiter character.
Control File	Specifies the control file.
Data File	Specifies the datafile to load.
Log File	Specifies the log file name. If the file exists, this appends to it.
Bad/Rejected Rows File	Specifies the bad/rejected rows filename (overwrite if the file exists).
Output Directory	Specifies the output directory for the log and bad/rejected rows files.
Log File Size	Session variable (LOAD_LOG_MAX_FILESIZE) that specifies the size (in MB) of the log and bad/rejected rows files. The default is 2000MB (2GB).
File Buffer Size	Specifies the chunk size (MB for fileBufSize or bytes for fileBufByteSize) at which to read the data from the source file. Leave 0 to use default value.
Allow Replay	Session variables (LOAD_REPLAY_REGION and MAX_QUERY_RESTARTS) that specify the number of query restarts for load continuation if a SPU has been reset or failed over. If n is a valid non-zero number, it specifies the number of allowable query restarts. If n is 0, the system defaults to the postgres default setting.

9. **Save and Close** the Job definition.

Creating NetezzaBackup Jobs in JAMS

1. Select the desired folder for the Netezza Backup Job, then ensure the Job Definitions tab is selected.
2. Click the **Add** button from the Control Bar to open the Add a New JAMS Job Definition dialog.

3. In the dialog, give the new Job a Name, Description (optional), and Execution Method. In this case, select **NetezzaBackup** to create a Netezza Backup Job.
4. By default, the full Job Definition dialog will open when the Job is initially saved. Click **OK**.
5. The Job Definition dialog will open.
6. Define Elements, Parameters, Security, Properties, and Documentation as desired.

NOTE: An Execute As property should be configured with a user that has access to Netezza.

7. On the Source tab, define the available properties as follows:

Property	Description
User	Database username. [NZ_USER] by default.
Password	Database password. [NZ_PASSWORD] by default.
Database	Database name. [NZ_DATABASE] by default.
Host	Netezza hostname or IP address. [NZ_HOST] by default.
Directory	Specifies a list of one or more space-separated, full pathnames of the directories where the data is to be stored. This option applies to filesystem only (not veritas or tivoli). Users can specify up to 16 directories.
Directory File	Specifies a file with a list of backup target directories, one per line.
Connector	Names the connector to which you are sending the backup, either filesystem, veritas, or tivoli.
Connector Arguments	Specifies a colon-separated list of passthrough arguments for the connector.
Backup Type	Specifies backup type.
Backup Users	Backs up all users and groups, and any privileges which were defined in the system database. Note that the command backs up all users and groups regardless of whether they are referenced by any permission grants. The system backs up all global adminlevel permissions (those that are not associated with particular databases). The system does not back up permissions associated with specific databases, because they are saved during regular database backup of each database.

Property	Description
Backup Schema Only	Saves only the schema of the specified database, but not the user data in tables or views. The schema includes the definitions of objects such as tables, views, synonyms, sequences, and others, as well as any access privileges defined in the database. This option is an easy way to replicate an empty database schema within a Netezza system.
Number of Streams	Backs up the data using the specified number of streams.
Backup Set ID	Specifies the backup users with to user for incremental backup, rather than the default. Note: the default backup set is the most recent backup set of the database you specify. You can override the default by using this option.
Secret Keyword	Specifies a string value needed to generate a 256-bit symmetric key, which is used to encrypt the host key in the backed up data.

8. **Save and Close** the Job definition.

Creating NetezzaSQL Jobs in JAMS

1. Create a Job in the desired folder and choose the **NetezzaSQL** Execution Method.
2. Select the desired folder for the Netezza Job, then ensure the Job Definitions tab is selected.
3. Click the **Add** button from the Control Bar to open the Add a New JAMS Job Definition dialog.
4. In the dialog, give the new Job a Name, Description (optional), and Execution Method. In this case, select **NetezzaSQL** to create a Netezza SQL Job.
5. By default, the full Job Definition dialog will open when the Job is initially saved. Click **OK**.
6. The Job Definition dialog will open.
7. Define Elements, Parameters, Security, Properties, and Documentation as desired.

NOTE: An Execute As property should be configured with a user that has access to Netezza.

8. On the Source tab, define the available properties as needed:

Property	Description
User	Database username. [NZ_USER] by default.
Password	Database password. [NZ_PASSWORD] by default.
Database	Database name. [NZ_DATABASE] by default.
Host	Netezza hostname or IP address. [NZ_HOST] by default.
Query	Specifies SQL query to run.
Query File	Execute queries from this file.
Unaligned Table Output	Specifies unaligned table output mode. Normally, output is well formatted with white space added so that columns align. If you use this option, the nzsqli command removes extra white space. Use this option when you want to compare results between two systems or when you want to transfer data.
Print Rows Only	Normally the nzsqli command includes column headings and a summary row for all SQL queries. Use 'Print Rows Only' option to eliminate the column headings and summary row. Use this option with the 'Unaligned Table Output' option to produce data in a transportable format.
Expand Table Output	Users may use this option to display the query results vertically instead of in the default table/2D grid.
HTML Table Output	Format the nzsqli output to use HTML tags.
Field Separator	Set the field separator.
Output File	File to send the output to.

9. **Save and Close** the Job definition.

Oracle

JAMS provides integrations with Oracle.

Automating Oracle EBS Jobs

JAMS natively supports automation using many leading business applications, including the Oracle E-Business Suite. Integration Packs for this and other preferred applications are selected during the JAMS installation process. Once the Oracle E-Business Suite integration is enabled, users can create, manage, deploy and monitor every kind of Oracle EBS batch process job, regardless of where it resides.

Installing the Oracle E-Business Suite Integration Pack

To work with Oracle EBS in JAMS, users must install the Oracle E-Business Suite Integration Pack on the JAMS Scheduler Server and deploy an Agent to the Oracle Server. AgentD and AgentX can be used with this integration.

- On the JAMS Scheduler Server, run the JAMS Installer.
- On the Feature Selection step, ensure the **JAMS Integration Pack for Oracle E-Business Suite** component is checked. Do not clear the check boxes for any components unless those components should be uninstalled.
- Finish the installation.

NOTE: Clearing the check boxes for components in the Feature Selection step will uninstall those components.

Configuring Oracle EBS Connection Strings

The Oracle EBS Connection String allows users to connect to Oracle from within JAMS. This connection string must be properly configured to run Oracle Jobs in JAMS. The Oracle EBS connection should be configured on each Oracle EBS Execution Method in JAMS.

Users may override the connection string configured on an Execution Method by configuring a Connection String Parameter at the Folder level or Job level.

The connection string in the OEBSCConnectionString Parameter must have the User Id and Password options defined. To define the values, enter placeholders for the User Id and Password options. For example, enter “User Id = {0}; Password = {1};”. The placeholders are replaced with values from the selected Oracle Credential field on the Source tab in the Job Definition.

NOTE: You must set up a Credential within JAMS before you create the Oracle Jobs. See [Working with User Credential Definitions](#) for more information.

NOTE: If the User on an Oracle EBS Job has OS Authentication permissions to Oracle, then JAMS will pass the credentials from the User into the defined connection string during execution, and an Oracle user is not needed in the Job's source. If the User on the Job does not have OS Authentication, an Oracle user must be defined in the dropdown list that has permissions to the oracle Database.

Configure the Oracle EBS Connection on the Execution Method

1. Run the JAMS Client as an administrator.
2. Select the **Execution Methods** shortcut from the menu.
3. Double-click an Oracle EBS Execution Method to open the Execution Method Properties dialog.
4. On the Parameters tab, double-click the **OEBSCConnectionString** Parameter.
5. Set the **Default Value** to the following, with "Host=Orchid64" and "SERVICE_NAME=MVP" adjusted to values that match the given environment.

```
User Id = {0}; Password = {1}; Data Source = (DESCRIPTION =
  (ADDRESS = (PROTOCOL = TCP)(HOST = Orchid64)(PORT = 1521))
  (CONNECT_DATA = (SERVICE_NAME = MVP)));
```

6. Click **Save and Close** the dialog.
7. Click **Save and Close** the Execution Method dialog.

Configure the Oracle EBS Connection on the Folder level

A Connection String should ONLY be defined at the Folder level if users want to override the Execution Method level Connection Strings. The Folder level Connection String will apply to all Oracle EBS Jobs within that folder.

1. Right-click the desired Folder and select **Properties**.
2. Select the **Parameters** tab, and check for a Parameter named OEBSCConnectionString. If this Parameter does not yet exist on the folder, click the **Add Item** button and select **Add Parameter**.
3. In the Add a Parameter dialog, name the Parameter **OEBSCConnectionString**.
4. Ensure the Data Type is set to Text.
5. Set the **Default Value** to the following, with "Host=Orchid64" and "SERVICE_NAME=MVP" adjusted to values that match the given environment.

```
User Id = {0}; Password = {1}; Data Source = (DESCRIPTION =
  (ADDRESS = (PROTOCOL = TCP)(HOST = Orchid64)
  (PORT = 1521))(CONNECT_DATA = (SERVICE_NAME = MVP)));
```

Parameter Name	OEBSConnectionString
Data Type	Text
Default Value	User Id = {0}; Password = {1}; Data Source = (DESCRIPTION = (AD...
Edit After Adding	<input checked="" type="checkbox"/>

Default Value
(Optional) The Default Value of the parameter

Ok Cancel

6. Save and Close the Folder Properties.

Configure the Oracle EBS Connection on the Job level

A Connection String Parameter should ONLY be defined at the Job level if users want to override Execution Method or Folder level Connection Strings.

1. Right-click a Job and select **Properties**.
2. Select the **Parameters** tab, and check for a Parameter named OEBSConnectionString. If this Parameter does not yet exist on the Job, click the **Add Item** button and select **Add Parameter**.
3. In the Add a Parameter wizard, name the Parameter **OEBSConnectionString**.
4. Ensure the Data Type is set to Text.
5. On the Parameter Value step, set the **Default Value** to the following, with "Host=Orchid64" and "SERVICE_NAME=MVP" adjusted to values that match the given environment.

```
User Id = {0}; Password = {1}; Data Source = (DESCRIPTION =
  (ADDRESS = (PROTOCOL = TCP)
    (HOST = Orchid64) (PORT = 1521)) (CONNECT_DATA = (SERVICE_NAME =
      MVP))) ;
```

Add a Parameter	
Parameter Name	OEBSConnectionString
Data Type	Text
Default Value	User Id = {0}; Password = {1}; Data Source = (DESCRIPTION = (AD...
Edit After Adding	<input checked="" type="checkbox"/>

Default Value
(Optional) The Default Value of the parameter

Ok Cancel

6. Click **Save and Close** on the Job Properties.

Creating Oracle EBS Jobs in JAMS

JAMS offers OracleEBSConcurrent, OracleEBSJobSsh, and OracleEBSRequestSet Execution Methods to run Oracle EBS Jobs.

The **OracleEBSConcurrent** Execution Method gives users a series of automatically populated drop-down fields to run Concurrent Jobs, including the ability to view and set each parameter on the Concurrent Job from within JAMS.

The **OracleEBSJobSsh** Execution Method gives users a fill in the blank style form to run Concurrent Jobs from within JAMS. The advantage of the fill in the blank form over the OracleEBSConcurrent Execution Method is that field values in the Job Source may be defined as JAMS Parameters. By using Parameters in the Job Source, users can create a self-service Oracle Job from within JAMS.

The **OracleEBSRequestSet** Execution Method gives users a series of automatically populated drop-down fields to run Request Sets, including the ability to see each Concurrent Job (and modify the Concurrent Job's Parameters) within the Request Set.

Creating OracleEBSConcurrent Jobs in JAMS

1. Select the desired folder for the OEBSSConcurrent Job, and ensure the Job Definitions tab is selected.
2. Click the **Add** button from the Control Bar to open the Add a New JAMS Job Definition dialog.
3. In the dialog, give the new Job a **Name**, **Description** (optional), and **Execution Method**.
In this case, select **OracleEBSConcurrent** to create an Oracle EBS Concurrent Job.
4. Click **OK**. The Job Definition dialog is displayed.
5. Define Elements, Parameters, Security, Properties, and Documentation as desired.
6. On the Source tab, set the **Oracle Credential**. This user should have access to the Oracle Database.
7. Select the **Application User** from the automatically populated drop-down list.
8. Select the **Responsibility** group from the automatically populated drop-down list.
9. Select the relevant **Application** from the automatically populated drop-down list.
10. Select the **Concurrent Program** to be run from the automatically populated drop-down list. The Parameters section of the Source will automatically populate with the Concurrent Program's available Parameters.
11. Set Source **Parameters** as desired, ensuring that required Parameters have values.
12. Click **Save and Close** to finish editing the Job Definition.

Creating OracleEBSJobSsh Jobs in JAMS

OracleEBSJobSsh Jobs in JAMS provide a fill-in-the-blank style form for users to configure their Concurrent Oracle Jobs in JAMS. The benefit of the fill-in-the-blank form over the automatically populated drop-down list of the OracleEBSConcurrent Execution Method is the parametrization of field values. By using Parameters in the Job Source, users can create a self-service Oracle Job from within JAMS.

1. Select the desired folder for the OracleEBSJobSsh Job, and ensure the Job Definitions tab is selected.
2. Click the **Add** button from the Control Bar to open the Add a New JAMS Job Definition dialog.
3. In the dialog, give the new Job a **Name**, **Description** (optional), and **Execution Method**.
In this case, select **OracleEBSJobSsh** to create an Oracle EBS Ssh Job.
4. Click **OK**. The Job Definition dialog is displayed.
5. Define Elements, Parameters, Security, Properties, and Documentation as desired.

6. In the Source, specify the information on the Job to run in Oracle EBS. See the available fields and their Oracle counterparts below.

JAMS	Oracle
Job Name	Program Shortname
Application Short Name	Program Application Shortname
Responsibility Short name	Responsibility App Shortname
Responsibility Name	Responsibility Name
Username	Username
Parameters	Parameters

Note that Parameters in the Parameters field are **space delimited**.

7. Click **Save and Close** on the Job Definition.

Creating OracleEBSRequestSet Jobs in JAMS

1. Select the desired folder for the OracleEBSRequestSet Job, and ensure the Job Definitions tab is selected.
2. Click the **Add** button from the Control Bar to open the Add a New JAMS Job Definition dialog.
3. In the dialog, give the new Job a Name, Description (optional), and Execution Method. In this case, select **OracleEBSRequestSet** to create an Oracle EBS Request Set Job.
4. Click **OK**. The Job Definition dialog is displayed.
5. Define Elements, Parameters, Security, Properties, and Documentation as desired.
6. On the Source tab, set the **Oracle Credential**. This user should have access to the Oracle Database.
7. Select the **Application User** from the automatically populated drop-down list.
8. Select the **Responsibility** group from the automatically populated drop-down list.
9. Select the relevant **Application** from the automatically populated drop-down list.
10. Select the **Request Set** to be run from the automatically populated drop-down list. The Request Set Jobs section will populate with a list of the Concurrent Programs in the Request Set.
11. Select a Concurrent Program and then click the **Parameters** button to open the parameters of a Concurrent Program within the Request Set.
12. In the Parameters dialog, set Parameter values as desired, and then click **OK**.
13. Click **Save and Close** to finish editing the Job.

Configuring the OracleStoredProc Execution Method

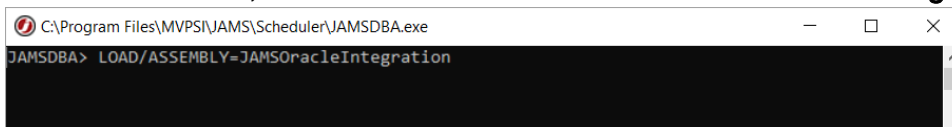
Installing the OracleStoredProc Execution Method

To install the OracleStoredProc Execution method, run the command below in JAMSDBA after JAMS is installed. This command also adds the Oracle Sequence Tasks and the Oracle Connection Type in the Connection Store.

It is recommended that you [create a maintenance window](#) if you are doing this during normal business hours due to restarting the JAMS Scheduler and JAMS Server services.

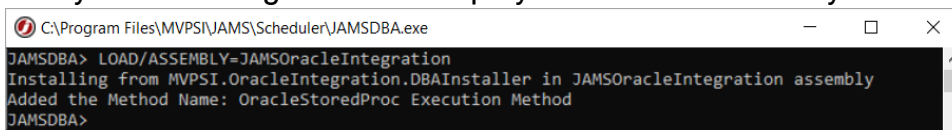
NOTE: If you are upgrading from JAMS V7.2 to V7.3 and you were using the Oracle Integration that includes the OracleStoredProc Execution Method, you may need to run the command below to use the Oracle Integration in JAMS V7.3.

1. Go to <JAMS Install Folder>\MVPSI\JAMS\Scheduler.
2. Right-click **JAMSDBA.exe** and select **Run as administrator**.
3. In the JAMSDBA, run the **LOAD/ASSEMBLY=JAMSOacleIntegration** command.



```
C:\Program Files\MVPSI\JAMS\Scheduler\JAMSDBA.exe
JAMSDBA> LOAD/ASSEMBLY=JAMSOacleIntegration
```

4. Verify the following screen is displayed after the assembly loads.



```
C:\Program Files\MVPSI\JAMS\Scheduler\JAMSDBA.exe
JAMSDBA> LOAD/ASSEMBLY=JAMSOacleIntegration
Installing from MVPSI.OracleIntegration.DBAInstaller in JAMSOacleIntegration assembly
Added the Method Name: OracleStoredProc Execution Method
JAMSDBA>
```

5. On the JAMS Scheduler, open the Windows Services application.
6. Click the JAMS Scheduler service and click **Restart**.
7. Click the JAMS Server service and click **Restart**.

Configuring the OracleStoredProc Execution Method

In order to use the Oracle Stored Procedure Execution Method you will first need to have a user name, password and a datasource. As long as the datasource is in a format that is compatible with an Oracle Connection String, it will open a connection to the Oracle DB server.

The Oracle Stored Procedure relies on a default value being present in the parameters of the Execution Method, which can be overridden at the Folder or Job level.

The default format for the Execution Method is:

- User Id={0}
- Password={1}
- DataSource={2}

NOTE: By default, the User Id, Password, and DataSource set on the given OracleStoredProc Job will be passed into the "{}" braces on the Execution Method. Users do not need to edit the OracleConnectionString parameter by default.

If using a different connection string, you will need to override the original connection string by creating a new Parameter at the Folder or Job level called OracleConnectionString.

Once the ConnectionString format has been set, include a datasource and User Name in the Execution Method's Source tab. Establish the connection. Once the User has authorized access to the Oracle DB Server, click on the dropdown list to view a populated listing of stored procedures.

NOTE: Users without a Data Source Name (DSN) for their Oracle Connection will need to specify the entire traditional Oracle Connection String in the Data Source field of the OracleStoredProc Job's Source tab.

When a stored procedure is selected, the executable will display the parameter values associated with the stored procedure and prepend a "@" symbol in order to distinguish the procedure parameters from the executable parameters.

Once all the properties have been populated, save and submit the Job.

NOTE: Submit Options located in the Job properties can now be used to manually enter your parameter values.

PeopleSoft Integration Pack

The JAMS Integration Pack for PeopleSoft allows users to initiate PeopleSoft processes and take advantage of PeopleSoft's monitoring and report distribution capabilities using the sophisticated and reliable JAMS scheduling software.

The JAMS Integration Pack acts as an intermediary between JAMS and PeopleSoft. When the adapter runs it initiates communication with the PeopleSoft Process Scheduler and instructs PeopleSoft to execute the PeopleSoft jobs as configured in JAMS. The process can then be monitored, canceled, and restarted in either PeopleSoft or JAMS.

Supported Process Types

The JAMS Integration Pack for PeopleSoft can initiate the following process types delivered by PeopleSoft:

- Application Engine
- COBOL SQL
- CRW Online
- Crystal
- Crystal Check
- nVision
- nVision-Report
- nVision-ReportBook
- PS Job
- SQR Process
- SQR Report
- XML Publisher

In addition, the JAMS Integration Pack for PeopleSoft can initiate custom process types that you have created. For the best results, these custom processes must be “API Aware”.

For more information, please refer to the PeopleSoft Process Scheduler PeopleBook under [“Using Process Request APIs”](#).

Supported PeopleTools Versions

The JAMS Integration Pack for PeopleSoft works with PeopleTools versions 8.5x.

To request certification for additional PeopleTools releases, please contact your JAMS customer service representative.

Supported Operating Systems

The JAMS Integration Pack for PeopleSoft works with any Unix/Linux or Windows operating system that is supported by Oracle.

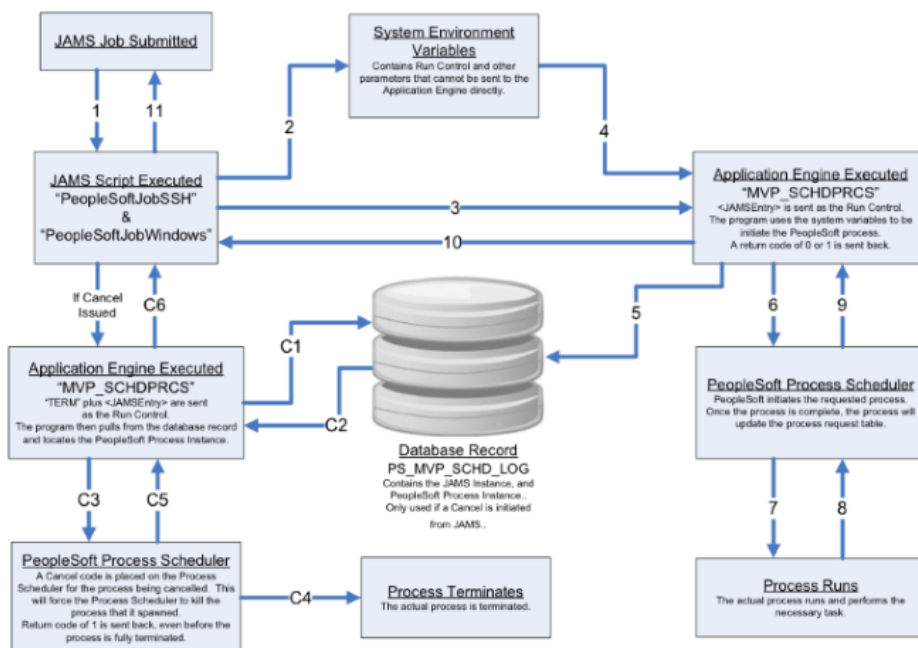
NOTE: The Windows operating system must run Microsoft PowerShell.

JAMS Integration Overview and Process Flow

The JAMS Integration Pack for PeopleSoft consists of two parts:

1. The JAMS components capture information about any PeopleSoft processes that you want to schedule.
2. An Application Engine process called MVP_SCHDPRCS that schedules the jobs through the PeopleSoft Process Scheduler API, monitors, and cancels the job when necessary.

The following diagram illustrates the detailed process flow of the JAMS Integration for PeopleSoft.

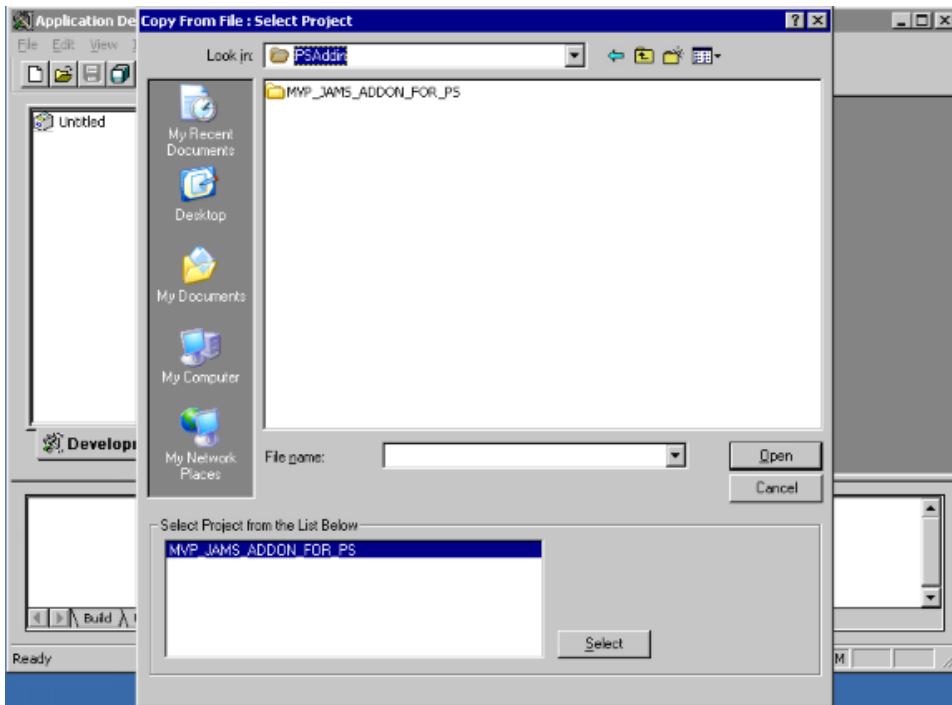


Installing the JAMS Integration Pack for PeopleSoft

The JAMS Integration for PeopleSoft contains an Application Engine program and supporting app designer objects to schedule the jobs. These objects are included in a project that must be installed and operating in the PeopleSoft environment.

1. Launch the **Application Designer** and connect to your PeopleSoft instance in 2-tier Mode.

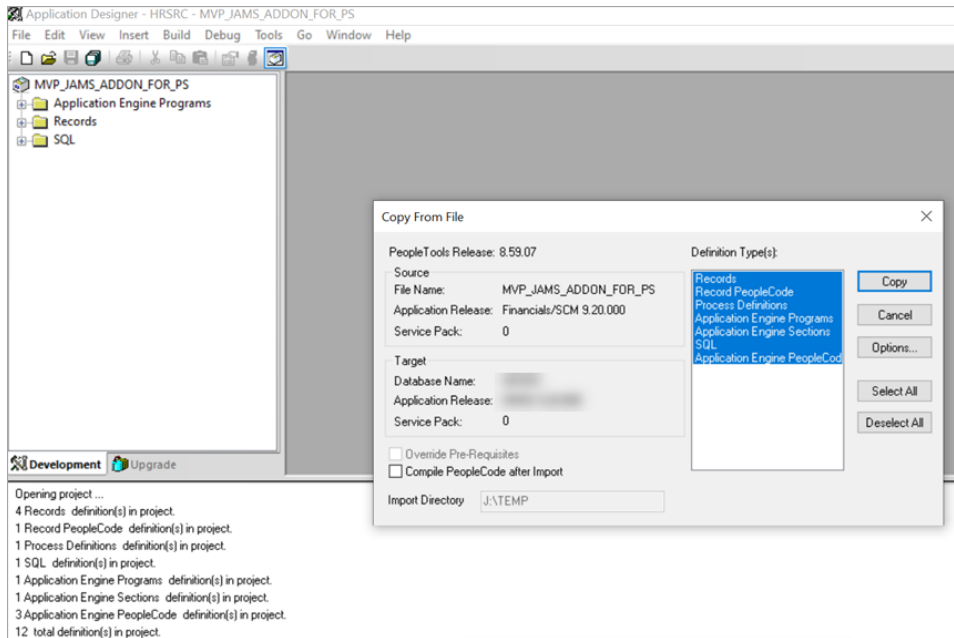
2. Navigate to Tools > Copy Project > From File.



3. From your JAMS Scheduler machine, copy the MVP_JAMS_ADDON_FOR_PS.ini and MVP_JAMS_ADDON_FOR_PS.xml files from the >Program Files\MVPSI\JAMS\Scheduler directory. Paste them into a folder named, MVP_JAMS_ADDON_FOR_PS that is manually created on the PeopleSoft server.

Browse to the location of that newly created folder. The Project name "MVP_JAMS_ADDON_FOR_PS" appears in the lower box. Click the **Select** button.

4. The Copy from File dialog box appears. Select the **Copy** button.

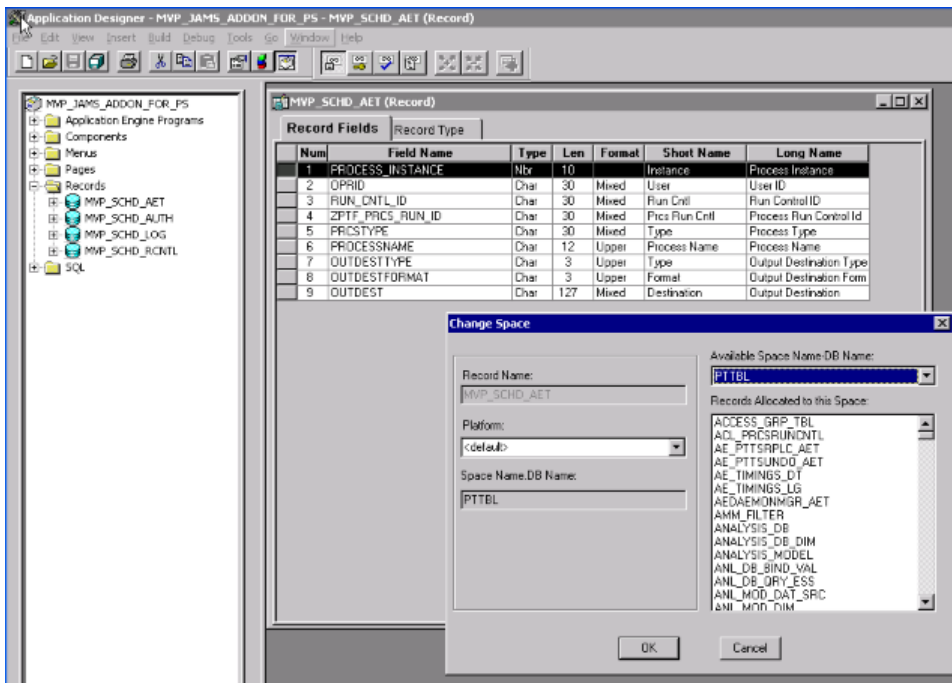


5. Set the default tablespace. Only one record needs to have the default tablespace.

The JAMS Integration for PeopleSoft requires one table to maintain its scheduling process. Your PeopleSoft System Administrator will likely want to specify which tablespace to use, depending on your organization's database platform.

Set the tablespace using the steps below:

- Expand the "Records" node in the project tree.
- Double-click MVP_SCHD_LOG to open the record definition.
- Navigate to Tools > Data Administration > Set Tablespace. The Change Space window appears.
- Select the appropriate Platform and Tablespace Name depending on your organizations database platform and click **OK**.
- Save the record.



6. Build the Project to create the PS_MVP_SCHED_RCNTL table. Only one record needs to be created.
 - Navigate to Build > Project.
 - Select the **Create Tables** and **Create Indexes** checkboxes.
 - Choose the **Execute SQL Now** button.
 - Select the **Build** button.
 - To complete the installation process, review the log file for errors.

Configuring the Execution Method

Before JAMS can execute a Job, additional information about your PeopleSoft environment will need to be provided. To get started, follow the steps below.

1. Open the JAMS Client as an administrator.
2. Select the **Execution Methods** shortcut from the menu.
3. Locate the listed PeopleSoft Job Execution Methods.
 - If your organization's PeopleSoft Process Scheduler runs on Unix or Linux, open the **PeopleSoftJobSsh**.
 - If your organization's PeopleSoft Process Scheduler runs on Windows, open **PeopleSoftJobWindows**.
4. When the definition dialog opens, navigate to the Parameters tab.

5. Locate the parameter **PS_ORACLE_HOME**. Open the parameter definition dialog by double clicking the parameter or highlighting it and clicking the **Properties** button.
6. Enter the correct information in the Default Value field.

Job Parameter: PS_ORACLE_HOME

4 of 26

Allow Entry	<input checked="" type="checkbox"/>
Help Text	
Hide	<input checked="" type="checkbox"/>
Must Fill	<input type="checkbox"/>
Prompt	Oracle home:
Required	<input type="checkbox"/>
Sequence	10004
Uppercase	<input type="checkbox"/>
Validation Data	
Validation Type	None

Value

Default Format	
Default Value	/u01/app/oracle/product/11.1.0/db_1
Encrypt	<input type="checkbox"/>
Variable	<input type="checkbox"/>

Default Value
The default value of the Parameter. Default values may be overridden.

7. Click the right arrow on the window header to move to the next parameter. Define parameters as necessary to connect to the PeopleSoft environment.
8. **Save and Close** the Parameters dialog.
9. **Save and Close** the Execution Method dialog.

NOTE: All but the "Additional_Param_#" parameters must be reviewed. Below is a reference table that describes each parameter.

Parameter	Description	Example
DBInstance	Database Instance/Name	EP91DMO
DBType	Database Type	ORACLEMSSQLDB2SYBASE
JAMS_PS_DisplayLogFiles	Set this value if the posted files (Log, Txt, Stdout, Out) should be listed. Ensure JAMS_PS_WaitForDist is set to True.	JAMS_PS_DisplayLogFiles=True

Parameter	Description	Example
JAMS_PS_PSMessagesLog	Set this value if the MessageLog record should be displayed in the log. The default is True.	JAMS_PS_PSMessagesLog=True
JAMS_PS_PSReports	Set this value to specify where the PeopleSoft Reports are located on the process scheduler. Ensure JAMS_PS_WaitForDist is set to True.	JAMS_PS_PSReports=c:\psreports\
JAMS_PS_SecondsToSleep	Set this value to specify how long the Sleep Function will run. The default is 5 seconds.	JAMS_PS_SecondsToSleep=5
JAMS_PS_SleepMethod	Set this value to specify the sleep method. The default is Java. Select Sleep to use the Unix command for sleep. Select Java for Windows. Only Linux can switch to Sleep.	JAMS_PS_SleepMethod=Java
JAMS_PS_WaitForDist	Set this value to False if you do not want to wait until the Report Distribution is finished. The default is True.	JAMS_PS_WaitForDist=True
PS_ORACLE_HOME	Oracle Home	/u01/app/oracle/product/11.1.0/db_1

Parameter	Description	Example
PS_ORACLE_SID	Oracle Database Name	EP91DMO
PS_TUXDIR	Tuxedo Directory	/usr/local/psoft/Oracle/Middleware/tuxedo10gR3
PS_HOME	PeopleSoft Home. You can use \$ORACLE_SID as a variable.	/usr/local/psoft\$ORACLE_SID
PS_CFG_HOME	Configuration Home	/usr/local/psoft/cfg850
PS_SQRDIR	SQR Directory. You can use \$ORACLE_SID as a variable.	/usr/local/psoft\$ORACLE_SID/bin/sqr/ORA/bin
PS_SERVER_CFG	Server Configuration Home. You can use the \$PS_CFG_HOME and \$ORACLE_SID variables.	\$PS_CFG_HOME/appserv/prcs\$ORACLE_SID/psprcs.cfg
PS_JAMS_TMP	JAMS Temp Folder	/home/psoft/PS_CACHE/files/JAMS/
PS_RUNLOCATION	Run Location	PSUNIX

Setting Up a JAMS Job for PeopleSoft

Before setting up any Job in JAMS, the user running the process should log into PeopleSoft, navigate to the run control page and enter the information required on the PeopleSoft page.

1. Select the desired folder for the PeopleSoft Job, then ensure the Job Definitions tab is selected.
2. Click the **Add** button from the Control Bar to open the Add a New JAMS Job Definition dialog.

3. In the dialog, give the new Job a Name, Description (optional), and Execution Method.

NOTE: Select **PeopleSoftJobWindows** or **PeopleSoftJobSsh** to create a PeopleSoft Job. If the organization's PeopleSoft Process Scheduler runs on UNIX or Linux, choose the **PeopleSoftJobSsh** Execution Method. If the Scheduler runs on Windows, select the **PeopleSoftJobWindows** Execution Method.

4. By default, the full Job Definition dialog will open when the Job is initially saved. Click **OK**.
5. The Job Definition dialog will open.
6. Define Elements, Parameters, Security, Properties, and Documentation as desired.

NOTE: An Execute As property should be configured with a user that has access to the PeopleSoft Server.

7. On the Source tab, populate the Run Control ID with the one that is already set up in PeopleSoft.
8. Enter the name of the PeopleSoft User who owns the Run Control.
9. Choose the Process Type from the dropdown and enter the Process Name.
10. Select the **Output Type** and **Output Destination Format** using the individual dropdowns. The Output Destination is required only if the selected Output Type is **File**.
11. **Save and Close** the Job definition.

Working With Crystal Report Jobs

Crystal Reports jobs in PeopleSoft require that Query Bind Variables be passed to the job on the command line. If scheduling a Crystal Report, use the following steps:

1. If the PeopleSoft process requires additional parameters, you must add these using the Parameters tab. Most Crystal Reports in PeopleSoft require additional parameters. Use the parameter name **Additional_Parm_x** for the name. For parameters 1 through 9, there is no leading zero. See the examples below to identify which parameters to use.
2. Running the Crystal Report inside of PeopleSoft is a great way to identify which parameters need to be passed. Once it has been run, open the process in the PeopleSoft Process Monitor (PeopleTools > Process Scheduler > Process Monitor).

3. Locate the process (Crystal in most cases) and select the **Details** link.

Process List

Server List

View Process Request For

User ID: VP1

Type:

Last

99

Days

Refresh

Server:

Name:

Instance:

to

Run Status:

Distribution Status:

☒ Save On Refresh

Process List

Customize

Find

View All

First

1-22 of 22

Last

Select	Instance	Seq.	Process Type	Process Name	User	Run Date/Time	Run Status	Distribution Status	Details
<input type="checkbox"/>	11472		Crystal	APY1020-	VP1	01/07/2012 3:32:12PM PST	Queued	N/A	Details

4. Click the **Parameters** link.

Process	
Instance:	11472
Name:	APY1020-
Run Status:	Queued
Type:	Crystal
Description:	Posted Voucher Listing
Distribution Status:	N/A

Run	Update Process
Run Control ID: posted_voucher_listing Location: Server Server: PSNT Recurrence:	<input type="radio"/> Hold Request <input type="radio"/> Queue Request <input type="radio"/> Cancel Request <input type="radio"/> Delete Request <input type="radio"/> Restart Request

Date/Time	Actions
Request Created On: 01/07/2012 3:32:15PM PST	Parameters Transfer
Run Anytime After: 01/07/2012 3:32:12PM PST	Message Log
Began Process At:	Batch Timings
Ended Process At:	View Log/Trace

5. Locate the Command Line and review all data, especially at the end of the command line. Most of the time the required parameters come after the "ORIENTL" parameter. In the following example, the five required parameters are: "US001" "2007-11-07" "2011-11-07" "VP1" "posted_voucher_listing". When adding these parameters into JAMS make sure to exclude the double quotes.

Process Request Parameters

Process	
Instance: 11472	Type: Crystal
Name: APY1020-	Description: Posted Voucher Listing

Parameters	
Command Line:	
%TOOLBIN%\PSCRRUN.EXE -CTORACLE -CDEP91DMO -COVP1 -CP%OPRPSWD% -I11472 -RP"APY1020-" -OT6 -OP"%LOG/OUTPUT DIRECTORY%" -LGENG -OF2 -ORIENTL "US001" "2007-11-07" "2011-11-07" "VP1" "posted_voucher_listing"	
Working Dir:	
%DBBIN%	

Authorized To View Reports from the Web	
ID Type	Distribution ID
User	VP1

[Return](#)

6. In the following example, the Crystal APY1020 requires additional parameters.

Sample_PeopleSoft_Job

Summary Source Schedule Properties Parameters Diagram History References Dc

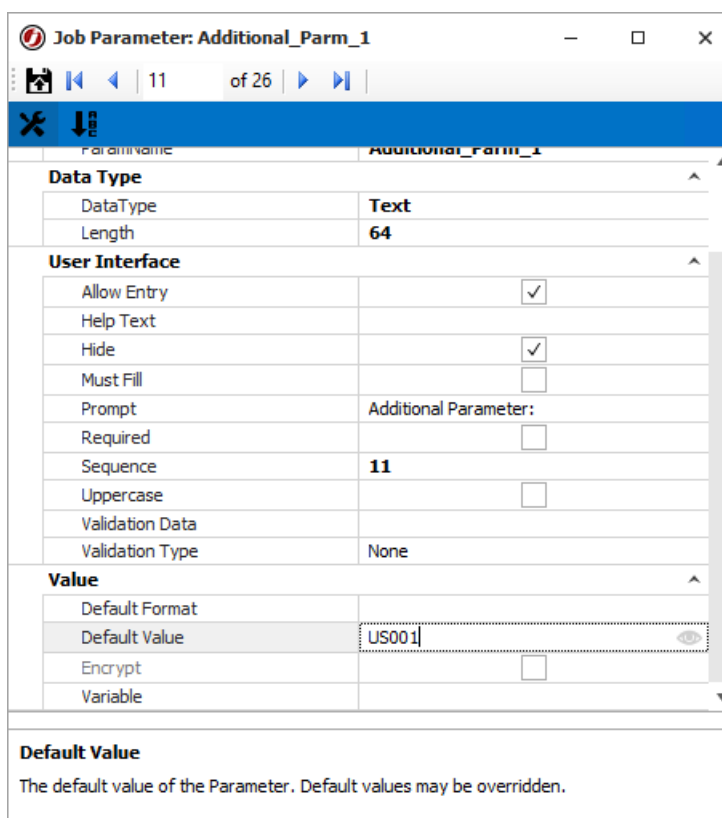
Parameters

Parameter Name	Type	Prompt	Default Value
Additional_Parm_1	Text	Additional Parameter:	
Additional_Parm_2	Text	Additional Parameter:	
Additional_Parm_3	Text	Additional Parameter:	
Additional_Parm_4	Text	Additional Parameter:	

Queries

Enabled	Description	State	Name
<input checked="" type="checkbox"/>	n[c	=	n[c

7. Here's an example of Parameter 1. Notice the Default Value is US001 without the double quotes.



Job Parameter: Additional_Parm_1	
ParamName	Additional_Parm_1
Data Type	
Data Type	Text
Length	64
User Interface	
Allow Entry	<input checked="" type="checkbox"/>
Help Text	
Hide	<input checked="" type="checkbox"/>
Must Fill	<input type="checkbox"/>
Prompt	Additional Parameter:
Required	<input type="checkbox"/>
Sequence	11
Uppercase	<input type="checkbox"/>
Validation Data	
Validation Type	None
Value	
Default Format	
Default Value	US001
Encrypt	<input type="checkbox"/>
Variable	<input type="checkbox"/>
Default Value The default value of the Parameter. Default values may be overridden.	

8. Create the remaining parameters (2007-11-07" "2011-11-07" "VP1" "posted_voucher_listing").
9. **Save and Close** the Parameters dialog.
10. **Save and Close** the Job Definition.
11. When the JAMS Job is submitted, the five parameters will appear (as long as the parameters are not hidden).

NOTE: The additional parameters do not necessarily have to be populated when creating the JAMS Job, but the parameter name itself must be set up at that time.

SAP

JAMS provides integrations with SAP. The articles in this section provide more information on using the SAP Execution Methods, the SAP Business Objects integration, as well as the SAP Job view within JAMS.

Run SAP Business Objects Data Services in JAMS

The SAPDataService execution method allows users to automate any job that already exists in an SAP Business Objects Server environment.

Installing the SAP Data Services Integration Pack

1. Run the JAMS Installer.
2. On the Feature Selection step, ensure the **JAMS Integration Pack for SAP Data Services** component is selected.
3. Complete the installation.

NOTE: Clearing the check boxes for components listed in the installer will uninstall them.

Configuring the SAP Data Service Execution Method

1. Run the JAMS Client as an administrator.
2. Select the **Execution Methods** shortcut from the Shortcuts menu.
3. Double-click the **SAPDataService** execution method to open the Execution Method Definition dialog.
4. Select the **Parameters** tab.
5. Double-click any parameter to open the Parameter Definition dialog.

- Set the SAPAuthentication and SAPEndpoint values to reflect the environment.

Job Parameter: SAPEndpoint

2 of 3

Misc

Description: Job Parameter: SAPEndpoint

Name

ParamName: SAPEndpoint

Data Type

DataType: Text

Length: 0

User Interface

Allow Entry: ☒

Help Text: ☐

Hide: ☐

Must Fill: ☐

Prompt: Address of Data Service Endpoint:

Required: ☒

Sequence: 10002

Uppercase: ☐

Validation Data: ☐

Validation Type: None

Value

Default Format: http://www.sampledomain.com/service...

Default Value:

Default Format

The format string that defines how the Parameter value will be formatted. Default formats vary depending on the Parameter's Data Type.

- Click **Save and Close** on the Parameters dialog.
- Click **Save and Close** on the Execution Method dialog.

NOTE: The user running SAP Data Service Jobs must have full access to the SAP BODS server.

Creating an SAP Data Services Job in JAMS

- On the Home screen, click **Create a Job**.
- Select **JAMS Integration Jobs**.
- Select **Create an SAP Business Objects Data Services job**. If you do not see the Jobs, close JAMS, install the Integration Pack again, and reopen JAMS.
- On the Add a New JAMS Job Definition screen, do the following:
 - In the Folder field, select a folder where the Job will be saved.
 - In the Name field, enter a name for the Job.
 - Optional - In the Description field, enter a description for the Job.
 - Optional - In the Scheduled Date field, select a date for the Job to start.
 - Optional - In the Calendar field, select the calendar to use.

6. Optional - In the Scheduled Time field, select the time for the Job to start.
7. Click **Ok**.
5. Define Schedule Items, Parameters, Security, Properties, and Documentation as desired.

NOTE: On the Properties tab, configure an **Execute As** property with a user that has access to the SAP BODS Server.

6. On the **Source** tab, set the SAP BODS Job options.
 1. In the Authentication field, use the drop-down list to select a CMS Authentication method for your environment.
 2. In the Repository field, use the drop-down list to select the relevant directory where the SAP Job is stored.
 3. In the Job Name field, use the drop-down list to select the desired SAP Job from the specified Repository.
 4. Under Execution Options, select the appropriate options for the Job. See the [SAP documentation](#) for more information.
7. When the relevant SAP Job has been configured in the new JAMS Job, click **Save and Close** on the Job definition.

Using SAP Execution Methods to Run JAMS Jobs

To run SAP Jobs, you can use a Sequence Task within a JAMS Sequence Job or one of two Execution Methods. The Sequence Task uses the definition that is maintained within SAP and uses JAMS for the scheduling. For information on the SAP Sequence Tasks, see [Sequence Tasks](#). The Execution Methods let you manage all the steps and their details within JAMS. They are used to support Job execution processes using SAP outside the [SAP Business Objects Data Services](#) Integration Pack.

Two SAP Execution Methods are available:

- SAPJobV2 provides support for executing multi-step background Jobs. It is also the replacement for the SAPJob and SAPStandardJob Execution Methods.
- SAPProcessChain provides process chain execution handling in JAMS.

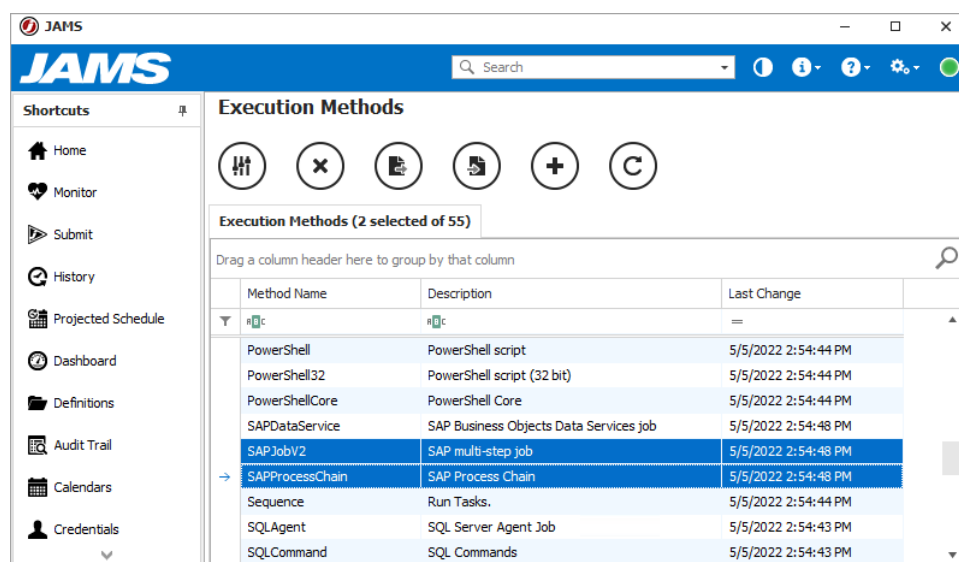
When you create a Job using an SAP Execution Method, it is submitted and executed normally in JAMS. For SAPJobV2, the JAMS log file displays the Job Number that matches in the entry in the SAP Client view. For SAPProcessChain, the Job output includes a process chain number (log ID). The JAMS log file displays the output of the process chain.

NOTE:

To use the SAP Process Chain and SAPJobV2 Execution Methods, you must have the following:

- SAP ERP Central Component (ECC) 6.0, with SAP Enhancement Package 7 (EHP) or newer.
- SAP NetWeaver 7.4 or newer.
- SAP Environment must be a Unicode system.
- Microsoft Visual C++ 2015-2022 Redistributable Pack (x64) installed on the JAMS Scheduler server.

The two screenshots below highlight these Execution Methods along with their respective Parameters. If you have already created the JAMSSAPServer and JAMSSAPClient Parameters, you can continue to use them. However, the recommended approach is to set the SAP Server and Client information within the SAP Connection in the Connection Store.

**NOTE:**

To define an Instance Number on your Execution Method, you have two options:

- Use the Connection in the Connection Store. For the SAP (Custom Application Server) Connection, set the Application Server and Instance Number. Using the Connection Store is the preferred method, but any existing Parameters are supported for backward compatibility.
- Use the JAMSSAPServer Parameter. Set the JAMSSAPServer parameter to "Servername:InstanceNumber" for the given Execution Method. For example, if the server was "192.0.2.0" and the Instance Number was "23", set the JAMSSAPServer parameter to be "192.0.2.0:23".

SAPJobV2

Execution Method | Parameters | Template | Properties | Job Properties | References

Parameter Name	Type	Prompt	Default Value
JAMSSAPServer	Text		192.0.2.0
JAMSSAPClient	Text		001

SAPProcessChain

Execution Method | Parameters | Template | Properties | Job Properties | References

Parameter Name	Type	Prompt	Default Value
JAMSSAPServer	Text		192.0.2.0
JAMSSAPClient	Text		001
JAMSTraceLevel	Text		Off

SAP Connections and Credentials

Before you create a Job, ensure you have created a Credential in JAMS and a Connection in the Connection Store. For the Connection, two SAP connection types are available.

Review the following items regarding creating a Connection or a Credential:

- The SAP user in the SAPCredential property on the SAP Connection must reference a JAMS Credential that has been configured for a SAP user with the permissions outlined in the "SAP Configuration Requirements" section below.
- The Credential in JAMS should have the same username and password as the credential in SAP.
- SAP Instance Numbers are used to define the listener port. The port is defined as "33XX", where "XX" is the Instance Number. If the Instance Number was "23", the SAP port would be "3323".
- To define language on an SAPJobV2 Job or an SAP Process Chain, specify the Logon Language within the Connection that you create in the Connection Store.

See [JAMS Security: Working with User Credential Definitions](#) and [Connections for Integrations](#) for more information.

NOTE: To run SAPJobV2 and SAP Process Jobs on an Agent, ensure the SAP Integration Pack from the JAMS installer is installed on the Agent.

SAP Configuration Requirements

These permissions describe the necessary roles and permissions in SAP to allow users to run a Job using the JAMS SAP Integration.

SAP_BC_BATCH_ADMIN -- This role contains all authorizations for background processing administration, including the creation of background jobs and general administrations functions (SMxx transaction codes, in particular SM36, SM37, SM50, and SM51).

SAP_BC_ENDUSER -- This role contains non-critical basis authorizations for all users, including job creation and job release.

S_BTCH_ADM -- This role allows user the ability to perform any operation on any job, including:

- Selecting jobs from all clients (from the Job Overview, Transaction SM37)
- Deleting any job
- Releasing jobs to start
- Changing jobs, including copying, canceling, checking, repeating, and capturing and debugging
- Displaying jobs, job steps, and job logs
- Triggering events manually (transaction SM64)
- Editing system events
- Working with raised events in the event history (transaction SM62)
- Using restricted job classes A or B
- Scheduling an external program in a job

S_XMI_XBP_A -- This profile needs to be added to the SAP user to allow JAMS to execute an SAP Job or Process Chain.

Five User Type are available in SAP. Based on SAP [recommendations](#), the User should be **“System”**.

Alias	
User Type	Dialog
Security Policy	Dialog
Password	System
Password Status	Communications Data
	Reference (Logon not possible)
	Service

Type	Purpose
Dialog	Individual, interactive system access.
System	Background processing and communication within a system (such as RFC users for ALE, Workflow, TMS, and CUA).
Communication	Dialog-free communication for external RFC calls.
Service	Dialog user available to a larger, anonymous group of users.
Reference	General, non-person related users that allows the assignment of additional identical authorizations, such as for Internet users created with transaction SU01. No logon is possible.

Overview of the properties of users with different user types.

Property \ User Type	Dialog	Communication	System	Service
Dialog Logon (SAP GUI)	X	-	-	X
Multiple Logons	-	X	X	X
RFC Logon	X	X	X	X
Background Job Execution	X	-	X	X
Password Change	X	X	-	-
Logon Ticket can be Generated	X	X	-	-

"X" represents Yes.

SAP Function Modules

Customers running SAP Jobs in secure SAP environments with heavy restrictions on function modules will need to allow the following modules

BAPI_XMI_LOGON	BAPI_XBP_JOB_START_ASAP	BAPI_XBP_JOB_ADD_ABAP_STEP
BAPI_XMI_LOGOFF	BAPI_XBP_JOB_COPY	BAPI_XBP_ADD_JOB_STEP
BAPI_XMI_SET_AUDITLEVEL	BAPI_XBP_JOB_ABORT	BAPI_XBP_EXT_COMM_SEARCH
BAPI_XBP_MODIFY_CRITERIA_TABLE	BAPI_XBP_EVENT_RAISE	RSPC_API_CHAIN_GET_STATUS
BAPI_XBP_GET_INTERCEPTED_JOBS	BAPI_XBP_JOB_JOBLOG_READ	RSPC_API_CHAIN_GET_PROCESSES
BAPI_XBP_JOB_STATUS_GET	RSPC_API_GET_CHAINS	RSPC_API_PROCESS_GET_LOG
BAPI_XBP_VARIANT_CHANGE	BAPI_XBP_JOB_READ	RSPC_API_PROCESS_SKIP
BAPI_XBP_VARINFO	BAPI_XBP_JOB_SELECT	RSPC_API_PROCESS_RESTART
BAPI_XBP_OUTPUT_DEVICE_SEARCH	BAPI_XBP_JOB_DEFINITION_GET	RSPC_API_CHAIN_RESTART
BAPI_XBP_REPORT_SEARCH	BAPI_XBP_JOB_CHILDREN_GET	RSPC_API_CHAIN_INTERRUPT
BAPI_XBP_VARIANT_INFO_GET	BAPI_XBP_JOBLIST_STATUS_GET	RSPC_API_CHAIN_START
BAPI_XBP_JOB_DELETE	BAPI_XBP_JOB_OPEN	RSPC_PROCESS_FINISH
BAPI_XBP_JOB_START_IMMEDIATELY	BAPI_XBP_JOB_CLOSE	RSPC_API_CHAIN_GET_LOG

SAP Multi-Step Job (SAPJobV2)

Creating an SAP Multi-Step Job (SAPJobV2) in JAMS

1. On the Home screen, click **Create a Job**.
2. Select **JAMS Integration Jobs**.
3. Select **Create an SAP multi-step job**. If you do not see the Jobs, close JAMS, install the Integration Pack again, and reopen JAMS.

4. On the Add a New JAMS Job Definition screen, do the following:
 1. In the Folder field, select a folder where the Job will be saved.
 2. In the Name field, enter a name for the Job.
 3. Optional - In the Description field, enter a description for the Job.
 4. Optional - In the Scheduled Date field, select a date for the Job to start.
 5. Optional - In the Calendar field, select the calendar to use.
 6. Optional - In the Scheduled Time field, select the time for the Job to start.
 7. Click **Ok**.
5. On the Job Definition screen, click the **Source** tab.
6. In the SAP Connection field, select the appropriate Connection that was created in the Connection Store.
7. In the Job Name field, verify the correct Job name is displayed.
8. In the Job Class field, set the scheduling priority for the Job to High, Medium, or Low.
9. In the Target Server field, enter the server where the Job will run. If you do not select a server, SAP will select the server and load balance.
10. Review the **Wait for Child Jobs** checkbox. Select the checkbox to force the parent Job to wait for the child Jobs to complete or clear the checkbox to allow the parent Job to complete before the child Jobs have run (finished).
11. Review the **Execute Immediately** checkbox. Select the checkbox to have the SAP Job run immediately, or clear the checkbox to have the Job run ASAP.
12. Click the **Save & Close** icon or add a Job Step.

Adding Job Steps

1. On the Source tab, click **Add** in the Job Steps section.
2. On the Step tab, do the following:
 1. Select **ABAP**, **Command**, or **Program** as the type of Step. The available options on the Step tab change based on the selected type.
 2. In the Name field, enter or select the name of the Step. This field will automatically populate options based on the ABAP and Command types.
 3. In the Variant field, enter the Variant. (This option is available for only the ABAP option.)
 4. In the Parameters field, enter the Parameters that will be passed to the Command or Program type. (This option is available for only the Command and Program options.)
 5. In the Operating System field, enter the operating system where the Command will run. (This option is available for only the Command option.)

6. In the Target Server field, enter the server where the Step will run.
7. (Optional) In the User Name field, enter the SAP user name that will be used to run the Step. This user does not need to exist as a Credential in JAMS, and it lets you specify a user other than the ExecuteAs user on the Job.
3. On the Options tab, review and update the options below as needed. (Additional information can be found in the SAP documentation.) This tab is disabled for the ABAP step.
 1. Select the **Log output** checkbox to set an indicator in SAP to write the standard output for an external command or program to the SAP job log.
 2. Select the **Log errors** checkbox to set an indicator in SAP to write the standard error output for an external command or program to the SAP job log.
 3. Select the **Wait for termination** checkbox to set an indicator in SAP for an external command or program run synchronously or asynchronously.
 4. Select the **Enable Trace** checkbox to set an indicator in SAP if SAP tracing level 3 is activated for tracing SAPXPG, the program that starts an external command or program.
4. On the Output tab, review and update the optional properties below as needed. This tab is disabled for the Command and Program steps.
 1. In the Output Device field, select the output device that is configured in SAP. This field will automatically populate options based on the Output Devices configured within SAP.
 2. In the Format Type field, select the format for the output.
 3. In the Title field, enter a title for the output.
 4. In the Department field, enter the name of the department.
 5. In the Copies field, enter the number of copies that will be created.
 6. In the Start Page field, enter the start page for the output.
 7. In the End Page field, enter the end page for the output.
5. Click **OK**.

SAP Process Chain

Creating an SAP Process Chain Job in JAMS

1. On the Home screen, click **Create a Job**.
2. Select **JAMS Integration Jobs**.
3. Select **Create an SAP Process Chain job**. If you do not see the Jobs, close JAMS, install the Integration Pack again, and reopen JAMS.

4. On the Add a New JAMS Job Definition screen, do the following:
 1. In the Folder field, select a folder where the Job will be saved.
 2. In the Name field, enter a name for the Job.
 3. Optional - In the Description field, enter a description for the Job.
 4. Optional - In the Scheduled Date field, select a date for the Job to start.
 5. Optional - In the Calendar field, select the calendar to use.
 6. Optional - In the Scheduled Time field, select the time for the Job to start.
 7. Click **Ok**.
5. On the Job Definition screen, click the **Source** tab.
6. In the SAP Connection field, select the appropriate connection that was created in the Connection Store.
7. Next to the Process Chain field, click the ... button.
8. On the **Find SAP Process Chain** dialog, enter a name or description to find the correct process chain.
9. Click **Find**.
10. Select the process chain.
11. Click **Ok**.
12. Click **Save and Close**.

Submitting an SAP Job

1. In the JAMS Shortcuts menu, click **Definitions**.
2. Expand the folder containing the SAP Job.
3. Right-click the Job and select **Submit**. The Submit dialog is displayed.
4. Click **Submit Run Request**.

Resuming a Process Chain

If a process fails inside the process chain, you can resume the process chain from the JAMS Monitor view. When the process fails, the icon for the process chain in the Monitor view will change. The Job stays in the "Executing" state, but the execution is halted until you select an option as described below.

SAP Process Chain Execution Method

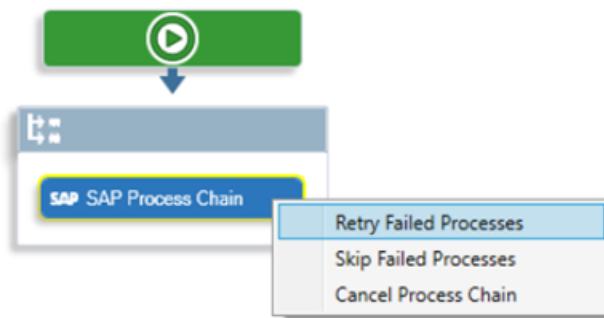
To resume a failed process chain when you are using the SAP Process Chain Execution Method:

1. From the JAMS Shortcut menu, click **Monitor**.
2. Right-click the Job and select **Release**.
3. On the Release Job dialog, select one of the following options:
 - **Release from wait with retry** - Retry the failed process in process chain before continuing.
 - **Release from wait** - Skip the failed process and resume the process chain.

SAP Process Chain Sequence Task

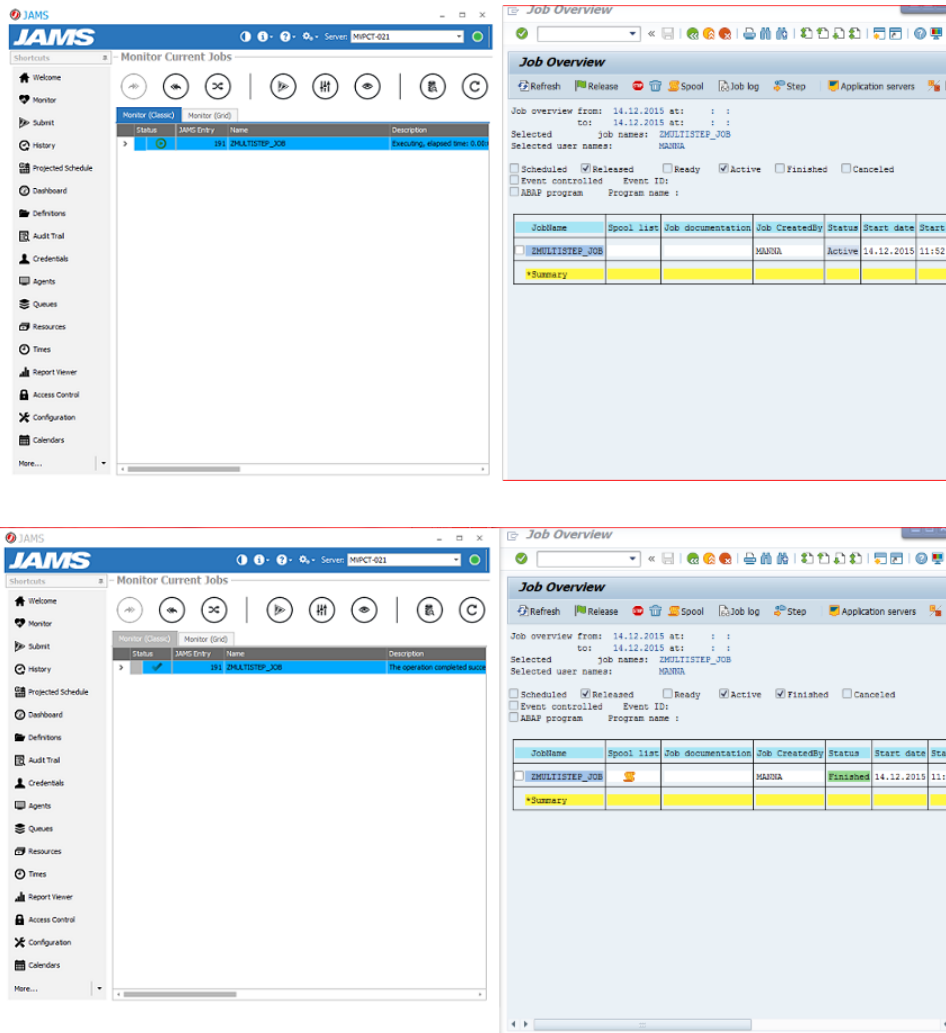
To resume a failed process chain when you are using the SAP Process Chain Sequence Task:

1. From the JAMS Shortcut menu, click **Monitor**.
2. Double-click the entry for the Sequence Job.
3. Click the **Sequence** tab. The Sequence editor is displayed.
4. Right-click the Task in the Design area and select one of the following options:
 - **Retry Failed Processes** - Retry the failed process in process chain before continuing.
 - **Skip Failed Processes** - Skip the failed process and resume the process chain.
 - **Cancel Process Chain** - Stop the process chain. The Task completes successfully.



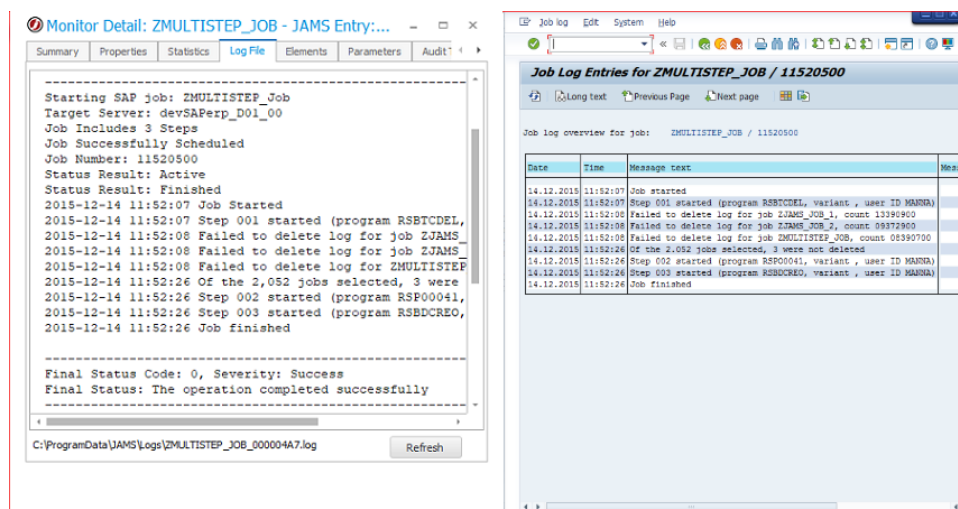
Parallel Execution for JAMS and SAP Jobs

Once JAMS submits the Job, SAP creates and executes a parallel job as shown in the following two screenshots.



Job Execution Logs

Once a Job executes, logs can be viewed from either the JAMS or SAP environments.



Authorized users may, in certain cases, cancel a currently executing Job. In SAP, such a job would go into a cancelled state. Logs can then be checked in both environments.

SAP Jobs View

The SAP Jobs View lets you view all Jobs that are in the SAP schedule for an SAP instance. It is similar to the Monitor view within JAMS, but it only shows jobs in SAP and not the Jobs in JAMS. The SAP Integration must have been installed from the JAMS Installer or JAMSDBA to use the SAP Jobs view.

You also need to have an SAP Connection created in the JAMS Connection Store before you can access this screen. See [Connections in JAMS](#) for more information.

NOTE: This view is used for the SAP Integration Pack. It is not applicable to the SAP Business Objects Data Services Integration.

Accessing the SAP Jobs View

NOTE: Previously, the Jobs View had a dialog to prompt users to enter connection information. That dialog has been replaced with a drop-down box to select an SAP Connection from the Connection Store. To use these new Connections, the JAMS Scheduler must be updated to V7.3 or higher to create SAP Connections for this View in the JAMS V7.3 or higher Client.

1. From the Shortcuts menu, click **SAP Jobs**.
2. On the SAP Server Login screen, select the appropriate connection from the Connection Store.

3. Click **OK**.
4. On the SAP Job Selection screen, do one or more of the following to display Jobs.
 - Enter the date and time range.
 - Enter a Job Name.
 - Enter a User Name.
 - Select a Job status. The options include Scheduled, Released, Ready, Active, Finished, or Canceled.
5. Click **OK**. The SAP Jobs are filtered and displayed.

NOTE: If you have Jobs in the Active state, the Duration field will not display a value. To view the duration for a Job, use the SAP Client.

Finding an SAP Job

You can search for Jobs in SAP by filtering by one or more criteria.

1. Access the SAP Jobs view.
2. Click the Query icon.
3. On the SAP Job Selection screen, do one or more of the following to find Jobs.
 - Enter the date and time range.
 - Enter a Job Name.
 - Enter a User Name.
 - Select a Job status. The options include Scheduled, Released, Ready, Active, Finished, or Canceled.
4. Click **OK**. The SAP Jobs are filtered and displayed.

Converting an SAP Job

If you want JAMS to manage an existing SAP Job, you can convert it into a Job in JAMS that uses the SAPJobV2 Execution Method.

NOTE: Before you convert a Job, ensure the properties are set properly within SAP.

You will need to delete or disable the scheduling for the current SAP Job in SAP after you convert it. After you convert the Job, it will be displayed in the JAMS Monitor view when the Job runs.

You can convert more than one Job at a time by selecting multiple Jobs and clicking **Convert All**.

NOTE: The Start Page and End Page properties will not be converted for the SAP Job. If you are using those properties, update the properties in the converted JAMS Job.

1. Access the SAP Jobs view.
2. Right-click the Job that you want to convert and select **Convert**. The Convert SAP Job dialog is displayed.
3. In the Folder field, select the folder where the Job will be saved.
4. In the Name field, enter a new name for the Job, if necessary.
5. Optional - In the Description field, enter a description for the Job.
6. In the Edit this job definition after adding field, select the checkbox to review and to make additional changes to the Job. (Fields including the **Connection**, **Execute As Users**, may need to be selected or modified if they were not set at the Folder level. If they were not selected at the Folder level, the values will be empty.)
7. Click **Convert**. The Source tab for the Job is displayed.
8. On the Source tab, review the selected options and make any necessary changes.
9. On the Properties tab, review the selected options and make any necessary changes.
10. Click **Save and Close**.

Deleting an SAP Job

You can delete a Job in SAP from within JAMS. If there are any dependencies to this Job, ensure they are resolved before you delete it.

1. Access the SAP Jobs view.
2. Right-click the Job that you want to delete and select **Delete**.
3. Click **Yes** when prompted. The Job is deleted and is no longer visible in the SAP Client or in the SAP Jobs view in JAMS.

Copying an SAP Job

You can copy a Job in SAP from within JAMS.

1. Access the SAP Jobs view.
2. Right-click the Job that you want to copy and select **Copy**.
3. On the Copy SAP Job dialog, enter a new name for the copy.
4. Click **Ok**.
5. Click the Refresh button to see the copied Job.

Starting an Intercepted SAP Job

If an SAP Job is in the Intercepted state, you can release it from within JAMS and have SAP run it as soon as possible. You can release the Job from JAMS or SAP.

1. Access the SAP Jobs view.
2. Right-click the Job that is in the Intercepted state and select **Release**.
3. Click **Yes** when prompted.
4. Click the Refresh button to view the updated status.

Running SAS Jobs with JAMS

Windows Command Procedure Method

JAMS can execute SAS programs using the Windows Command Procedure Execution Method.

Below is an example of an SAS job-source:

```
c:\YourSASPath\sas.exe
-autoexec d:\sas\autoexec.sas
-config d:\sas\config.sas
-nologo
-log d:\work\logs\pull_sample200102131244.log
-sysin d:\YourDir\YourProgram.sas
```

SAS Execution Method

As an alternative, you can create a SAS execution method from within JAMS. This allows the source tab of a Job definition to contain the contents of your SAS program.

1. Select the Execution Methods shortcut from the Menu.
2. Click the **Add** button from the Control Bar to open the Add an Execution Method dialog.
3. In the Add an Execution Method dialog, give the Execution Method a Name, Description (optional), and Base Method.

NOTE: Use the Command Base Method for SAS.

4. Click **OK**. By default, JAMS will open the full Execution Method Definition dialog when the Execution Method is initially saved.

5. Edit the Parameters, Template, Properties, and Job Properties as necessary.
6. **Save and Close** the Execution Method.

JAMS VBScript Execution Method

SAS Projects (EGP files) can be saved as Visual Basic scripts from the SAS Enterprise Guide and executed using the [JAMS VBScript Execution Method](#).

Symitar in JAMS

Installing the Symitar Integration Pack

If the JAMS Integration Pack for Symitar is not installed, run the JAMS Installer and check the "JAMS Integration Pack for Symitar" component in the Feature Selection step.

NOTE: Un Checking other components will uninstall those components.

NOTE: In some installations, the host.config file must be configured in order to allow a machine to access the Symitar server.

Creating a Symitar Job

1. Create a Job in the Desired folder and choose the **Workflow** Execution Method.
2. Select the desired folder for the Symitar Job, then ensure the Job Definitions tab is selected.
3. Click the **Add** button from the Control Bar to open the Add a New JAMS Job Definition dialog.
4. In the dialog, give the new Job a Name, Description (optional), and Execution Method. In this case, select **Workflow** to create the Symitar Job.
5. By default, the full Job Definition dialog will open when the Job is initially saved. Click **OK**.
6. The Job Definition dialog will open.
7. Define Elements, Parameters, Security, Properties, and Documentation as desired.

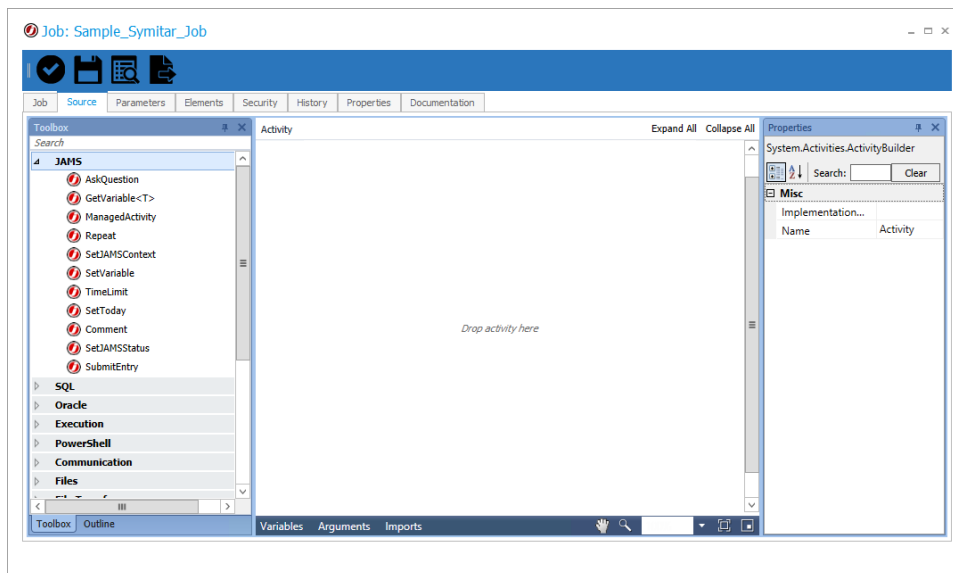
NOTE: An **Execute As** property should be configured with a user that has access to the Symitar Server.

8. Select the **Source** tab, and define the Job using the Workflow Interface.
(Keep reading this document for more information on the Workflow Interface and Symitar Activities.)
9. **Save and Close** the Job Definition dialog.

Defining a Symitar Job using the Workflow Interface

Workflow Interface

The source for a Symitar Job displays the Workflow Interface. This interface is comprised of three panels: the **Toolbox** panel, **Activity** panel, and **Properties** panel. Symitar Activities can be found in the Toolbox panel, under the "SYM" group. Activities will be used to define the Symitar Job.



Toolbox Panel

The Toolbox Panel contains dozens of grouped Activities that can be used in the workflow. These activities are the building blocks of a Workflow Job. Activities may be used exclusively, or combined with other activities to create a workflow.

Activity Panel

The Activity Panel is the Designer section of the workflow. Drag activities from the toolbox to this area to add them to the workflow, or move those activities around in the workflow window. Right-click on items in this panel to get additional options.

Properties Panel

Shows the available properties of any selected element or activity. The properties of each activity can be quickly set from this panel.

Building Symitar Workflows

A single Symitar Activity can be utilized by dragging one of the available activities from the Toolbox Panel into the Activity Panel and setting the properties for that Activity.

To utilize multiple activities in a workflow, JAMS requires a Parent Activity to contain other activities. Note that JAMS will automatically add "Sequences" to the workflow if another parent activity is not set. Sequences may be nested inside of other Sequences or Parent Activities in a workflow.

Automatically Add Sequences

1. With at least one activity in the Activity panel, drag another activity from the toolbox panel to the Activity panel.
2. Hover the new activity over any existing activity. Grey bars will appear to show possible placement locations.
3. Release the activity in the desired location. A sequence will be added if necessary. Note that moving existing activities will also cause the placement location bars to appear, and will also result in the creation Sequences when necessary.

Passing Parameters to a Symitar Job

Typically, executing a Symitar job involves responding to a series of prompts; for example, what is the job name? . . . the queue?

A Symitar job may include other prompts that are unique to the job. When starting a Symitar job, if JAMS encounters an unexpected prompt, it looks for a workflow variable/JAMS parameter named "SymParam1". If this parameter is found, JAMS inserts its value as the response to the prompt. However, if this variable/parameter is not found, JAMS uses the default value defined in the Symitar job.

NOTE: When looking for a "SymParam", JAMS searches first for a Workflow variable with that name. If that variable doesn't exist, it then looks for a JAMS Parameter with that name. Each time JAMS looks for a parameter it increments the number, so the second parameter would be "SymParam2".

Symitar Activities

A small overview of properties and function of each Symitar Activity are listed here. When defining Symitar Activities, it is important to keep in mind:

- Symitar pathways will be case sensitive in UNIX or LINUX environments
- DateTime values are formatted as YYYYMMDD

DataFiletoPC

Downloads a DataFile from the FTP Server to the PC. DataFile is downloaded from the SYMnnn/DATAFILES directory, where nnn is a three-digit SYM Number, such as '007'.

Property	Description
JAMUsername	The User ID for AIX Login. Any JAMS user.
LocalDirectory	The local directory is the download location for the SYM DataFile.
DisplayName	Use to modify the property name as displayed in the activity panel.
Password	The AIX password. Use when overriding the value in the SymConfig Activity. As an alternative, if you wish to hide the password from viewing, (1) create a user in JAMS, (2) set the "Logon As" to the AIX Username, (3) click the password button and (4) enter the password for this account, and (5) then use that JAMS User Name in the JAMUsername workflow property.
ServerFileName	The name of the DataFile on the SYM Server in the '/SYMnnn/DATAFILES' directory.
Servername	The name of the AIX server. Use when overriding the value set in the SymConfig activity.
SymNumber	The Sym Number. Use when overriding the value in the SymConfig activity. This parameter defaults to '000'.
UserName	The AIX UserName. Use when overriding the value in the SymConfig activity.

DeleteLetterFile

Deletes a LetterFile from the '/SYMnnn/LETTERSPECS' directory

Property	Description
JAMUsername	The User ID for AIX Login. Any JAMS user.
DisplayName	Use to modify the property name as displayed in the activity panel.

Property	Description
Password	The AIX Password (Use when overriding the value in the SymConfig Activity). As an alternative, if you wish to hide the password from viewing, create a User in JAMS, set the "Logon As" to the AIX Username, click the password button and enter the password for this account, and then use that JAMS User Name in the JAMSUserName workflow property.
ServerFileName	The name of the LetterFile to delete from the '/SYMnnn/LETTERSPECS' Directory on the SYM Server.
ServerName	The name of the AIX Server. Use when overriding the value set in the SymConfig activity.
SymNumber	The Sym Number. Use when overriding the value in the SymConfig Activity. The parameter defaults to '000'.
UserName	The AIX UserName. Use when overriding the value in the SymConfig activity.

EmailReports

Send a report generated from ReportFileToPC as an email attachment. The report must be assigned to a Workflow Variable of type SymReport.

Property	Description
BCCAddress	The address of user(s) to receive blind carbon copies of the email.
CCAddress	The address of user(s) to receive carbon copies of the email.
FromAddress	The address to be used to send the email.
Reports	The report(s) to be sent as attachments to the email.
Subject	The subject of the email to be sent.
ToAddress	The address of the user(s) to receive the email.
BodyMessage	The body of the message to be sent.
SMTPServer	The SMTP server to be used to email handling.
JAMSUsername	The User ID for AIX Login. Any JAMS user.
DisplayName	Use to modify the property name as displayed in the activity panel.

Property	Description
Password	The AIX Password (Use when overriding the value in the SymConfig Activity). As an alternative, if you wish to hide the password from viewing, create a User in JAMS, set the "Logon As" to the AIX Username, click the password button and enter the password for this account, and then use that JAMS User Name in the JAMSUserName workflow property.
ServerName	The name of the AIX Server. Use when overriding the value set in the SymConfig activity.
SymNumber	The Sym Number. Use when overriding the value in the SymConfig Activity. The parameter defaults to '000'.
UserName	The AIX Username. Use when overriding the value in the SymConfig activity.

FileToPC

Downloads a SYM File from the FTP Server to the PC. The File is downloaded from the specified directory.

Property	Description
JAMSUsername	The User ID for AIX Login. Any JAMS user.
LocalDirectory	The download location for the SYM File.
DisplayName	Used to modify the property name as displayed in the activity panel.
Password	The AIX Password (Use when overriding the value in the SymConfig Activity). As an alternative, if you wish to hide the password from viewing, create a User in JAMS, set the "Logon As" to the AIX Username, click the password button and enter the password for this account, and then use that JAMS User Name in the JAMSUserName workflow property.
ServerDirectory	Server Directly beneath the '/SYMnnn/' folder
ServerFileName	The Name of the file on SYM server in the specified Directory.
ServerName	The Name of the AIX Server. Use when overriding the value in the SymConfig Activity.
SymNumber	The SYM Number. Use when overriding the value in the SymConfig activity. Defaults to '000'
UserName	The AIX Username. Use when overriding the value in the SymConfig activity.

LetterFiletoPC

Downloads a LetterFile from the FTP Server to the PC. The LetterFile is downloaded from the SYMnnn/LETTERSPECS directory, where nnn is a three-digit SYM Number, such as '007'.

Property	Description
JAMSUsername	The User ID for AIX Login. Any JAMS user.
LocalDirectory	The local directory is the download location for the SYM LetterFile.
DisplayName	Used to modify the property name as displayed in the activity panel.
Password	Use when overriding the value in the SymConfig Activity. As an alternative, if you wish to hide the password from viewing, (1) create a user in JAMS, (2) set the "Logon As" to the AIX Username, (3) click the password button and (4) enter the password for this account, and (5) then use that JAMS User Name in the JAMSUserName workflow property.
ServerFileName	The name of the LetterFile on the SYM Server in the '/SYMnnn/LETTERSPECS' directory.
ServerName	The name of the AIX server. Use when overriding the value set in the SymConfig activity.
SymNumber	Use when overriding the value in the SymConfig activity. This parameter defaults to '000'.
UserName	The AIX UserName. Use when overriding the value in the SymConfig activity.

PCToDataFile

Uploads a DataFile from the PC to the FTP server. The DataFile is uploaded to the SYMnnn/DATAFILES directory, where nnn is a three-digit SYM Number, such as '007'.

Property	Description
JAMSUsername	The User ID for AIX Login. Any JAMS user.
LocalDirectory	The local directory of the Data File to send to the '/SYMnnn/DATAFILES' directory on the SYM Server.
LocalFileName	The Local file name of the DataFile to upload to the '/SYMnnn/DATAFILES' Directory on the SYM server.
DisplayName	Used to modify the property name as displayed in the activity panel.

Property	Description
Password	Use when overriding the value in the SymConfig Activity. As an alternative, if you wish to hide the password from viewing, (1) create a user in JAMS, (2) set the "Logon As" to the AIX Username, (3) click the password button and (4) enter the password for this account, and (5) then use that JAMS User Name in the JAMSUserName workflow property.
ServerName	The name of the AIX Server. (Use when overriding the value set in the SymConfig activity).
SymNumber	Use when overriding the value in the SymConfig Activity. This parameter defaults to '000'
UserName	The AIX Username. Use when overriding the value in the SymConfig activity.

PCToEditFile

Uploads an EditFile from a local location to a remote folder via FTP operation

Property	Description
Overwrite	Turn overwriting on or off
JAMSUsername	The User ID for AIX Login. Any JAMS user.
LocalDirectory	Local Directory of the Edit File to send to the '/SYMnnn/REPORT' directory on the server
LocalFileName	Local File Name of the Edit File to send to the '/SYMnnn/REPORT' directory on the server
DisplayName	Used to modify the activity name property as displayed in the activity panel.
Password	Use when overriding the value in the SymConfig Activity. As an alternative, if you wish to hide the password from viewing, (1) create a user in JAMS, (2) set the "Logon As" to the AIX Username, (3) click the password button and (4) enter the password for this account, and (5) then use that JAMS User Name in the JAMSUserName workflow property.
ServerName	The name of the AIX server. Use when overriding the value set in the SymConfig activity.
SymNumber	The Sym Number. Use when overriding the value in the SymConfig activity. The parameter defaults to '000'.
UserName	The AIX Username. Use when overriding the value in the SymConfig activity.

PCToFile

Uploads a file from the PC to the specified directory on the SYM server.

Property	Description
JAMUsername	The User ID for AIX Login. Any JAMS user.
LocalDirectory	The local directory of the file to send to the specified SYM directory
LocalFileName	The local File Name of the file to send to the specified SYM directory
DisplayName	Used to modify the activity name property as displayed in the activity Panel.
Password	Use when overriding the value in the SymConfig Activity. As an alternative, if you wish to hide the password from viewing, (1) create a user in JAMS, (2) set the "Logon As" to the AIX Username, (3) click the password button and (4) enter the password for this account, and (5) then use that JAMS User Name in the JAMSUserName workflow property.
ServerDirectory	Server directory beneath the '/SYMnnn/' folder where the file will be sent
ServerName	The name of the AIX server. Use when overriding the value set in the SymConfig activity.
SymNumber	The Sym Number. Use when overriding the value in the SymConfig activity. The parameter defaults to '000'.
UserName	The AIX UserName. Use when overriding the value in the SymConfig activity.

PCToLetterFile

Uploads a LetterFile from the PC to the FTP Server. The LetterFile is uploaded to the SYMnnn/LETTERSPECS directory, where nnn is a three-digit SYM Number, such as '007'.

Property	Description
JAMUsername	The User ID for AIX Login. Any JAMS user.
LocalDirectory	Local directory of the Letter File to send to the '/SYMnnn/LETTERSPECS' Directory on the Server
LocalFileName	Local File Name of the Letter File to send to the '/SYMnnn/LETTERSPECS' Directory on the Server

Property	Description
DisplayName	Used to modify the activity name property as displayed in the activity panel.
Password	Use when overriding the value in the SymConfig Activity. As an alternative, if you wish to hide the password from viewing, (1) create a user in JAMS, (2) set the "Logon As" to the AIX Username, (3) click the password button and (4) enter the password for this account, and (5) then use that JAMS User Name in the JAMSUserName workflow property.
ServerName	The name of the AIX server. Use when overriding the value set in the SymConfig activity.
SymNumber	The Sym Number. Use when overriding the value in the SymConfig activity. The parameter defaults to '000'.
UserName	The AIX Username. Use when overriding the value in the SymConfig activity.

ReportFileToPC

Downloads a ReportFile from a remote location to a local folder via FTP

Property	Description
JAMSUsername	The User ID for AIX login. Any JAMS User.
LocalDirectory	The Local Directory where the Report File will be located.
DisplayName	Used to modify the property name as displayed in the Activity panel.
LocalFileName	The name for the local report file. The default is the source name.
Password	Use when overriding the value in the SymConfig Activity. As an alternative, if you wish to hide the password from viewing, (1) create a user in JAMS, (2) set the "Logon As" to the AIX Username, (3) click the password button and (4) enter the password for this account, and (5) then use that JAMS User Name in the JAMSUserName workflow property.
Report	The SymReport that identifies the report. This is returned by RunJobFile.
ServerName	The name of the AIX server. Use when overriding the value set in the SymConfig activity.
SymNumber	The Sym Number. Use when overriding the value in the SymConfig activity. The parameter defaults to '000'.
UserName	The AIX Username. Use when overriding the value in the SymConfig activity.

Creating an in-memory copy of the desired report

1. Click the **Variables** tab, located at the bottom of the Activity Panel, then select **Create Variable**.
2. Name the variable the desired report name.
3. Click on the listed Variable Type. A drop-down list will appear. Select **Browse for Types** from the dropdown list.
4. In the dialog that appears, type *SymReport* into the search box or navigate to the SymReport variable type. Select the SymReport variable type and click **OK**.
5. Ensure the Scope is set to Sequence.
6. Leave the "Default" column blank.
7. Select the **ReportFileToPC** activity in the Activity Panel, and set the Report property to the name of the created variable.

RunJobFile

Runs a JobFile on the Symitar System. By default JobFiles are run in synchronous mode which means that the program will wait for the JobFile to finish.

Property	Description
JAMSSymUserID	For this JAMS user, enter the SymUserId as the "logon as" name, and set the password by clicking the "Set Password" button. When JAMS executes a job file, it will assemble the JAMSSymUserID as "username.password" (sans quotes).
JAMSUsername	The User ID for AIX Login. Any JAMS user.
BatchSequenceNumber	Returns the batch sequence number of the Job.
JobFileBatchQueue	A numeric value (0-3) that will determine the batch queue to run the JobFile. Use 99 to find the first available empty queue. 0 will be used if 99 is specified and all queues are busy.
JobFileName	The name of the JobFile to run.
ReportsProduced	Returns a collection of reports produced by the job.
SymOperatorId	Operator ID to log into SYMOP. In most cases this will be \"0000\" or the same as the user's SymUserId

Property	Description
SymUserId	SYM specific userid. Must be in the format of symuserid.password. As an alternative, if you want to hide your symuserid/password from viewing, (1) create a user in JAMS, (2) set the "Logon As" to the symuserid, (3) click the password button and enter the password for this symuserid, and then (4) use that JAMS User Name in the JAMSSymUserID workflow property. JAMS will then correctly format the symuserid and password.
DisplayName	Use to modify the property name as displayed in the Activity panel.
Password	The AIX Password (Use when overriding the value in the SymConfig Activity). As an alternative, if you wish to hide the password from viewing, (1) create a User in JAMS, (2) set the "Logon As" to the AIX Username, (3) click the password button and enter the password for this account, and then (4) use that JAMS User Name in the JAMSUserName workflow property.
ServerName	The name of the AIX server. Use when overriding the value set in the SymConfig activity.
Sym	The Sym Number. Use when overriding the value in the SymConfig Activity. This parameter defaults to '000'.
TelnetPort	Enter the port number. This parameter defaults to '23'.
Username	The AIX UserName. Use when overriding the value in the SymConfig activity.

SymConfig

Creates a parent-child relationship for all SYM activities nested within SymConfig and set the default values for these child activities. If the child activities contain their own settings, those values will be used in place of the values set in the SymConfig activity

Property	Description
JAMSUsername	The User ID for AIX Login. Any JAMS user.
DisplayName	Use to modify the property name as displayed in the Activity panel.
Password	The AIX password. As an alternative, if you wish to hide the password from viewing, (1) create a user in JAMS, (2) set the "Logon As" to the AIX Username, (3) click the password button and (3) enter the password for this account, and then (4) use that JAMS User Name in the JAMSUserName workflow property.

Property	Description
ServerFolderDirectory	The Path of the SYM Server File.
ServerName	The name of the AIX Server
ServerRunTimeoutInSecs	A numeric value (in seconds).
ServerTransferMode	ASCII or Binary.
SymNumber	The Sym Number.
UserName	The AIX UserName.

Still have questions about Symitar or Workflows? Try watching our Workflow video at: <https://www.jamsscheduler.com/videos/>.

NOTE: This video was recorded in JAMS V6. The sample Workflow principles shown there apply to JAMS V7.

z/OS Integration Pack

JAMS for z/OS incorporates z/OS Job scheduling into the JAMS platform and requires nothing to install on the z/OS host. When creating the agent used with the z/OS integration, you must select the Passive agent type and the zOS Type of platform.

Use this step-by-step guide to define a new z/OS Job using the JAMS Client.

Setting up the Connection with JAMS as a TSO User

To set up a connection with JAMS, you will need to first define a TSO user on the mainframe as well as in JAMS (from the Credentials shortcut).

1. Open the **Credential Definitions** view from the JAMS Menu.
2. Click the **Add** button on the Control Bar. The JAMS Credential definition dialog opens.
3. Set the Credential Name, Logon As, Password, and Password Confirmation values.
4. Click **OK**. By default, JAMS will open the full Credential Definition dialog after the Credential definition is initially saved.
5. On the Security tab of the Credential dialog, Add the ACE for the TSO user. This user must have login rights as well as rights to execute items that will be added in later JAMS jobs.
6. **Save and Close** the Credential definition.

For further information on adding a new username in JAMS, go to the topic: [JAMS Security: Working with User Definitions](#).

JAMS interfaces with JES as a TSO user. The TSO login ID is interpreted by JAMS as a valid JAMS User, which is encrypted on the JAMS side and controlled by RACF on the z/OS side. The TSO user must have permissions to submit and receive Jobs on both platforms and the TSO username must contain 7 characters or less.

On the JAMS side, the z/OS Integration must be installed on the JAMS Server.

Setting up a z/OS Job in JAMS

NOTE: The Job name being referred to within the JCL needs to be the same as the Execute As user plus one character. If your credential user in JAMS is called "IBMUUSER", the Job name within the JCL would need to be "IBMUUSERx" (where x is A-Z or 0-9).

1. Select the desired folder for the z/OS Job, and then ensure the Job Definitions tab is selected.
2. Click the **Add** button from the Control Bar to open the Add a New JAMS Job Definition dialog.
3. In the dialog, give the new Job a Name, Description (optional), and Execution Method. In this case, select **z/OS** to create a z/OS Job.
4. By default, the full Job Definition dialog will open when the Job is initially saved.
5. Click **OK**. The Job Definition dialog will open.
6. Define Schedule Items, Parameters, Security, Properties, and Documentation as desired.

NOTE: An Execute As property should be configured with the previously defined TSO user.

NOTE: An Agent property should be configured with the Mainframe Agent hostname or IP address.

7. On the Source tab, insert the JCL code.
8. **Save and Close** the Job definition.

Properties to Configure Connection Retries

A z/OS Job in JAMS has several properties that you can configure to control the number of connection retries and the time in between each retry. The z/OS Max Ftp Retry is useful if FTP exceptions have occurred during the FTP connection. The z/OS Max Status Retry option is useful if z/OS is holding Jobs in a state, such as INPUT, and is delayed in sending responses back to JAMS.

Option	Description
z/OS Max Ftp Retry	Enter the maximum number of retries that JAMS will run to get a successful FTP connection. The default is zero, which sets JAMS to send an unlimited number of retries until a successful FTP connection is made.
z/OS Ftp Retry Interval	Enter the time to wait between each FTP retry. The default is 500 milliseconds.
z/OS Max Status Retry	Enter the maximum number of retries that JAMS will run to get a z/OS status response for the JAMS Job that was sent. This applies to the initial connection to the mainframe. The default is zero, which sets JAMS to send an unlimited number of retries until it receives a response from z/OS.
z/OS Status Retry Interval	Enter the time to wait between each retry for z/OS status response. The default is 5,000 milliseconds.

How does JAMS run a Mainframe Job?

Jobs are stored on an FTP server located on the mainframe using a proprietary IBM-based FTP language. JAMS communicates with the mainframe using this FTP language. Once a Job is executed in JAMS, an FTP connection is opened to the FTP Daemon in JES and the Job is sent to the mainframe and executed.

JAMS monitors the Job spool output and waits for the Job to complete by polling the FTP connection. Once the output files are complete, JAMS purges the files from the JES spool.

Steps can be run in JES in the same way as tasks in a JAMS Sequence. JCL defines the behavior of the Job depending on what is happening with a particular Job task.

What Information does JES Return?

The JAMS Scheduler retrieves all Job related output from the JES Spool. This includes any DD statements with SYSOUT held on the spool. The output appears in the JAMS log and can be used by operations administrators to debug problems and, if necessary, restart the Job.

JAMS for z/OS parses the Job output and provides job success (“0”) or failure information for the JAMS Scheduler to react and report on a variety of message codes, including HASP and return code analysis.

What happens when a Job fails?

If a Job fails due to a network outage or other problem, all relevant information is included in the output and also displayed in the JAMS log. Customization of these messages can be set using a configuration file.

A special option in the Job definition includes a listing of Jobs on the JES Spool allowing a Job to rerun on the JAMS side for output recovery.

JAMS for z/OS Capabilities

Manages and Submits JCL Files to JES2 on z/OS for Execution

JAMS provides the ability to store a JCL source in its internal database. The JCL then becomes a part of the JAMS Job Definition along with other parameters and information needed to run the job.

Schedules Jobs with Dependencies on Multiple z/OS Hosts

The JAMS Scheduler can support multiple z/OS hosts and track Job dependencies between them and non-z/OS hosts.

The JAMS Scheduler Retrieves and Displays the Job Output

The JAMS Scheduler retrieves all job related output from the JES Spool. This includes any DD statements with SYSOUT held on the spool. The output appears in the JAMS log and can be used by operations administrators to debug problems and, if necessary, restart the jobs.

Includes Automatic Determination for all Job Results

JAMS for z/OS parses the job output and provides job success or failure information for the JAMS Scheduler to react and report on a variety of message codes, including HASP and return code analysis.

Utilizes Standard JES Settings

JAMS for z/OS uses JESINTERFACELEVEL 1 and doesn’t require special customization of Parmlib settings.

Requires No installation on z/OS Hosts

JAMS for z/OS uses z/OS FTP with JES and requires no additional mainframe software installation.

Provides Automatic Job Purging Capabilities

JAMS for z/OS automatically purges the job output files for the JES2 Spool after each Job has completed and been retrieved.

Provides Reliable Tracking of Jobs Running on the Mainframe

The JAMS Scheduler can recover its connection to JES if it is lost due to a network outage or other problem, even when a job is in the middle of executing on z/OS. In the event of failure on the JAMS server, JAMS can restart for that specific job and seamlessly recover.

Uses a Secure TSO User Login

JAMS interfaces with JES as a TSO user. The TSO login ID is interpreted by JAMS as a valid JAMS user and encrypted on the JAMS side and controlled by RACF on the z/OS side.

Provides Full Integration with the JAMS Scheduler

JAMS for z/OS becomes part of the integrated capabilities of the JAMS Scheduler across different platforms allowing it to control jobs based on dependencies on all supported hosts.

JAMS for z/OS is Easy to Set Up

Setting up a z/OS job is now as easy as defining any other JAMS Job.

Provides a Listing of Pending Jobs on the z/OS Spool

A special option in the JOB definition includes a listing of JOBS on the JES Spool allowing a job to rerun on the JAMS side for output recovery.

Scenario: Synchronizing Data with JAMS for z/OS

Suppose an organization has the need to reliably synchronize data from an OLTP system based on a MS SQL Server or Oracle with a mainframe-based database. This scenario is easily implemented with JAMS for z/OS.

In order to accomplish this, the JAMS Scheduler would initially run a job, based on an event, that generates files comprised of data extracted from a MS SQL Server database. It would then upload those files to the z/OS platform via FTP before submitting JCL through JAMS for z/OS. These uploaded files would then update a mainframe database. The reverse scenario is just as easy where a MS SQL Server is updated with files generated by the mainframe.