IBM OpenPages with Watson
Version 8.2.0

Upgrade Guide for Oracle users

Note

Before using this information and the product it supports, read the information in “Notices” on page 55.

Product Information

This document applies to IBM OpenPages with Watson Version 8.2.0 and may also apply to subsequent releases.

Contents

Chapter 1. Introduction

IBM OpenPages® with Watson™ is an integrated governance, risk, and compliance platform that enables companies to manage risk and regulatory challenges across the enterprise.

Audience

This guide provides instructions for upgrading OpenPages with Watson deployments that use Oracle. Use this guide if you are upgrading OpenPages with Watson in-place. If you are migrating or doing a fresh installation, see the *IBM OpenPages with Watson Installation and Deployment Guide*.

Please read the following important information regarding IBM OpenPages with Watson documentation

IBM® maintains one set of documentation serving both cloud and on-premises IBM OpenPages with Watson deployments. The IBM OpenPages with Watson documentation describes certain features and functions which may not be available on the cloud.

If you have any questions about the functionality available in the product version that you are using, contact IBM OpenPages Support by using the IBM Support portal.

Finding information

To find product documentation on the web, including all translated documentation, access IBM Documentation.

Accessibility features

Accessibility features help users who have a physical disability, such as restricted mobility or limited vision, to use information technology products. OpenPages documentation has accessibility features. PDF documents are supplemental and include no added accessibility features.

Installation locations for IBM OpenPages with Watson

The installation directory is the location of product artifacts after a package, product, or component is installed. The following table lists the conventions that are used to refer to the installation location of installed components and products:

Important: Directory locations that contain spaces are not supported. IBM OpenPages with Watson or any software that is used by it must not be installed into a directory with spaces. For example, do not install database server, database client, or application server software into the `Program Files` directory.

If you're using IBM OpenPages for IBM Cloud Pak for Data, see the *IBM OpenPages with Watson Administrator's Guide*.

Table 1. Variable notations for installation directories	
Directory	**Description**
<installation_server_home>	The directory where the IBM OpenPages with Watson installation server is installed. For example: • On Windows: `C:\IBM\OPInstall\OP_<version>_Installer` • On Linux®: `/home/opuser/IBM/OPInstall/OP_<version>_Installer`
<agent_home>	The directory where the IBM OpenPages with Watson installation agent is installed on a remote server. For example: • On Windows: `C:\IBM\OPAgent` • On Linux: `/home/opuser/IBM/OPAgent`
<OP_HOME>	The directory where OpenPages with Watson is installed. For example: • On Windows: `C:\IBM\OpenPages` • On Linux: `/opt/opuser/IBM/OpenPages` In the installation app, you specify the *<OP_HOME>* directory in the **OP Home Directory** field each **Application Server** card.
<ORACLE_HOME>	The installation location of the Oracle database software. For example: • On Windows: – `C:\app\oracle\product\19.0.0\client_1` (Oracle Admin Client) – `C:\oracle\instantclient_19_9` (Oracle Instant Client) – `C:\app\oracle\product\19.0.0\dbhome_1` (server) • On Linux: – `/home/oracle/app/oracle/product/19.0.0/client_1` (Oracle Admin Client) – `/home/oracle/instantclient_19_9` (Oracle Instant Client) – `/home/oracle/app/oracle/product/19.0.0/dbhome_1` (server)
<WLP_HOME>	The installation location of IBM WebSphere® Liberty. For example: • On Windows: *<OP_HOME>*`\wlp` • On Linux: *<OP_HOME>*`/wlp`

Table 1. Variable notations for installation directories (continued)	
Directory	**Description**
<WLP_USER_HOME>	The location of OpenPages with Watson application files and server configuration files. For example: • On Windows: *<OP_HOME>*`\wlp-usr` • On Linux: *<OP_HOME>*`/wlp-usr`
<COGNOS_HOME>	The installation location of Cognos® Analytics. For example: • On Windows: `C:\IBM\cognos\analytics` • On Linux: `/usr/IBM/cognos/analytics`
<JAVA_HOME>	The installation location of IBM SDK, Java™ Technology Edition or Java Runtime Environment (JRE). IBM SDK example on an application server: • On Windows: `C:\IBM\java_8.0_64` • On Linux: `/opt/IBM/java_8.0_64` JRE example on a reporting server where Cognos Analytics is installed: • On Windows: `C:\IBM\cognos\analytics\jre` • On Linux: `/usr/IBM/cognos/analytics/jre` **Note:** In Cognos Analytics 11.1.5 and later, the path is: • On Windows: `C:\IBM\cognos\analytics\ibm-jre\jre` • On Linux: `/usr/IBM/cognos/analytics/ibm-jre/jre` IBM SDK example on a search server: • On Windows: `C:\IBM\java_8.0_64\` • On Linux: `/opt/IBM/java_8.0_64/`
<CC_HOME>	The installation location of OpenPages with Watson CommandCenter. For example: • On Windows: `C:\IBM\OpenPages\CommandCenter` • On Linux: `/opt/IBM/OpenPages/CommandCenter`
<SEARCH_HOME>	The installation location of global search. The *<SEARCH_HOME>* directory contains the `opsearchtools.jar`, Apache Solr, and other global search files. The global search indexing directory is also stored in the *<SEARCH_HOME>* directory. For example: • On Windows: `C:\IBM\OpenPages\OPSearch` • On Linux: `/opt/IBM/OpenPages/OPSearch` In the installation app, you specify the *<SEARCH_HOME>* directory in the **Search Home Directory** field on the **Search Server** card.

Changes to the installation process

If you installed previous versions of IBM OpenPages with Watson, you will notice many differences. The installation process has changed to make it easier to install and maintain IBM OpenPages with Watson.

IBM OpenPages with Watson now uses IBM WebSphere Liberty. When you install OpenPages, WebSphere Liberty is automatically installed and configured for you.

The following sections describe the main changes to OpenPages on WebSphere Liberty.

Deployment manager

You no longer need to set up a deployment manager for OpenPages.

When you migrate, the installer server updates the `deploy.properties` file automatically. You do not need to edit the file to remove the deployment manager. When you open your deployment in the installation app, review each card, enter passwords, and then continue with the migration.

The admin application server is still *AppServer1*.

Nodes and cells

WebSphere Liberty does not use "nodes" or "cells". Each horizontal cluster member is its own instance of WebSphere Liberty. Vertical cluster members share the same instance of WebSphere Liberty.

If you have a shared cell deployment, you can upgrade or migrate to 8.2, and then do some manual steps to remove OpenPages from the cell.

WebSphere installation user (`wasuser`)

You no longer need the `wasuser` operating system user account. OpenPages installs WebSphere Liberty with the `opuser` account.

The WebSphere username and password are no longer required by tools and utilities, such as OPBackup.

File locations

Table 2. File locations

	Pre-8.2	**8.2.0 and later**
Application server runtime	*<WAS_HOME>*	*<OP_HOME>*/wlp
OpenPages application files	*<OP_Home>*/profiles/<node>/installedApps/<cell>/op-apps.ear	*<OP_HOME>*/wlp-usr/shared/apps/op-apps.ear
Server profiles	*<OP_Home>*/profiles/*<node>*/servers/*<profile>* Where *<profile>* was configured in the WebSphere Administrative Console	*<OP_HOME>*/wlp-usr/servers
Server logs	*<OP_Home>*/profiles/*<node>*/logs/<server>	*<OP_HOME>*/wlp-usr/servers/*<server_name>*Server*<#>*/logs

The WebSphere Liberty documentation uses the placeholder `${server.ouput.dir}`. In OpenPages, the equivalent directory is *<OP_HOME>*/wlp-usr/servers/*<server_name>*Server*<#>*.

For example, ${server.ouput.dir}/logs is the *<OP_HOME>*/wlp-usr/servers/*<server_name>*Server*<#>*/logs directory on an OpenPages application server.

Environment variables

OpenPages configures the following environment variables for WebSphere Liberty:

- *<WLP_HOME>*: This directory is where WebSphere Liberty is installed on the application server.
- *<WLP_USER_HOME>*: This directory is where OpenPages application files and server configuration files are stored. The application and configuration files are stored in a separate directory to simplify updates to WebSphere Liberty.

Java

You need to install IBM SDK, Java Technology Edition on each application server before you install OpenPages. You can get the IBM SDK from the OpenPages installation package.

Starting and stopping application servers

You use the following scripts to start and stop application servers: startAllServers.sh|.cmd and stopAllServers.sh|.cmd. The stopAllServers.sh|.cmd script no longer requires a username and password.

The following scripts are no longer used:

- startManager.sh|.cmd, stopManager.sh|.cmd
- startNode.sh|.cmd, stopNode.sh|.cmd
- startServer.sh|.cmd, stopServer.sh|.cmd

For Microsoft Windows, the OpenPages service is now called: *<OpenPages_server_name>*Server*#*. The following services are no longer used:

- IBMWAS*<version>*Service - *<OpenPages_dmgr_name>*
- IBMWAS*<version>*Service - *<OpenPages_node_name>*
- IBMWAS*<version>*Service - *<OpenPages-node-name>*Server*<#>*

Application server log files

Application server activity, including server startup, is now logged in the following file: *<OP_HOME>*/wlp-usr/servers/*<server_name>*Server*<#>*/logs/messages.log.

The startServer.log and SystemOut.log files are no longer used.

Application server configuration

In previous releases, you used the IBM WebSphere Integrated Solutions Console to configure application server settings. You now use the following files to configure application server properties.

- *<OP_HOME>*/wlp-usr/servers/*<server_name>*Server*<#>*/bootstrap.properties: This file contains server properties, such as the OpenPages application port number.
- *<OP_HOME>*/wlp-usr/servers/*<server_name>*Server*<#>*/configDropins/overrides/jvm.options: Use this file to customize the options for the JVM, such as Java heap size.
- *<OP_HOME>*/wlp-usr/servers/*<server_name>*Server*<#>*/configDropins/overrides/op-apps.xml: Use this file to customize OpenPages, for example to change the context root, configure single sign-on, set up TLS/SSL, and so on.

If you previously customized the web.xml, application.xml, or settings in the IBM WebSphere Integrated Solutions Console, you need to re-apply the configurations in WebSphere Liberty.

Application server tuning

In previous releases, it was necessary to configure the OpenPages application servers to avoid timeouts, Java heap errors, and other issues. You no longer need to do this task. The application server tuning parameters are set when you install OpenPages. You can adjust the settings if needed, however.

Location for customized JSPs

Previously, customized JSPs were stored in the following locations:

```
<OP_HOME>/profiles/<OpenPages-node-name>/installedApps/
  <OpenPages-cell-name>/op-apps.ear/openpages.war
```

```
<OP_HOME>/profiles/<OpenPages-node-name>/installedApps/
  <OpenPages-cell-name>/op-apps.ear/publishweb.war
```

```
<OP_HOME>/profiles/<OpenPages-node-name>/installedApps/
  <OpenPages-cell-name>/op-apps.ear/sosa.war
```

Now, store your customized JSPs in the following locations:

```
<OP_HOME>/wlp-usr/shared/apps/op-apps.ear/openpages.war
```

```
<OP_HOME>/wlp-usr/shared/apps/op-apps.ear/publishweb.war
```

```
<OP_HOME>/wlp-usr/shared/apps/op-apps.ear/sosa.war
```

Keystore on application servers

The installation process creates a default keystore: *<OP_HOME>*/wlp-usr/servers/*<server_name>*Server*<#>*/resources/security/key.p12. The initial password of the keystore is the same as the OpenPagesAdministrator password that you set when you install OpenPages. You can change the keystore password. For more information, see the *IBM OpenPages with Watson Administrator's Guide*.

Workflow Server card removed

The **Workflow Server** card is no longer available in the installation app because the functionality to integrate IBM OpenPages with Watson with IBM Business Process Manager was removed.

Other changes

- The J2EE libraries are stored in *<WLP_HOME>*/dev/api/spec.

Special characters in passwords

You can use certain special characters in certain passwords.

If you are upgrading or migrating from 8.1.0.1 or earlier, install the 8.2 installation server, complete the upgrade or migration process, and then update passwords to use special characters.

The special characters that you can use in passwords are:

```
. + - [ ] * ~ _ # : ?
```

Note: Spaces are not supported.

You can use these special characters in database user passwords and operating system accounts for database schema owners.

If you use special characters in passwords, you must surround the password in quotation marks. Use the following syntax:

Oracle connection strings

For Oracle databases, when you provide a password in a connection string, use \" around the password. For example:

```
sqlplus sys/\"DB~Password\"@op as sysdba
```

Oracle script parameters in SQL*Plus

For Oracle databases, when you provide a password in a script parameter, use the following syntax:

- On Windows, use double quotation marks around the password.

```
sqlplus /nolog @sql-wrapper.sql
 update-storage c:\temp\upd-storage-output.log
 op openpages "pass~word" LFS eng11 eng11
 Windows c:\OpenPages\openpages-storage
```

- On Linux, use single quotation marks around the password.

```
sqlplus /nolog @sql-wrapper.sql
 update-storage /home/op/upd-storage-output.log
 op openpages 'pass~word' LFS  eng11 end11
 Unix /usr/opdata/openpages-storage
```

Installation scripts, tools, and utilities

For tools and utilities that take the password as a parameter, use the following syntax:

- On Windows, use double quotation marks around the password.

```
op-validate-dba-install.bat "DB~Password"
```

- On Linux, use single quotation marks around the password.

```
./op-validate-dba-install.sh 'DB~Password'
```

Passwords in property files

For `.env` files and `.properties` files, do not use any quotation marks around passwords.

Chapter 2. Determine your upgrade path

Use this list to determine your options for upgrading IBM OpenPages with Watson.

If your source environment is at version 7.4.x, 8.0.x, or 8.1.x
You have two options:

- Upgrade (Also called an "in-place" upgrade or an "over the top" upgrade.)

  With this option, you install version 8.2 on top of your existing deployment. See Chapter 3, “Prepare for the upgrade,” on page 11.

- Migration upgrade

  With this option, you do a fresh installation of 8.2 and then migrate files and data. See the *IBM OpenPages with Watson Installation and Deployment Guide*.

  Use this option, for example, if you want to use new hardware.

If your source environment is at version 7.3.x
You must do a migration upgrade. See the *IBM OpenPages with Watson Installation and Deployment Guide*.

If your source environment is at version 7.2.x or earlier
You must first migrate to 7.4.x, 8.0.x, or 8.1.x. You can then upgrade or migrate to 8.2. Or you can migrate to 7.3.x and then migrate to 8.2.

Chapter 3. Prepare for the upgrade

Prepare to upgrade IBM OpenPages with Watson.

Note: These topics apply to in-place upgrades only. If you are migrating to 8.2, see the *IBM OpenPages with Watson Installation and Deployment Guide*. For more information about upgrade paths, see Chapter 2, “Determine your upgrade path,” on page 9.

Review new features and fixes

Before you upgrade OpenPages, review new features and fixes.

For more information about new features, see the latest version of the New Features Guide.

For additional information about OpenPages, see the latest version of the Release Notes.

You can find information about defect corrections on the OpenPages with Watson Fix List.

Make sure that you review the following information before you upgrade: Critical installation and configuration issues for IBM OpenPages GRC.

Backing up your environment

Before you upgrade, back up IBM OpenPages with Watson.

About this task

When you upgrade, the installation server automatically backs up most files for you. Some files need to be backed up manually, however. You also need to back up the databases, the `openpages-storage` directory, and any files that you customized, such as reports and JSPs.

Note: Version 8.1 introduced changes to the database statistics collection method for Oracle. If you customized the `<OP_HOME>/aurora/bin/collect-schema-stats.sql` script, back up the file if you want to keep your customizations. After the upgrade is complete, you can restore your customizations.

Procedure

1. Stop the application servers (admin and non-admin), reporting servers (active and standby), database server, and the search server (if you use global search).
2. Back up the OpenPages database.

   For more information, see “Backing up the OpenPages database (Oracle)” on page 12.
3. Back up the Cognos content store.

   For more information, see “Backing up the Cognos content store (Oracle)” on page 13.
4. Back up the `openpages-storage` directory.

   The `openpages-storage` directory can be located on a server in your deployment or it can be on a separate network share.

   The default location is *<OP_HOME>*`/openpages-storage`.
5. If you modified the `web.xml`, `application.xml`, or if you customized settings in the IBM WebSphere Integrated Solutions Console, make a note of your changes.

   After the migration is complete, you need to re-implement your changes.
6. On each application server, as the OpenPages installation user (`opuser`), create a backup of the top level `OpenPages` directory. Name the backup `OpenPages-`*`<current-version>`*. For example, if you are upgrading from version 8.1.0.1, name the backup `OpenPages-8101`.

   Note: Do not remove or rename the `OpenPages` directory.

If you made changes to the <OP_HOME>/aurora/conf/Server<#>-server.properties and Server<#>-sosa.properties files, you can use the backups to restore you changes after the upgrade.

You can also use this backup directory to restore the current OpenPages version if you need to roll back the upgrade.

7. On each reporting server, as the OpenPages installation user (opuser), create a backup of the top level CommandCenter directory. Name the backup CommandCenter-*<current-version>*. For example, if you are upgrading from version 8.1.0.1, name the backup CommandCenter-8101.

   Note: Do not remove or rename the CommandCenter directory.

   You can use this backup directory to restore the current *<CC_HOME>* directory if you need to roll back the upgrade.
8. On the search server, as the OpenPages installation user (opuser), create a backup of the top level OpenPages directory. Name the backup OpenPages-Search-*<current-version>*. For example, if you are upgrading from version 8.1.0.1, name the backup OpenPages-Search-8101.

   Note: Do not remove or rename the OpenPages directory.

   You can use this backup directory to restore the current search server version if you need to roll back the upgrade.
9. If you modified the standard reports that are provided with OpenPages, copy them to a backup folder or to your personal folders.

   OpenPages standard reports can be overwritten when you upgrade.

   After the upgrade is complete, you can change the reports and restrict access to them.
10. If you customized other files, such as JSPs, back up the files.

Backing up the OpenPages database (Oracle)

Run the OPBackup utility to back up the IBM OpenPages with Watson database.

About this task

Use this procedure if your OpenPages database is at version 7.4.x or later. If you are backing up a 7.3 database, see the *IBM OpenPages with Watson Installation and Deployment Guide*.

Run the OPBackup utility with the dbonly parameter.

Note:

You can back up the databases by using other methods. Some examples of alternative methods include:

- Doing a full physical backup by using RMAN
- Doing a combination of full and incremental backup by using RMAN
- Doing an Oracle data pump export.

If you want to use an alternative method, it is critical that you have the necessary skills available within your organization to complete all aspects of the backup and restore activity.

For more information about backing up your environment, see the *IBM OpenPages with Watson Administrator's Guide*.

Procedure

1. Make sure that no OpenPages with Watson processes are running, such as object reset jobs.
2. Shut down all OpenPages components: application servers (admin and non-admin), reporting servers (active and standby), and the search server.

   For more information, see the *IBM OpenPages with Watson Installation and Deployment Guide*.
3. Open a command or shell window on the admin application server.

4. Go to the *<OP_HOME>*/aurora/bin directory.
5. Do a full database backup of the OpenPages schema by using OPBackup.

   Windows:

   ```
   OPBackup.cmd <backup_directory> dbonly
   ```

   Linux:

   ```
   ./OPBackup.sh <backup_directory> dbonly
   ```

   The *<backup_directory>* is the full path to a directory on the database server. This directory is where the log files are saved. If the file path is not specified, the OPBackup command uses the location that is specified by the **BACKUP_LOCATION** parameter in the *<OP_HOME>*/aurora/bin/op-backup-restore.env file.

   A dump file is created in the OP_DATAPUMP_DIRECTORY directory.

   To find the OP_DATAPUMP_DIRECTORY directory, run the following SQL as the system user:

   ```
   select directory_name, directory_path from dba_directories
   where directory_name = upper ('OP_DATAPUMP_DIRECTORY');
   ```

6. Examine the backup log and make note of the dump file name. The naming convention is openpage_*<timestamp>*.dmp.

Backing up the Cognos content store (Oracle)

You can use OPCCBackup to back up the Cognos content store.

About this task

Run the OPCCBackup utility with the dbonly parameter.

Note: You can back up the content store by using other methods. Some examples of alternative methods include:

- Doing a full physical backup by using RMAN
- Doing a combination of full and incremental backup by using RMAN
- Doing an Oracle data pump export.

If you want to use an alternative method, it is critical that you have the necessary skills available within your organization to complete all aspects of the backup and restore activity.

For more information about backing up your environment, see the *IBM OpenPages with Watson Administrator's Guide*.

Procedure

1. Make sure that no OpenPages with Watson processes are running, such as object reset jobs.
2. Shut down all OpenPages components: application servers (admin and non-admin), reporting servers (active and standby), and the search server.

   For more information, see the *IBM OpenPages with Watson Installation and Deployment Guide*.
3. Ensure that all Cognos components are shut down.
4. Open a command or shell window on the admin application server in your source environment.
5. Go to the *<OP_HOME>*/CommandCenter/tools/bin directory.
6. Do a full database backup of the Cognos schema by using OPCCBackup.

   Windows:

   ```
   OPCCBackup.cmd <backup_directory> dbonly
   ```

Linux:

```
./OPCCBackup.sh <backup_directory> dbonly
```

The *<backup_directory>* is the full path to a directory on the database server. This directory is where the log files are saved. If the file path is not specified, the OPCCBackup command uses the location that is specified by the **OP_CC_BACKUP_HOME** parameter in the *<CC_HOME>*/tools/bin/op-cc-backup-restore.env file.

A dump file is created in the OP_DATAPUMP_DIRECTORY directory. The file is called openpage_cc_*<timestamp>*.dmp.

To find the OP_DATAPUMP_DIRECTORY directory, run the following SQL as the system user:

```
select directory_name, directory_path from dba_directories
where directory_name = upper ('OP_DATAPUMP_DIRECTORY');
```

If you get the warning tools.jar file is not found, see the following technote.

Backing up solutions helpers, images, and other files

Back up the solutions helpers, images, and custom deliverables.

About this task

Do this task if any of the following conditions apply:

- You installed the solutions schema
- You received custom deliverables from the OpenPages Technical Services Team
- You have custom code

Procedure

1. Create a backup directory.
   For example, C:\OpenPages*<current_version>*\patch\helper_backup.
2. Copy your custom helper JSPs, images, and custom code to the backup directory so that you can restore them later.

Upgrade prerequisite software

Before you upgrade, update the software that is required by IBM OpenPages with Watson

Review the software prerequisites for application servers, reporting servers, the database server, and the search server. For more information, see the *IBM OpenPages with Watson Installation and Deployment Guide*.

Important: Do not uninstall IBM WebSphere Application Server Network Deployment. After the upgrade is complete, you can do an optional task to remove it.

Required

- Upgrade to a supported version of Oracle. Version 12.2.0.1 is the minimum supported version for OpenPages. You can also upgrade to Oracle 18c or 19c. See “Oracle upgrade options and Oracle PDB” on page 15.
- Update Cognos Analytics to version 11.1.3 or a later continuous release. You can do an in-place upgrade, also called an "over the top" upgrade. See “Upgrading Cognos” on page 18.
- Install IBM SDK, Java Technology Edition on each application server and on the search server. For more information, see “Getting a copy of the IBM SDK (Windows)” on page 20 or “Getting a copy of the IBM SDK (Linux)” on page 21.

Optional

- Fix pack 8.2.0.2 adds support for Oracle Instant Client. If you want to use it, see the *IBM OpenPages with Watson Installation and Deployment Guide*.

Also, ensure that your users have a supported browser.

If you use optional apps and components, such as IBM OpenPages SDI Connector for UCF Common Controls Hub, you can update them after you upgrade OpenPages.

Oracle upgrade options and Oracle PDB

You have several options when you upgrade IBM OpenPages with Watson.

Upgrade Oracle

You can upgrade Oracle to version 12.2.0.1, 18c, or 19c.

Oracle 12.x

If you are using Oracle 12.1.0.2, you must upgrade to a supported version of Oracle before you upgrade OpenPages. You can upgrade to 12.2.0.1, 18c, or 19c. For information, see the *IBM OpenPages with Watson Installation and Deployment Guide*.

If you are using Oracle 12.2.0.1, you can upgrade to Oracle 18c or 19c.

Oracle 18c

If you are upgrading from OpenPages 8.1.0.0 or later, you can upgrade to Oracle 18c at any time.

If you are upgrading from 7.4.x or 8.0.0.1, you can upgrade Oracle to 18c during or after the upgrade to OpenPages 8.2. Your deployment temporarily uses a configuration that is not supported for end users. Do not allow end users to log in to OpenPages until you complete the upgrade to 8.2. Versions of OpenPages before 8.1 do not support Oracle 18c.

When you upgrade Oracle, you have two options:

- Upgrade Oracle in-place. See “Upgrading Oracle from 12.x to 18c or 19c (in-place)” on page 16.
- Install Oracle and migrate the database. See “Upgrading Oracle from 12.x to 18c or 19c (migration)” on page 17.

Oracle 19c

If you are upgrading from OpenPages 8.1.0.1 or later, you can upgrade to Oracle 19c at any time.

If you are upgrading from 7.4.x, or 8.0.x, or 8.1.0.0, you can upgrade Oracle to 19c during or after the upgrade. Your deployment temporarily uses a configuration that is not supported for end users. Do not allow end users to log in to OpenPages until you complete the upgrade. Versions of OpenPages before 8.1.0.1 do not support Oracle 19c.

When you upgrade Oracle, you have two options:

- Upgrade Oracle in-place. See “Upgrading Oracle from 12.x to 18c or 19c (in-place)” on page 16.
- Install Oracle and migrate the database. See “Upgrading Oracle from 12.x to 18c or 19c (migration)” on page 17.

Oracle PDB (multitenant)

You can use a pluggable database (PDB) for the OpenPages database. OpenPages supports Oracle PDB with Oracle 12.2.0.1, 18c, and 19c.

You can upgrade OpenPages, create a container database (CDB), and then import the OpenPages database into a blank pluggable database.

For information, see the *IBM OpenPages with Watson Installation and Deployment Guide*.

Upgrading Oracle from 12.x to 18c or 19c (in-place)

You can upgrade Oracle to version 18c (18.x) or 19c (19.x). Use these steps if you want to upgrade Oracle by installing on top of your existing Oracle deployment (in-place upgrade).

About this task

This topic provides an overview of the upgrade process. For more information, see the Oracle installation and upgrade guides.

Talk to your Oracle database administrator (DBA) before you begin this procedure.

Note: If you use special characters in database passwords, before you upgrade, ensure that your database passwords do not contain the @ character.

Procedure

1. Do the Oracle pre-upgrade steps and check that your system meets the installation prerequisites.

   For more information, see the Oracle Database Upgrade Guide.
2. Stop all OpenPages application servers (admin and non-admin), all reporting servers (active and standby), and the search server.

   For more information, see the *IBM OpenPages with Watson Installation and Deployment Guide*.
3. Back up the OpenPages and Cognos databases by using the OPBackup and OPCCBackup utilities.

   For more information, see:

   - “Backing up the OpenPages database (Oracle)” on page 12
   - “Backing up the Cognos content store (Oracle)” on page 13
4. Install the new version of Oracle.

   For more information, see the *IBM OpenPages with Watson Installation and Deployment Guide* and the Oracle documentation.
5. Go to the ORACLE_HOME directory, and then start the **Database Upgrade Assistant**.

   - Windows: Click **Start** > **Oracle *<HOME_NAME>*** > **Configuration and Migration Tools** > **Database Upgrade Assistant**.
   - Linux: Run dbua from the *<ORACLE_HOME/bin* directory. For example:

     ```
     cd /home/oracle/app/product/19.0/bin/
     ./dbua
     ```
6. Upgrade the databases.

   Confirm the options with your Oracle database administrator. Typically, the default values can be used, with the following exceptions:

   a) Select the OpenPages database and the Cognos database. Type the sysdba credentials. Click **Next**.

      Check the list of required and recommended actions. Resolve any issues.

   b) On the **Select Upgrade Options** tab, use the default options.

   c) On the **Select Recovery Options** tab, use the options that are recommended by your Oracle database administrator.

      To skip backups completely, select **I have my own backup and restore strategy**.

   d) On the **Configure Network** and **Configure Management** tabs, use the default options.

   e) Review the **Summary** page. Verify that **Target Oracle Home** and **Source Oracle Home** are correct. Verify the other values on the **Summary** page, and then click **Next**.
7. Do the Oracle post upgrade steps.

   For more information, see Post-Upgrade Tasks for Oracle Database

8. Do the following steps on all application servers and all reporting servers:
   a) Install the new version of the Oracle Client software.
      Use the same version as the Oracle Database software.
   b) Update the *<ORACLE_HOME>* environment variable to point to the upgraded Oracle installation.
   c) Copy the following files from your prior Oracle Client installation to the upgraded Oracle Client installation:
      - `sqlnet.ora` (if it exists)
      - `tnsnames.ora`

      Verify that the HOST parameter in the `tnsnames.ora` file is set to the host name of your upgraded Oracle server.
9. Do the following steps on the active reporting server:
   a) Log on to the active reporting server as a user with administrative privileges.
   b) Stop the `IBMOpenPagesFrameworkModelGenerator` service.
   c) Go to the `<CC_HOME>/framework/conf` directory.
   d) Open the `framework.properties` file in a text editor. Ensure that the `oracle.client.path` property contains the location of the new Oracle Client `bin` directory.
   e) Save and close the file.
   f) Restart the `IBMOpenPagesFrameworkModelGenerator` service.
10. Start all OpenPages application servers (admin and non-admin), all reporting servers (active and standby), and the search server.

    For more information, see the *IBM OpenPages with Watson Installation and Deployment Guide*.

Upgrading Oracle from 12.x to 18c or 19c (migration)

You can install Oracle 18c (18.x) or 19c (19.x) on new hardware and then migrate the OpenPages and Cognos databases.

About this task

If you want to upgrade Oracle in-place, see "Upgrading Oracle from 12.x to 18c or 19c (in-place)" on page 16.

Note: If you use special characters in database passwords, before you upgrade, ensure that your database passwords do not contain the @ character.

Procedure

1. Do the Oracle pre-upgrade steps and check that your system meets the installation prerequisites.
   For more information, see the Oracle Database Upgrade Guide.
2. Install the new version of Oracle.
   For more information, see the *IBM OpenPages with Watson Installation and Deployment Guide* and the Oracle documentation.
3. Stop all OpenPages application servers (admin and non-admin), all reporting servers (active and standby), and the search server.

   For more information, see the *IBM OpenPages with Watson Installation and Deployment Guide*.
4. Back up the OpenPages and Cognos databases by using the OPBackup and OPCCBackup utilities.

   For more information, see:
   - "Backing up the OpenPages database (Oracle)" on page 12
   - "Backing up the Cognos content store (Oracle)" on page 13

5. Restore the OpenPages and Cognos databases by using the OPRestore and OPCCRestore utilities.
6. Do the following steps on all application servers and all reporting servers:
   a) Install the new version of the Oracle Client software.
      Use the same version as the Oracle Database software.
   b) Update the *<ORACLE_HOME>* environment variable to point to the upgraded Oracle installation.
   c) Copy the following files from your prior Oracle Client installation to the upgraded Oracle Client installation:
      - `sqlnet.ora` (if it exists)
      - `tnsnames.ora`

      Verify that the HOST parameter in the `tnsnames.ora` file is set to the host name of your upgraded Oracle server.
7. Do the following steps on the active reporting server:
   a) Log on to the active reporting server as a user with administrative privileges.
   b) Stop the `IBMOpenPagesFrameworkModelGenerator` service.
   c) Go to the <CC_HOME>/`framework/conf` directory.
   d) Open the `framework.properties` file in a text editor. Ensure that the `oracle.client.path` property contains the location of the new Oracle Client `bin` directory.
   e) Save and close the file.
   f) Restart the `IBMOpenPagesFrameworkModelGenerator` service.
8. Start all OpenPages application servers (admin and non-admin), all reporting servers (active and standby), and the search server.

Upgrading Cognos

Upgrade to a supported version of Cognos Analytics.

About this task

If you are using Cognos 11.0.x or 11.1.x, you can upgrade Cognos in-place.

After you upgrade Cognos, copy the `bcprov-jdk15to18-1.68.jar` file that is provided with IBM OpenPages with Watson to the Java location that is used by the IBM Cognos server, and then register the `BouncyCastleProvider` in the JRE master security provider file, `java.security`.

Note the following recent name changes:

- `bcprov-jdk14-145.jar` is `bcprov-jdk15to18-1.68.jar` in 8.2.0.2 or later
- `org.bouncycastle145.jce.provider.BouncyCastleProvider` is `org.bouncycastle.jce.provider.BouncyCastleProvider` in 8.2.0.2 or later
- `CAMCryptoBC` is BC in 8.2.0.2 or later

Procedure

1. Log on to the reporting server as a user with administrative privileges.
2. Stop all Cognos services.
3. Upgrade Cognos Analytics. See Upgrading your current version of Cognos Analytics 11.
4. Locate the `bcprov-jdk15to18-1.68.jar` file.
   The file is available on each application server in the *<OP_HOME>*`/temp/jre/lib/ext/` directory.
5. If the Cognos software is using the JRE that is installed with Cognos, do the following steps:
   a) Copy the `bcprov-jdk15to18-1.68.jar` file to the *<COGNOS_HOME>*`/analytics/jre/lib/ext` directory.

Note: If you are using Cognos Analytics 11.1.5 or later, copy the file to *<COGNOS_HOME>*/analytics/ibm-jre/lib/ext.

b) Register the BouncyCastleProvider in the JRE master security provider file, if it is not already registered.

To register the provider, add the following line to the java.security file that is stored in the *<COGNOS_HOME>*/analytics/jre/lib/security directory.

```
security.provider.<#>=
    org.bouncycastle.jce.provider.BouncyCastleProvider
```

Note: If you are using Cognos Analytics 11.1.5 or later, the java.security file is in the *<COGNOS_HOME>*/analytics/ibm-jre/lib/security directory.

Where: The number sign, *<#>*, is one increment above the last number in the list. For example, security.provider.9.

6. If the Cognos software is using the JRE that is installed with IBM SDK, Java Technology Edition, do the following steps:

a) Copy the bcprov-jdk15to18-1.68.jar file to the *<JAVA_HOME>*/lib/ext directory.

b) Register the BouncyCastleProvider in the JRE master security provider file, if it is not already registered.

To register the provider, add the following line to the java.security file that is stored in the *<JAVA_HOME>*/lib/security directory.

```
security.provider.<#>=
    org.bouncycastle.jce.provider.BouncyCastleProvider
```

Where: The number sign, *<#>*, is one increment above the last number in the list. For example, security.provider.9.

7. If the Cognos software is using a JRE that is installed in another location on the reporting server, do the following steps:

Replace *<JAVA_LOCATION>* with the directory where the JRE is installed.

a) Copy the bcprov-jdk15to18-1.68.jar file to the *<JAVA_LOCATION>*/lib/ext directory.

b) Register the BouncyCastleProvider in the JRE master security provider file, if it is not already registered.

To register the provider, add the following line to the java.security file that is stored in the *<JAVA_LOCATION>*/lib/security directory.

```
security.provider.<#>=org.bouncycastle.jce.provider.BouncyCastleProvider
```

Where: The number sign, *<#>*, is one increment above the last number in the list. For example, security.provider.9.

8. Restart the reporting servers.
9. If you upgraded to 11.1.5 or later and you are using the JRE that is installed with Cognos, you need to update the Java location.

In Cognos Analytics 11.1.5 and later, the path is:

- On Windows: C:\IBM\cognos\analytics\ibm-jre\jre
- On Linux: /usr/IBM/cognos/analytics/ibm-jre/jre

For more information, see How to Change the Java Location on an OpenPages Reporting Server.

10. If you upgraded to 11.1.5 or later and you are using the JRE that is installed with Cognos, re-import the OpenPages SSL certificates into the Cognos JRE.

For more information, see the *IBM OpenPages with Watson Administrator's Guide*.

Getting a copy of the IBM SDK (Windows)

Before you install OpenPages, install IBM SDK, Java Technology Edition and set up the system environment variables for Java on each application server and the search server. You can also use the steps to install the IBM SDK on the installation server.

About this task

For application servers, the version of the IBM SDK must be the same on each of the servers.

Procedure

1. Locate the IBM SDK on the IBM OpenPages with Watson installation media.

   The path is `\OP_<version>_Main\IBM_Java\WIN64\java_8.0_64`
2. Copy the IBM SDK to the local hard disk of the server.

   You can copy the IBM SDK to any directory on the server.

   For example, copy the IBM SDK to the root of the C drive under `C:\IBM`.
3. Set the system environment variables for Java.
   a) In the Windows search box, type `environment variables`, and then click **Edit system environment variables**.
   b) On the **Advanced** tab, click **Environment variables**.
   c) In the **System Variables** pane, click **New**.
   d) Type `JAVA_HOME` in the **Variable name** field.
   e) Type `C:\IBM\java_8.0_64` in the **Variable value** field.
   f) Click **OK**.
   g) Under System variables, select the **Path** variable, and then click **Edit**.
   h) Type `%JAVA_HOME%\bin;` at the beginning of the list of paths in the **Variable value** field.
   i) Click **OK**.

   Note: Start a new command prompt to see the changes to the environment variables.
4. Verify the version of Java that is on the server.

   Run the `java -version` command. The result should be similar to the following sample:

```
java version "1.8.0_241"
Java(TM) SE Runtime Environment (build 8.0.6.5 - pwa6480sr6fp5-20200111_02(SR6 FP5))
IBM J9 VM (build 2.9, JRE 1.8.0 Windows Server 2016 amd64-64-Bit Compressed References
20200108_436782 (JIT enabled, AOT enabled)
OpenJ9 - 7d1059c
OMR - d059105
IBM - c8aee39)
JCL - 20200110_01 based on Oracle jdk8u241-b07
```

If the location of Java changes later, you can update it. For more information, see the following technotes:

- How to Change the Java Location on an OpenPages Application Server
- How to Change the Java Location on an OpenPages Global Search Server

Tip: You can also change the location of Java on the reporting server. See How to Change the Java Location on an OpenPages Reporting Server.

Getting a copy of the IBM SDK (Linux)

Before you install OpenPages, install IBM SDK, Java Technology Edition and set up the system environment variables for Java on each application server and the search server. You can also use the steps to install the IBM SDK on the installation server.

About this task

For application servers, the version of the IBM SDK must be the same on each of the servers.

Procedure

1. Locate the IBM SDK on the IBM OpenPages with Watson installation media.

   The path is `/OP_<version>_Main/IBM_Java/Linux64/java_8.0_64`.
2. Copy the IBM SDK to the local hard disk of the server.

   You can copy the IBM SDK to any directory on the server.

   For example, copy the IBM SDK to `/opt/IBM/`.
3. Grant read, write, and execute permissions on Java to the OpenPages installation user (`opuser`).

   Run the following command:

   ```
   chmod -R +x /opt/IBM/java_8.0_64
   ```
4. Set the system environment variables for Java.
   a) Based on the shell that you are using and the account under which the server will run, edit the `.profile` or `.bashrc` file.
   b) Ensure that `JAVA_HOME` is set to `/opt/IBM/java_8.0_64`.
   c) Ensure that `PATH` includes `$JAVA_HOME/bin` as the first item.

   Note: Start a new shell window to see the changes to the environment variables.
5. Verify the version of Java that is on the server.

   Run the `java -version` command. The result should be similar to the following sample:

   ```
   java version "1.8.0_241"
   Java(TM) SE Runtime Environment (build 8.0.6.5 - pxa6480sr6fp5-20200111_02(SR6 FP5))
   IBM J9 VM (build 2.9, JRE 1.8.0 Linux amd64-64-Bit Compressed References 20200108_436782
   (JIT enabled, AOT enabled)
   OpenJ9   - 7d1059c
   OMR      - d059105
   IBM      - c8aee39)
   JCL - 20200110_01 based on Oracle jdk8u241-b07
   ```

If the location of Java changes later, you can update it. For more information, see the following technotes:

- How to Change the Java Location on an OpenPages Application Server
- How to Change the Java Location on an OpenPages Global Search Server

Tip: You can also change the location of Java on the reporting server. See How to Change the Java Location on an OpenPages Reporting Server.

Verifying servers before you upgrade

Before you upgrade, verify the status of the servers in your deployment.

Procedure

1. Ensure that no users are logged in to the OpenPages application.

   Users must not log in until the upgrade is complete.

2. If you use single sign-on (SSO) and you configured it to require an SSO login to access the REST API URLs under `/grc/api/*`, disable SSO.
3. Ensure that no database scripts are running.

   Database scripts, other than the upgrade scripts, must not be run until the upgrade is complete.
4. Ensure that there are no long running OpenPages processes.

   Examples of long running processes include FastMap imports and global search indexing processes.
5. If you use global search, ensure that the search services are stopped.

   For more information, see the *IBM OpenPages with Watson Installation and Deployment Guide*.
6. Stop the deployment manager, all OpenPages application servers (admin and non-admin), and all reporting servers (active and standby).

Chapter 4. Upgrade OpenPages

Complete the following tasks to upgrade IBM OpenPages with Watson.

Note: This list applies to in-place upgrades only. If you are migrating to 8.2, see the *IBM OpenPages with Watson Installation and Deployment Guide*. For more information about upgrade paths, see Chapter 2, "Determine your upgrade path," on page 9.

- Complete the preparation tasks:
  - Download the installation kit from Passport Advantage
  - "Review new features and fixes" on page 11
  - "Backing up your environment" on page 11
  - "Upgrade prerequisite software " on page 14
  - "Verifying servers before you upgrade" on page 21
- Upgrade the OpenPages database manually. For more information, see "Upgrade the OpenPages database (Oracle)" on page 23.
- Prepare the installation server. For more information, see "Preparing the installation server" on page 29
- Upgrade IBM OpenPages with Watson. For more information, see "Upgrading OpenPages" on page 39.
- Do the postinstallation tasks for in-place upgrades. For more information, see "Postinstallation tasks for upgrades" on page 40.
- Optional: Remove IBM WebSphere Application Server. For more information, see the *IBM OpenPages with Watson Installation and Deployment Guide*.

Upgrade the OpenPages database (Oracle)

You upgrade the OpenPages database by running scripts. Use these topics if you are upgrading OpenPages in-place.

You must run all of the upgrade scripts in sequence to upgrade the database schema.

Two of the scripts require DBA privileges: a pre-upgrade script and a post-upgrade script. If you have DBA privileges, you can run all of the scripts. If you do not have DBA privileges, contact your database administrator.

A schema user can run the scripts that do not require DBA privileges.

Note for 7.4.x and 8.0.x customers: The database upgrade scripts modify and drop some database structures to free up space in the database. To gain the full benefit of these changes, the PROPERTYVALS table needs to be reorganized. You can do the table reorganization after you upgrade the database or after you complete the upgrade to version 8.2. For information about how to reorganize a table, see the Oracle documentation.

Pre-upgrade step – Requires DBA privileges
During this step, your database administrator runs a script to prepare the database for the upgrade.

You need SYSDBA privileges to run this script.

Validate the pre-upgrade step
During this step, you run a script to verify that the pre-upgrade script completed successfully and that the database schema is ready for the upgrade.

Upgrade step
During this step, you run a script to upgrade the schema. The script determines the current version of the database schema, and then runs the upgrade scripts that are needed to upgrade the schema.

Post upgrade step – Requires DBA privileges
During this step, your database administrator runs a script to complete the database upgrade and to set database tuning parameters.

You need SYSDBA privileges to run this script.

Validate the post-upgrade step
During this step, you run a script to validate the post-upgrade step.

Preparing for the database upgrade (Oracle)

Prepare for the upgrade of the database schema.

Procedure

1. Shut down all OpenPages components: application servers (admin and non-admin), reporting servers (active and standby), and the search server.

   For more information, see the *IBM OpenPages with Watson Installation and Deployment Guide*.
2. Ensure that the Oracle database server is running.
3. Log on to the Oracle database server computer as a user with administrative privileges.
4. Go to the `/OP_<version>_Main/OP_<version>_Configuration/Database/ORACLE/UPGRADE_SCRIPTS` directory.
5. Verify that you have write permission on the `sql-wrapper.sql` file.
6. Edit the `sql-wrapper.sql` file.

   Note: Change only the parameters that are described in this step.

   Table 3. Parameters in the `sql-wrapper.sql` file for Oracle databases

Property	Description
`opx_datafile_storage_dir`	Defines the physical locations of the datafiles that are associated with the tablespaces that are created. This should be set to a value that is appropriate for your environment
`opx_dflt_sid`	The TNS alias of the Oracle database for OpenPages.
`opx_db_owner`	The OpenPages database owner
`opx_oracle_dba_user`	The user name of a DBA user. If your database administrator is going to run the DBA scripts for you, then you can leave this value empty when you run the non-DBA scripts.
`opx_override_ver_check`	Use the default value, `N`, unless you are re-running the database upgrade scripts after a failure. If the database upgrade failed in the middle of the schema upgrade process, set this parameter to `Y`. When you re-run the upgrade script, the upgrade process resumes from the last successful schema upgrade step.

7. If you want to run a custom script during the upgrade process, see "Running custom scripts during the database upgrade (Oracle)" on page 25.
8. If your database administrator is going to run the scripts that require DBA privileges, prepare the files for your database administrator.
   a) Go to the `/OP_<version>_Main/OP_<version>_Configuration/Database/ORACLE/UPGRADE_SCRIPTS` directory.
   b) Open the `op-dba-upgrade-file-list.txt` file.
   c) Send your DBA the `sql-wrapper.sql` file that you updated along with the files listed in the `op-dba-upgrade-file-list.txt` file.

d) Send your DBA the instructions to run the DBA scripts.

- "Running the pre-upgrade DBA script (Oracle)" on page 25
- "Running the post-upgrade DBA script (Oracle)" on page 28

Running custom scripts during the database upgrade (Oracle)

If you want to run custom scripts during the database upgrade process, edit the `sql-wrapper.sql` file to specify the scripts to run.

About this task

You can use the `custom_data_upgrade_script` parameter to configure a custom script.

The script that you specify is run during the database upgrade step. The custom script is called by the `op-database-product-upgrade.sh/bat` script after the other upgrade steps, such as DDL changes, PL/SQL code changes, and database level data changes are complete.

Procedure

1. Open the `sql-wrapper.sql` file.
2. Edit the following parameters:

   ```
   define custom_data_upgrade_script=no-op.sql
   ```

   Replace `no-op.sql` with the script that you want to run.
3. Place your custom scripts in the same directory as the `sql-wrapper.sql` file.

Running the pre-upgrade DBA script (Oracle)

Ask your database administrator to run the pre-upgrade script. Or, if you have SYSDBA privileges, you can run the script.

Before you begin

- The Oracle database server is running. All other OpenPages servers are stopped.
- The `JAVA_HOME` system variable is defined.
- `apache-ant-1.8.1` has been deployed to `/OP_<version>_Main/OP_<version>_Configuration/Database/ORACLE/UPGRADE_SCRIPTS`
- The `ORACLE_HOME` system variable is defined.

About this task

Run the following script: `op-database-dba-upgrade.sh|.bat`. The script uses the parameters defined in the `sql-wrapper.sql` file.

Procedure

1. Log on to the Oracle database server computer as a database administrator (DBA).
2. Locate the scripts that are required.

   If you are a database administrator, get the scripts from your OpenPages team.

   Or, you can get the scripts from the `/OP_<version>_Main/OP_<version>_Configuration/Database/ORACLE/UPGRADE_SCRIPTS` directory.
3. Verify that you have execute permission on the scripts.
4. Open the `sql-wrapper.sql` file. Verify that the values are suitable for your environment.

a) For the opx_oracle_dba_user parameter, enter a user that has SYSDBA privileges, for example SYS.

b) If you customized the table space names, update the define opx_dflt_* parameters with the custom table space names.

c) If you want to run custom scripts during the upgrade, see "Running custom scripts during the database upgrade (Oracle)" on page 25.

5. Run the following command:

   - On Windows:

     ```
     op-database-dba-upgrade.bat pre "<sysdba_password>"
     ```

   - On Linux:

     ```
     ./op-database-dba-upgrade.sh pre '<sysdba_password>'
     ```

   Note: Quotation marks are required around a password only if the password contains special characters. See "Special characters in passwords" on page 6.

6. Verify that the return code is 0, indicating success.

   You can also check the log file, op-database-dba-pre-upgrade.log.

What to do next

Validate the pre-upgrade script.

Validating the pre-upgrade DBA step (Oracle)

Run the script to validate the pre-upgrade DBA steps.

Before you begin

- The Oracle database server is running. All other OpenPages servers are stopped.
- The JAVA_HOME system variable is defined.
- apache-ant-1.8.1 has been deployed to /OP_*<version>*_Main/OP_*<version>*_Configuration/Database/ORACLE/UPGRADE_SCRIPTS
- The ORACLE_HOME system variable is defined.

Procedure

1. Log on to the Oracle database server computer as the OpenPages application user, opuser.
2. Go to the /OP_*<version>*_Main/OP_*<version>*_Configuration/Database/ORACLE/UPGRADE_SCRIPTS directory.
3. Verify that you have execute permission on the scripts.
4. Open the sql-wrapper.sql file. Verify that the values are suitable for your environment.
5. Run the following command:

   - On Windows:

     ```
     op-database-product-upgrade.bat preupgrade "<op_schema_owner_password>" ""
     ```

     The second parameter is not used, but must be included in the command. Use "".

   - On Linux:

     ```
     ./op-database-product-upgrade.sh preupgrade '<op_schema_owner_password>'
     ```

Note: Quotation marks are required around a password only if the password contains special characters.

6. Verify that the script completed successfully.

   Look for the following message: `Status:Success` or a return code of 0.

   You can also check the log file, `op-validate-dba-pre-upgrade.log`.

What to do next

Run the script to upgrade the database schema.

Upgrading the schema (Oracle)

Run the script to upgrade the database schema.

Before you begin

- The Oracle database server is running. All other OpenPages servers are stopped.
- The `JAVA_HOME` system variable is defined.
- `apache-ant-1.8.1` is deployed to `/OP_<version>_Main/OP_<version>_Configuration/Database/ORACLE/UPGRADE_SCRIPTS`
- The `ORACLE_HOME` system variable is defined.
- The `op-database-product-upgrade.sh|.bat preupgrade` script completed successfully.

Procedure

1. Log on to the Oracle database server computer as the OpenPages application user, `opuser`.
2. Go to the `/OP_<version>_Main/OP_<version>_Configuration/Database/ORACLE/UPGRADE_SCRIPTS` directory.
3. Verify that you have execute permission on the scripts in the `UPGRADE_SCRIPTS` directory and its subdirectories.
4. Open the `sql-wrapper.sql` file. Verify that the values are suitable for your environment.
5. Run the following command:

   The second parameter is not used, but must be provided. Use a dummy value, such as `xxx`.

   - On Windows:

     ```
     op-database-product-upgrade.bat upgrade "<op_schema_owner_password>" xxx
     ```

   - On Linux:

     ```
     ./op-database-product-upgrade.sh upgrade '<op_schema_owner_password>' xxx
     ```

   Note: Quotation marks are required around a password only if the password contains special characters. See “Special characters in passwords” on page 6.

6. Verify that the return code is 0, indicating success.

   You can also check the log file, `op-database-product-upgrade.log`.

What to do next

Ask your database administrator to run the post-upgrade DBA script.

Running the post-upgrade DBA script (Oracle)

Ask your database administrator to run the post-upgrade script. Or, if you have SYSDBA privileges, you can run the script.

Before you begin

- The Oracle database server is running. All other OpenPages servers are stopped.
- The JAVA_HOME system variable is defined.
- apache-ant-1.8.1 is deployed to /OP_*<version>*_Main/OP_*<version>*_Configuration/Database/ORACLE/UPGRADE_SCRIPTS
- The ORACLE_HOME system variable is defined.
- The op-database-product-upgrade.sh|.bat upgrade script completed successfully.

About this task

Run the following script: op-database-dba-upgrade.sh|.bat. The script uses the parameters defined in the sql-wrapper.sql file.

Procedure

1. Log on to the Oracle database server computer as a database administrator (DBA).
2. Locate the scripts that are required.

   If you are a database administrator, get the scripts from your OpenPages team.

   Or, you can get the scripts from the /OP_*<version>*_Main/OP_*<version>*_Configuration/Database/ORACLE/UPGRADE_SCRIPTS directory.
3. Verify that you have execute permission on the scripts.
4. Open the sql-wrapper.sql file. Verify that the values are suitable for your environment. In the opx_oracle_dba_user parameter, enter a user that has SYSDBA privileges, for example SYS.
5. Run the following command:
   - On Windows:

     ```
     op-database-dba-upgrade.bat post "<sysdba_password>"
     ```

   - On Linux:

     ```
     ./op-database-dba-upgrade.sh post '<sysdba_password>'
     ```

   Note: Quotation marks are required around a password only if the password contains special characters. See “Special characters in passwords” on page 6.
6. Verify that the return code is 0, indicating success.

   You can also check the log file: op-database-dba-post-upgrade.log.

What to do next

Validate the post-upgrade DBA step.

Validating the post-upgrade DBA step (Oracle)

Run the script to validate the post-upgrade DBA steps.

Before you begin

- The Oracle database server is running. All other OpenPages servers are stopped.

- The JAVA_HOME system variable is defined.
- apache-ant-1.8.1 has been deployed to /OP_*<version>*_Main/OP_*<version>*_Configuration/Database/ORACLE/UPGRADE_SCRIPTS
- The ORACLE_HOME system variable is defined.

Procedure

1. Log on to the Oracle database server computer as the OpenPages application user, opuser.
2. Go to the /OP_*<version>*_Main/OP_*<version>*_Configuration/Database/ORACLE/UPGRADE_SCRIPTS directory.
3. Verify that you have execute permission on the scripts.
4. Open the sql-wrapper.sql file. Verify that the values are suitable for your environment.
5. Run the following command:
   - On Windows:

     ```
     op-database-product-upgrade.bat postdba "<op_schema_owner_password>" ""
     ```

     The second parameter is not used, but must be included in the command. Use "".
   - On Linux:

     ```
     ./op-database-product-upgrade.sh postdba '<op_schema_owner_password>'
     ```

   Note: Quotation marks are required around a password only if the password contains special characters. See “Special characters in passwords” on page 6.
6. Verify that the script completed successfully.

   Look for the following message: Status:Success or a return code of 0.

   You can also check the log file, op-validate-dba-post-upgrade.log.
7. Remove the passwords from the sql-wrapper.sql file for security purposes.

Results

The OpenPages database schema is upgraded.

Preparing the installation server

Install the 8.2 installation server and migrate your deployments and users to the new installation server.

Note: The installation server cannot be upgraded in-place. You need to install the 8.2 installation server into a new directory.

You have two options:

- You can install the installation server and migrate your deployments and users during the installation process.
- You can install the installation server and migrate your deployments and users later.

If you manage the installation agents manually, you also need to install the 8.2 agent software on each remote server. If you want to keep the existing agent software on a remote server, for backup purposes for example, do the following steps:

1. Stop the pre-8.2 agent if it is running.
2. Install the 8.2 agent software in a new directory.
3. In the 8.2 installation app, update the **Agent Directory** field on the server card to point to the new directory.
4. Click **Validate**.

Note: You can install different versions of the installation server on the same host. If you do so, you must use a different port number and directory for each installation server. You can specify the port number of the 8.2 server during the setup process.

Setting up the installation server on Windows

You can set up the installation server on a server in your deployment or on a separate computer. Use a computer that can communicate with the servers in your OpenPages environment.

After you set up the installation server, you can use the OpenPages installation app to create and manage deployments.

Note: If you already set up the installation server and you want to update it with a fix pack, see "Update the installation server and agents" on page 35.

Before you begin

The computer where you set up the installation server must meet the following requirements:

- IBM SDK, Java Technology Edition or Java Runtime Environment (JRE) is installed.
- Java is included in the PATH system environment variable.

You might also want a PDF reader application on the computer. When you install or upgrade OpenPages, you can download validation reports in PDF format.

Procedure

1. Download the OpenPages 8.2 package from Passport Advantage.
2. Log on to the computer as an administrator.
3. If an earlier version of the installation server is running, stop it.
4. Do one of the following steps:
   - Update the antivirus policy on the installation server computer to allow `Node.js`.
   - Disable antivirus software on the installation server computer. You can re-enable it after you install the installation server.
5. Create a new directory.

   If you have more than one version of the installation server on the same host, use a separate directory for each version.

   For example, `C:\IBM\OPInstall<version>`.
6. Locate the installation files.

   The files are stored in `\OP_<version>_Main\OP_<version>_Installer`.
7. Copy the contents of the `\OP_<version>_installer` directory to the directory that you created.
8. Change directory to `<installation_server_home>\OP_<version>_installer\install\Windows`.
9. Open a command prompt as an administrator.
10. Run the installation script.

    You can use the following optional arguments:

    - `/p:<password>` – Sets the password for the initial installation app user, called `admin`. If you exclude the argument, the `install.bat` script prompts you for the password.
    - `/n:<port>` – Sets the port that the installation server runs on when you start it. If you have multiple installation servers that run on the same hardware, ensure that each installation server uses a different port number. Specify an integer in the range 0 - 65535. If you exclude this argument, the default port number (8443) is used.
    - `/m:<old_directory>` – Migrates existing deployments and installation server user accounts to the 8.2 installation server. Use this argument if you have 7.4, 8.0.x, or 8.1.x deployments that you

want to use with the new installation server. For *<old_directory>*, enter the full path to the 7.4, 8.0.x, or 8.1.x installation server home directory. Alternatively, you can migrate deployments and users after you install the 8.2 installation server. For more information, see "Migrating deployments and installation server users" on page 37.

- `/s` – Prevents the installation server from starting after the `install.bat` completes. If you exclude this argument, the installation server starts automatically after the `install.bat` script completes.

Syntax:

```
install.bat –acceptLicense [/p:password] [/m:<old_directory>] [/n:<port>] [/s]
```

11. If you did not use the `/p` parameter, type a password and then press Enter.
12. After the installation completes, re-enable the antivirus software on the installation server.
    Do this step if you disabled the antivirus software in step "4" on page 30.
13. Update the installation server to the latest fix pack version.
    See "Update the installation server and agents" on page 35.

Results

The OpenPages installation server is installed.

If you used the `/s` argument, start the installation server. For more information, see the *IBM OpenPages with Watson Installation and Deployment Guide*.

You can now log in. For the user name, type `admin`. For the password, type the password that you set when you ran the `install.bat` script. For more information, see the *IBM OpenPages with Watson Installation and Deployment Guide*.

Note: If you used the `/m` argument but some deployments or users were not migrated, do not run the `install.bat` script again. Instead, repeat the migration without reinstalling. See "Migrating deployments and installation server users" on page 37.

Setting up the installation server on Linux

You can set up the installation server on a server in your deployment or on a separate computer. Use a computer that can communicate with the servers in your OpenPages environment.

After you set up the installation server, you can use the OpenPages installation app to create and manage deployments.

Important: If you use Windows servers in your deployment, set up the OpenPages installation server on a Windows computer. See "Setting up the installation server on Windows" on page 30.

Note: If you already set up the installation server and you want to update it with a fix pack, see "Update the installation server and agents" on page 35.

Before you begin

The computer where you set up the installation server must meet the following requirements:

- IBM SDK, Java Technology Edition or Java Runtime Environment (JRE) is installed.
- Java is included in the PATH system environment variable.
- `JAVA_HOME` is set.

You might also want a PDF reader application on the computer. When you install or upgrade OpenPages, you can download validation reports in PDF format.

About this task

This video demonstrates how to set up the installation server. The steps are similar for 8.2: https://youtu.be/OiyuKjYyPrg.

Procedure

1. Log on to the computer as an administrator.
2. If an earlier version of the installation server is running, stop it.
3. Do one of the following steps:
   - Update the antivirus policy on the installation server computer to allow `Node.js`.
   - Disable antivirus software on the installation server computer. You can re-enable it after you install the installation server.
4. Create a directory.

   If you have more than one version of the installation server on the same host, use a separate directory for each version.

   For example, `/home/opuser/IBM/OPInstall<version>`.
5. Locate the installation files.

   The files are stored in `/OP_<version>_Main/OP_<version>_Installer`.
6. Copy the contents of the `OP_<version>_Installer` directory to the directory that you created.
7. Change directory to `/home/opuser/IBM/OPInstall/OP_<version>_Installer/install/Linux`.
8. Grant the `+rwx` permission to the user on the installation server directory, subdirectories, and scripts.
9. Open a shell and run the setup script.

   You can use the following optional arguments:

   - `-p <password>` – Sets the password for the initial installation app user, called `admin`. If you exclude the argument, the `install.bat` script prompts you for the password.
   - `-n <port>` – Sets the port that the installation server runs on when you start it. If you have multiple installation servers that run on the same hardware, ensure that each installation server uses a different port number. Specify an integer in the range 0 - 65535. If you exclude this argument, the default port number (8443) is used.
   - `-m <old_directory>` – Migrates existing deployments and installation server user accounts to the 8.2 installation server. Use this argument if you have 7.4, 8.0.x, or 8.1.x deployments that you want to use with the new installation server. For *<old_directory>*, enter the full path to the 7.4, 8.0.x or 8.1.x installation server home directory. Alternatively, you can migrate deployments and users after you install the 8.2 installation server. For more information, see “Migrating deployments and installation server users” on page 37.
   - `-s` – Prevents the installation server from starting after the `install.sh` script completes. If you exclude this argument, the installation server starts automatically after the `install.sh` script completes.

   Syntax:

   ```
   ./install.sh –-acceptLicense [-p password] [-m <old_directory>] [-n <port>] [-s]
   ```
10. If you did not use the `-p` parameter, type a password and then press Enter.
11. Close the shell window.
12. After the installation completes, re-enable the antivirus software on the installation server.

    Do this step if you disabled the antivirus software in step “3” on page 32.
13. Update the installation server to the latest fix pack version.

    See “Update the installation server and agents” on page 35.

Results

The OpenPages installation server is installed.

If you used the `-s` argument, start the installation server. For more information, see the *IBM OpenPages with Watson Installation and Deployment Guide*.

You can now log in. For the user name, type `admin`. For the password, type the password that you set when you ran the `install.sh` script. For more information, see the *IBM OpenPages with Watson Installation and Deployment Guide*.

Note: If you used the `-m` argument but some deployments or users were not migrated, do not run the `install.sh` script again. Instead, repeat the migration without reinstalling. See "Migrating deployments and installation server users" on page 37.

Installing agents manually

The installation server can automatically install the agent software on remote servers. But you can install the agent software manually, if you prefer.

Before you begin

The computer where you install the agent software must meet the following requirements:

- IBM SDK, Java Technology Edition or Java Runtime Environment (JRE) is installed.
- Java is included in the PATH system environment variable.

About this task

When you specify the deployment properties for a remote server, you are asked to provide the user name and password of an administrator account on the remote server. The installation server uses these credentials to install the agent software on the remote server. However, your organization might have policies that restrict the use of administrator credentials. In this case, you can install the agent software manually before you install IBM OpenPages with Watson.

The overall process involves the following steps:

1. Install the agent software manually and start the agent on each remote server, except the database server. The agent software is not needed on the database server.
2. In the installation app, enter the deployment properties for the remote servers.
   - Enable the **Remote Deploy** option.
   - Leave the **Local User Name** and **Local User Password** fields empty.
   - In the **Agent Directory** field, type the full path to the directory on the remote server where you installed the agent software. This directory is the *<agent_home>* directory.

Procedure

1. Log on to the remote server as an administrator.
2. Do one of the following steps:
   - Update the antivirus policy on the remote server to allow `Node.js`.
   - Disable antivirus software on the remote server. You can re-enable it after you install the agent software.
3. Create a directory.
   For example:
   - Windows: `C:\IBM\OPAgent`
   - Linux: `/home/opuser/IBM/OPAgent`

   This directory will be the *<agent_home>* directory for the remote server.

4. Copy the agent installation software to the remote server.
   a) Locate the following file on the installation server: *<installation_server_home>*/op-installer-agent.zip.
   b) Copy op-installer-agent.zip to the *<agent_home>* directory that you created on the remote server.
   c) Extract the op-installer-agent.zip file into the *<agent_home>* directory.
5. Open a shell or command window. If you are using Windows, open the command window as an administrator.
6. Go to the *<agent_home>*/install/*<OS>* directory.
7. Run the following script to install the agent software:
   - Windows:

     ```
     install.bat -acceptLicense [/n <port>] [/s]
     ```

     You can use the following optional arguments:
     - /n:*<port>* – Sets the port that the installation agent runs on when you start it. Specify an integer in the range 0 - 65535. If you exclude this argument, the default port number (8443) is used.
     - /s – Prevents the installation agent from starting after the install.bat script completes. If you exclude this argument, the installation agent starts automatically after the install.bat script completes.
   - Linux:

     ```
     chmod 755 install.sh
     ./install.sh --acceptLicense [-n <port>] [-s]
     ```

     You can use the following optional arguments:
     - -n *<port>* – Sets the port that the installation agent runs on when you start it. Specify an integer in the range 0 - 65535. If you exclude this argument, the default port number (8443) is used.
     - -s – Prevents the installation agent from starting after the install.sh script completes. If you exclude this argument, the installation agent starts automatically after the install.sh script completes.
8. When the script completes, close the shell or command window.
9. Start the agent.
   See "Starting the installation agent manually" on page 38.
10. Update the agent software to the latest fix pack version.
    For more information, see "Update the installation server and agents" on page 35.
11. Repeat these steps on each remote server, except the database server.

What to do next

When you enter the server properties in the installation app or in the deploy.properties file, do the following steps:

- Enable the **Remote Deploy** option.
- In the **Agent Directory** field, type the full path to the *<agent_home>* directory on the remote server.
- Leave the **Local User Name** and **Local User Password** fields empty.

Ensure that the agents are started before you do any installation tasks. See "Starting the installation agent manually" on page 38.

Update the installation server and agents

Update the installation server to use the latest 8.2.x version.

The latest version of the installation server is provided in the fix pack installation kit.

Do the following tasks:

- Update the installation server.
- If you installed the agent software manually on remote servers, update the agent software on each remote server.

Updating the installation server

Before you install a new version of IBM OpenPages with Watson (a release, fix pack, or interim fix), update the OpenPages installation server to the latest 8.2.x fix pack version.

About this task

This video demonstrates how to update the installation server: https://youtu.be/FghmmHO5Ug8.

Procedure

1. Download the latest OpenPages fix pack from Fix Central.
2. Log on to the OpenPages installation server computer as the user who installed the installation server.

   Alternatively, you can log in as any user who has full permissions on the installation server directories and who can run `Node.js`.
3. Locate the `openpages_installer_<version>.zip` file in the fix pack kit.

   The file is stored in `/OP_<version>_Main/OP_<version>_Installer`.
4. Copy the file to the *<Installation_server_home>*`/src/assets/maintenance` directory on the installation server.
5. Stop the installation server if it is running.
6. Update the installation server.
   a) Open a command prompt as an administrator or open a shell window.
   b) Go to the *<Installation_server_home>* directory and run the following command:

   ```
   npm run upgrade
   ```
7. Start the installation server.
8. Verify the update. Log in to the installation app, open any deployment, and click **About** to see the version number.
9. If you installed the agent software manually on the remote servers in your deployment, update the agent software on each remote server.

   For more information, see “Updating agents manually” on page 36.

   Note: Do not click **Validate** until you have updated the agent software on each remote server.

   If the installation server installed the agent software on your remote servers, you do not need to update the agents manually. The installation server updates the agents automatically when you click **Validate**.

Updating agents manually

Use this procedure to update the agent software manually to an 8.2.0.x fix pack version.

About this task

The installation server can automatically update the agent software on remote servers. But you can update the agent software manually, if you prefer.

When you specify the deployment properties for a remote server, you are asked to provide the user name and password of an administrator account on the remote server. The installation server uses these credentials to update the agent software on the remote server. However, your organization might have policies that restrict the use of administrator credentials. In this case, you can update the agent software manually before you install IBM OpenPages with Watson or apply a fix pack.

The overall process involves the following steps:

1. Update the installation server. See "Updating the installation server" on page 35.
2. Update the agent software manually and start the agent on each remote server, except the database server. The agent software is not needed on the database server.
3. In the installation app, enter the deployment properties for the remote servers.
   - Enable the **Remote Deploy** option.
   - You can leave the **Local User Name** and **Local User Password** fields empty.
   - In the **Agent Directory** field, type the full path to the directory on the remote server where the agent software is installed. This directory is the *<agent_home>* directory.
4. Validate your deployment and continue with the installation of OpenPages or the fix pack.

Procedure

1. Log on to the remote server as the user who installed the agent software.

   Alternatively, you can log in as any user who has full permissions on the agent directories and who can run `Node.js`.
2. Stop the agent.

   For more information, see "Stopping the installation agent manually" on page 39.
3. Copy the installation file to the remote server.
   a) Locate the following file in the 8.2.0.x fix pack kit: `openpages_installer_<version>.zip`

      The file is stored in `/OP_<version>_Main/OP_<version>_Installer`.
   b) Copy `openpages_installer_<version>.zip` to the *<agent_home>*`/src/assets/maintenance` directory on the installation server.

      Do not extract the file.
4. Update the agent software.

   On Windows:

   a) Verify that no command prompts or applications, such as Windows Explorer, are accessing the *<agent_home>* directory or its subdirectories.
   b) Open a command prompt as an administrator.
   c) Go to the *<agent_home>* directory
   d) Run the following command.

   ```
   npm run upgrade
   ```

   On Linux:

   a) Open a shell and go to the *<agent_home>* directory.

b) Run the following command.

```
npm run upgrade
```

When the process completes, the following message is displayed:

```
Installer upgrade is successful...
```

5. Start the agent.

   See “Starting the installation agent manually” on page 38.

6. Repeat these steps on each remote server, except the database server.

What to do next

When you fill in the server properties, do the following:

- Enable the **Remote Deploy** option.
- In the **Agent Directory** field, type the full path to the *<agent_home>* directory on the remote server.
- You can leave the **Local User Name** and **Local User Password** fields empty.

  Note: If you leave the **Local User Name** and **Local User Password** fields empty, you must start the agents manually. See “Starting the installation agent manually” on page 38.

Migrating deployments and installation server users

You can migrate deployments and user accounts from a 7.4.x, 8.0.x, or 8.1.x installation server to the 8.2 IBM OpenPages with Watson installation server.

About this task

Do this task if the following conditions are met:

- You have a 7.4.x, 8.0.x, or 8.1.x installation server (the source installation server).
- You set up the 8.2 installation server (the target installation server).
- When you set up the 8.2 installation server, you did not migrate deployments and user accounts from the source installation server by using the `/m` (Microsoft Windows) or `-m` (Linux) argument.

  Or, you migrated, but some deployments or user accounts did not get migrated.

When you migrate, keep the following points in mind:

- Deployments and users that already exist in the 8.2 installation server are not migrated.
- User accounts in the source installation server that are missing passwords are not migrated.
- A deployment is not migrated if any validation, installation, or configuration processes are running.

Procedure

1. If the installation app is open, log out and close the browser window.
2. Stop the 7.4.x, 8.0.x, or 8.1.x installation server.

   Windows

   - Stop the `ibmopenpagesgrcplatforminstaller<version>.exe` (7.4 or 8.0) or `ibmopenpageswithwatsoninstaller<version>.exe` (8.1) service.
   - Or, go to the `<installation_server_home>` directory of the installation server that you want to stop. Open a command prompt as an administrator, and then run the following command:

     ```
     npm run stop
     ```

Linux

a. Open a shell and go to the <installation_server_home> directory of the installation server that you want to stop, for example /home/opuser/IBM/OPInstall/OP_<*version*>_Installer.

b. Run the following command:

```
npm run stop
```

3. Log on to the computer where you set up the 8.2 installation server.
4. Migrate deployments and user accounts to the 8.2 installation server.

a) Open a shell or command window and go to the <installation_server_home> directory, for example /home/opuser/IBM/OPInstall/OP_<*version*>_Installer.

b) Run the following command:

Replace *<old_directory>* with the full path to the 7.4.x, 8.0.x, or 8.1.x installation server home directory.

```
npm run migration <old_directory>
```

If a deployment or user account is not migrated, fix any issues, and then run the migration again.

Starting the installation agent manually

You can start the agent on a remote server manually.

About this task

When you specify the deployment properties for a remote server, you are asked to provide the user name and password of an administrator account on the remote server. The installation server uses these credentials to start and stop the agent software on the remote server. If you do not specify login credentials in the deployment properties and you install the agent software manually, you need to start and stop the agent manually. You cannot use the installation app to start or stop the agent.

You might also choose to start and stop agents manually if you prefer to use the command line.

Procedure

1. Log on to the remote server as the user who installed the agent software.

   Alternatively, you can log in as any user who has full permissions on the agent directories and who can run Node.js.
2. Start the installation agent.

Windows

Do one of the following steps:

- Start the ibmopenpageswithwatsoninstaller<*version*>.exe service.
- Go to the *<agent_home>*\install\Windows directory. Right-click the startup.bat file and click **Run As Administrator**.

Linux

a. Open a shell and go to the *<agent_home>*/install/Linux directory.

b. Run the following command:

```
./startup.sh
```

Results

The installation agent is running.

Stopping the installation agent manually

You can stop the agent on a remote server manually.

About this task

When you specify the deployment properties for a remote server, you are asked to provide the user name and password of an administrator account on the remote server. The installation server uses these credentials to start and stop the agent software on the remote server. If you do not specify login credentials in the deployment properties and you install the agent software manually, you need to start and stop the agent manually. You cannot use the installation app to start or stop the agent.

You might also choose to start and stop agents manually if you prefer to use the command line.

Procedure

1. Log on to the remote server as the user who installed the agent software.

   Alternatively, you can log in as any user who has full permissions on the agent directories and who can run `Node.js`.
2. Stop the installation agent.
   - Windows: Stop the `ibmopenpageswithwatsoninstaller<version>.exe` service. Or, open a command prompt as an administrator, go to the *<agent_home>* directory, and run the following command:

     ```
     npm run stop
     ```
   - Linux: Go to the *<agent_home>* directory and run the following command:

     ```
     npm run stop
     ```

Results

The installation agent is stopped.

Upgrading OpenPages

Upgrade to get the latest fixes and features.

Before you begin

- Complete the following preparation tasks:
  - Download the installation kit from Passport Advantage®
  - “Review new features and fixes” on page 11
  - “Backing up your environment” on page 11
  - “Verifying servers before you upgrade” on page 21
- If you installed the agents on remote servers manually, ensure that the agents are running.
- Upgrade the OpenPages database. For more information, see “Upgrade the OpenPages database (Oracle)” on page 23.
- Ensure that IBM Cognos Configuration is not running.
- Ensure that the deployment manager, all OpenPages application servers (admin and non-admin), and the search server are stopped.
- Ensure that all reporting servers (active and standby) are running.

About this task

Use the installation server to upgrade OpenPages.

Procedure

1. Log in to the OpenPages installation app.
   For more information, see the *IBM OpenPages with Watson Installation and Deployment Guide*.
2. Open the deployment that you want to upgrade.
   If your deployment is already open, refresh the page.
3. Review the settings on each server card.
4. Click the **Deployment Task** list and select **Upgrade**, and then select the version that you want to install.

   If **Upgrade** is not displayed in the **Deployment Task** list, click **Validate**.

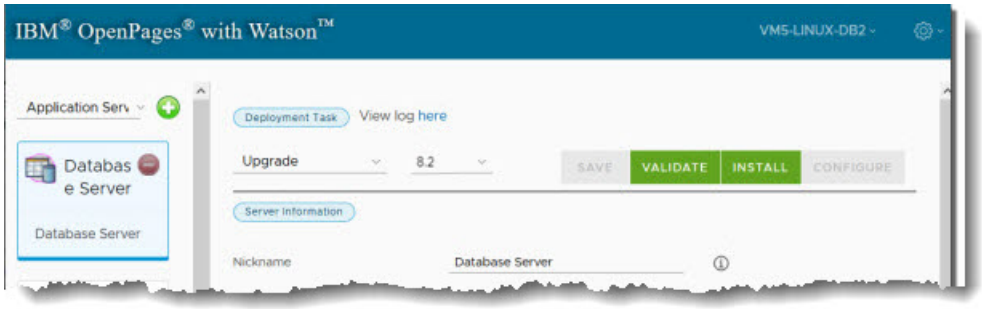

Figure 1. Selecting the upgrade task

5. On each application server card, verify that **Java Home Directory** is set to the IBM SDK, Java Technology Edition that you installed on the application server.

   If you previous used the Java that is installed with IBM WebSphere Application Server Network Deployment, you might need to update the **Java Home Directory** field.
6. Click **Validate**.
7. Click **Install**.

   Tip: You can log out and close the browser window. The **Install** process continues to run. When you log in to the installation app again, the app shows the status of your deployment. You can also close the browser window during the **Configure** process.
8. Click **Configure**.

What to do next

Do the post-installation tasks. For more information, see “Postinstallation tasks for upgrades” on page 40.

Postinstallation tasks for upgrades

After you upgrade IBM OpenPages with Watson, you must complete some additional tasks.

Updating optional apps

If you use optional apps, such as IBM OpenPages Loss Event Entry or IBM OpenPages connectors, you need to do some additional upgrade tasks,

Use the following table to determine what you need to do.

Table 4. Upgrade tasks for optional apps	
App or component	**Upgrade tasks**
IBM OpenPages Loss Event Entry	Upgrade Loss Event Entry. For more information, see the *IBM OpenPages with Watson Installation and Deployment Guide*.
IBM OpenPages connectors • IBM OpenPages SDI Connector for UCF Common Controls Hub • IBM QRadar® connector	If you are upgrading from 8.1 or earlier, install IBM Security Directory Integrator 7.2.0.3 and update the configuration. See the *IBM OpenPages with Watson Installation and Deployment Guide*
Approval app	If you deployed the approval app in version 7.2.0.1 or later, you need to upgrade the approval app. For more information, see the *IBM OpenPages with Watson Installation and Deployment Guide* .

Note: If you used IBM Business Process Manager in a prior release, remove the integration. For more information, see Removing the IBM BPM integration from OpenPages with Watson.

Restore custom application server settings

After you upgrade, restore application server configuration settings and customizations.

If you made changes to the following files in your pre-upgrade environment, re-implement your changes after the upgrade.

- `<OP_HOME>/aurora/conf/aurora.properties`
- `<OP_HOME>/aurora/conf/Server<#>-server.properties`
- `<OP_HOME>/aurora/conf/Server<#>-sosa.properties`

If you modified the `web.xml` file, `application.xml` file, or if you customized settings in the IBM WebSphere Integrated Solutions Console in your pre-8.2 environment, re-implement the changes in WebSphere Liberty.

For example, see the following topics in the *IBM OpenPages with Watson Administrator's Guide*:

- Shortening the URL for OpenPages with Watson
- Enabling secure session cookies in WebSphere Liberty
- Configuring extended access logging on WebSphere Liberty

If single sign-on (SSO) was configured in the source system, you need to update the SSO configuration in your target environment. For more information, see the *IBM OpenPages with Watson Installation and Deployment Guide*.

Restoring solutions helpers, images, and other files

Restore custom solutions helpers, images, and other custom deliverables that you backed up.

About this task

If you backed up the following items, restore them:

- Solutions schema
- Custom deliverables from the IBM OpenPages Technical Services Team
- Custom code

Review your custom code. The location of the `dojo` and `idx` files has changed. The new paths are:

- *<OP_HOME>*`/wlp-usr/shared/apps/op-apps.ear/sosa.war/dojo`
- *<OP_HOME>*`/wlp-usr/shared/apps/op-apps.ear/sosa.war/idx`

Configure new features

Some new features need to be configured.

GRC Calculations

Version 8.2 includes sample calculations. When you upgrade, the installation process loads the sample calculations but does not enable them. Depending on your environment, your schema might not contain all of the object types, object associations, and fields that are used in the sample calculations. To view the sample calculations, click  > **Solution Configuration** > **Calculations**.

For more information, see "Setting up calculations" in the *IBM OpenPages with Watson Administrator's Guide*.

GRC Workflow

Version 8.2 includes sample workflows. When you upgrade or migrate from a release prior to 8.0.0.2, the installation process loads the sample workflows but does not enable them. Depending on your environment, your schema might not contain all of the object types, object associations, and fields that are used in the sample workflows. To view the sample workflows, click  > **Solution Configuration** > **Workflows**.

If you are upgrading or migrating from 8.0.0.2 or later, the installation process does not load the sample workflows. You can load them manually. For more information, see “Loading the sample workflows” on page 49.

Application permissions for new features in the Task Focused UI

Review the following application permissions. Add them to your role templates to give users access to the features and functionality.

Table 5. Application permissions for Task Focused UI features

Feature	Application permissions
GRC Calculations	To give users the ability to create and manage calculations, add the **SOX** > **Administration** > **Calculation** permission to role templates.
Scheduler	To give users the ability to create and manage scheduled jobs, add the **SOX** > **Administration** > **Scheduler** permission to role templates.
Watson Assistant	To give administrators the ability to configure assistants, add the **SOX** > **Administration** > **Watson Assistant** permission to role templates. To give users access to the user interface that enables them to interact with assistants in OpenPages, add the **SOX** > **User Interfaces** > **Watson Assistant UI** permission to role templates.
LDAP server configuration for user provisioning	To give administrators the ability to configure the LDAP server for user provisioning via the Task Focused UI, add the **SOX** > **Administration** > **LDAP Server** permission to role templates.
Activity tab in views	To give users access to the **Activity** tab in views, add the **Audit Trail** permission to role templates.
Watson Natural Language Classifiers (also called cognitive services)	To give administrators the ability to configure cognitive services in the Task Focused UI, add the **SOX** > **Administration** > **Cognitive** permission to role templates.
Dashboard administration	To give administrators the ability to create and manage dashboards, add the **SOX** > **Administration** > **Dashboards** permission to role templates.

Table 5. Application permissions for Task Focused UI features (continued)	
Feature	**Application permissions**
View Designer, **Display debug info** menu item	To give users access to the View Designer, add the **SOX** > **Administration** > **Task Focused UI** permission to role templates. This permission also controls whether the > **Other** > **Display debug info** menu item is displayed.
Encryption keystore	To give administrators the ability to configure and manage the encryption keystore in the Task Focused UI, add the **SOX** > **Administration** > **Encryption Keystore** permission to role templates.

Application permissions for tools and utilities

The application permissions that are required for the following tools have changed. Update your role templates to include the permissions.

Table 6. Application permissions for tools and utilities	
Tool or utility	**Application permissions**
ObjectManager	To give users the ability to do `load`, `validate`, and `batch` operations with ObjectManager, assign the following permissions: • **API** > **Administration** > **Background Process** > **Get Process Info** • **SOX** > **Administration** > **ImportConfiguration** To give users the ability to do `dump` operations with ObjectManager, assign the following permissions: • **API** > **Administration** > **Background Process** > **Get Process Info** • **SOX** > **Administration** > **ExportConfiguration**
Notification Manager	To give users the ability to create notifications with the Notification Manager tool, add the **SOX** > **Administration** > **Notification Manager** permission to role templates.

Solutions postinstallation tasks

After you upgrade OpenPages, you might need to do some postinstallation tasks to update OpenPages solutions.

Note:

Version 8.2 introduces significant enhancements to solutions. The updates are available in fresh installations only. If you want to update these solutions, contact IBM OpenPages Support.

- A new solution, IBM OpenPages Business Continuity Management is now available.
- IBM OpenPages Regulatory Compliance Management has been updated.

If you upgraded from 8.1.0.x

- If you use IBM OpenPages Model Risk Governance and you want to use IBM Watson® OpenScale, load the required fields and field groups. See “Updating MRG” on page 48.
- If you use IBM OpenPages Third Party Risk Management (prevously called OpenPages Third Party Risk Management), load the new dashboard. See “Updating TPRM” on page 48.
- If you want to use the sample workflows that are provided with OpenPages, see “Loading the sample workflows” on page 49.

If you upgraded from 8.0.0.2 or a later 8.0.0.x release

- If you use IBM OpenPages Internal Audit Management, load the timesheet helpers.

  If you loaded the timesheet helpers in 8.0.0.2 or a later 8.0.0.x fix pack, reload them to get the latest updates. See "Updating the timesheet helpers" on page 46

  If you did not load the timesheet helpers in 8.0.0.x, load them to get the new helpers and reports. See "Loading the timesheet helpers" on page 44.

  When you are ready to start using the new timesheet entry helper, disable the old one. See "Disabling the old timesheet entry helper" on page 47.
- If you use IBM OpenPages Model Risk Governance and you want to use IBM Watson OpenScale, load the required fields and field groups. See "Updating MRG" on page 48.
- If you use IBM OpenPages Third Party Risk Management (prevously called OpenPages Third Party Risk Management), load the new dashboard. See "Updating TPRM" on page 48.
- If you want to use the sample workflows that are provided with OpenPages, see "Loading the sample workflows" on page 49.

If you upgraded from 7.4 or 8.0.0.1

- If you use IBM OpenPages Internal Audit Management, load the new timesheet helpers and reports. See "Loading the timesheet helpers" on page 44.

  When you are ready to start using the new timesheet entry helper, disable the old one. See "Disabling the old timesheet entry helper" on page 47.
- If you use IBM OpenPages Model Risk Governance and you want to use IBM Watson OpenScale, load the required fields and field groups. See "Updating MRG" on page 48.
- If you use IBM OpenPages Third Party Risk Management (previously called OpenPages Third Party Risk Management), load the new dashboard. See "Updating TPRM" on page 48.
- If you want to use the sample workflows that are provided with OpenPages, go to  > **Solution Configuration** > **Workflows**. Review the workflows. Ensure that your environment has the object types, field groups, and fields that are required by the workflow. When you're ready to use a workflow, enable it.

Loading the timesheet helpers

If you use the IBM OpenPages Internal Audit Management solution, load the timesheet helpers and reports.

About this task

You run a script to load the new Timesheet Entry Helper and the Timesheet Approval Helper. The script does not remove the old helpers. Your users can continue to use the old helpers.

Procedure

1. Log on to the admin application server as a user with administrative privileges.
2. Open a command prompt or shell.
3. Go to the `/OP_<version>_Main/OP_<version>_Configuration/Modules/Upgrade/IAM/` directory.
4. Open the `schema_loader_modules_properties.sh|.bat` file in a text editor.

   Update the following properties:

```
OBJMGR_HOME=<full_path_to_OP_HOME/bin>
PATCH_LOADER_DATA=<full_path_to_the_IAM_directory>
OPXUserName=<Super_Administrator_user_name>
OPXUserPassword=<Super_Administrator_password>
```

Tip: In the installation app, the super administrator is set on the **Database Server** card in the **OP Admin Username** field. You can also find the user name in the `deploy.properties` file in the `op_admin_username` parameter.

Save your changes and close the file.

For example:

- Windows:

```
OBJMGR_HOME=C:\OP\OpenPages\bin
PATCH_LOADER_DATA=C:\OP_<version>_Main\OP_<version>_Configuration\Modules\Upgrade\IAM
OPXUserName=OpenPagesAdministrator
OPXUserPassword=password
```

- Linux:

```
OBJMGR_HOME=/home/opuser/OP/OpenPages/bin
PATCH_LOADER_DATA=/home/OP_<version>_Main/OP_<version>_Configuration/Modules/Upgrade/IAM
OPXUserName=OpenPagesAdministrator
OPXUserPassword=password
```

5. Run the following script:
   - Windows:

```
openpages-modules-loader-data.bat
```

   - Linux:

```
./openpages-modules-loader-data.sh
```

6. Edit the `schema_loader_modules_properties.sh | .bat` file. Set the `OPXUserPassword` property to `****`, for security reasons.
7. Log on to the active reporting server as a user with administrative privileges.
8. Open a command prompt or shell.
9. Go to the *<CC_HOME>*`/temp/bin` directory.
10. Run the following script to import the timesheet helper reports:
    - Windows:

```
importIAMReports.bat <op_admin_username> <op_admin_password>
```

    - Linux:

```
./importIAMReports.sh <op_admin_username> <op_admin_password>
```

    Replace *<op_admin_user>* and *<op_admin_password>* with the user name and password of the OpenPages super administrator.
11. Configure the timesheet helpers.

    For more information, see the *IBM OpenPages with Watson Administrator's Guide*.
12. When you are ready to roll out the new helpers to your users, update profiles to use the new timesheet helpers and dashboards.

    Update the tabs and reports on the home page.

    - Add **Timesheet Entry Helper** and **Timesheet Approval Helper**.
    - Remove **Timesheet Entry** and **Administration Timesheet Entry**.

    For more information, see the *IBM OpenPages with Watson Administrator's Guide*.

    Update the **My Reports** > **OpenPages V6** > **Audit Management Reports** list.

    - Add the new reports: **Auditor Utilization Dashboard**, **Auditor Timesheet Dashboard**, and **Pending Timesheet Approvals Dashboard**.

- Remove any reports that you no longer need.

For more information, see the *IBM OpenPages with Watson Administrator's Guide*.

13. Optional: Disable the old **Timesheet Entry** helper.
    See "Disabling the old timesheet entry helper" on page 47.

Updating the timesheet helpers

If you use the IBM OpenPages Internal Audit Management solution, update the timesheet helpers.

Procedure

1. Log on to the admin application server as a user with administrative privileges.
2. Open a command prompt or shell.
3. Go to the `/OP_<version>_Main/OP_<version>_Configuration/Modules/Upgrade/IAM/` directory.
4. Open the `schema_loader_modules_properties.sh|.bat` file in a text editor.

   Update the following properties:

   ```
   OBJMGR_HOME=<full_path_to_OP_HOME/bin>
   PATCH_LOADER_DATA=<full_path_to_the_IAM_directory>
   OPXUserName=<Super_Administrator_user_name>
   OPXUserPassword=<Super_Administrator_password>
   ```

   Tip: In the installation app, the super administrator is set on the **Database Server** card in the **OP Admin Username** field. You can also find the user name in the `deploy.properties` file in the `op_admin_username` parameter.

   Save your changes and close the file.

   For example:

   - Windows:

     ```
     OBJMGR_HOME=C:\OP\OpenPages\bin
     PATCH_LOADER_DATA=C:\OP_<version>_Main\OP_<version>_Configuration\Modules\Upgrade\IAM
     OPXUserName=OpenPagesAdministrator
     OPXUserPassword=password
     ```

   - Linux:

     ```
     OBJMGR_HOME=/home/opuser/OP/OpenPages/bin
     PATCH_LOADER_DATA=/home/OP_<version>_Main/OP_<version>_Configuration/Modules/Upgrade/IAM
     OPXUserName=OpenPagesAdministrator
     OPXUserPassword=password
     ```

5. Run the following script:

   - Windows:

     ```
     openpages-modules-upgrade-loader-data.bat
     ```

   - Linux:

     ```
     ./openpages-modules-upgrade-loader-data.sh
     ```

6. Edit the `schema_loader_modules_properties.sh | .bat` file. Set the `OPXUserPassword` property to `****`, for security reasons.
7. Optional: If the old timesheet helpers are enabled, disable them.

Disabling the old timesheet entry helper

When you are ready to begin using the new Timesheet Entry Helper, disable the old Timesheet Entry helper.

About this task

In UAT and production environments, disable the old helper before your users begin to use the new Timesheet Entry Helper.

The script does not disable the Administration Timesheet Entry helper.

Procedure

1. Log on to the admin application server as a user with administrative privileges.
2. Open a command prompt or shell.
3. Go to the `/OP_<version>_Main/OP_<version>_Configuration/Modules/Upgrade/IAM/` directory.
4. Open the `schema_loader_modules_properties.sh|.bat` file in a text editor.

   Update the following properties:

   ```
   OBJMGR_HOME=<full_path_to_OP_HOME/bin>
   PATCH_LOADER_DATA=<full_path_to_the_IAM_directory>
   OPXUserName=<Super_Administrator_user_name>
   OPXUserPassword=<Super_Administrator_password>
   ```

   Tip: In the installation app, the super administrator is set on the **Database Server** card in the **OP Admin Username** field. You can also find the user name in the `deploy.properties` file in the `op_admin_username` parameter.

   For example:

   - Windows:

     ```
     OBJMGR_HOME=C:\OP\OpenPages\bin
     PATCH_LOADER_DATA=C:\OP\OpenPages\Module\loaderdata\IAM
     OPXUserName=OpenPagesAdministrator
     OPXUserPassword=password
     ```

   - Linux:

     ```
     OBJMGR_HOME=/home/opuser/OP/OpenPages/bin
     PATCH_LOADER_DATA=/home/opuser/OP/OpenPages/Module/loaderdata/IAM
     OPXUserName=OpenPagesAdministrator
     OPXUserPassword=password
     ```

5. Run the following script to disable the old timesheet entry helper:

   - Windows:

     ```
     disable-old-timesheet-entry-helper.bat
     ```

   - Linux:

     ```
     ./disable-old-timesheet-entry-helper.sh
     ```

6. Edit the `schema_loader_modules_properties.sh | .bat` file. Set the `OPXUserPassword` property to `****`, for security reasons.

What to do next

Update profiles to remove the old helper from the Home page and from the **My Reports** list.

Updating MRG

If you use IBM OpenPages Model Risk Governance, do the following steps to update the solution.

About this task

Do these steps to enable integration between MRG and IBM Watson OpenScale. The loader file adds the fields and field groups that are required for the integration.

Procedure

1. Copy the `MRG_OpenScale_Fields-op-config.xml` from the installation media to the admin application server.

   The file is located in the `/OP_<version>_Main/OP_<version>_Configuration/Modules/MRG/OpenScale` directory.
2. Open a command line.

   If you are using Microsoft Windows, open a command prompt with the **Run as Administrator** option.
3. Go to the *<OP_HOME>*`/bin` directory.
4. Run the following command to load the files.

   Replace *<loader-file-path>* with the location of the `MRG_OpenScale_Fields-op-config.xml` file.

   ```
   ObjectManager.cmd|sh l c <OpenPages Administrator user> <OpenPages Administrator password>
    <loader-file-path> MRG_OpenScale_Fields
   ```

   If you encounter any errors, review the log file, *<loader-file-path>*`/ObjectManager.log`.

Updating TPRM

If you use IBM OpenPages Third Party Risk Management, do the following steps to update the solution.

About this task

Do these steps to load the new dashboards for IBM OpenPages Third Party Risk Management.

Procedure

1. Create a directory on the admin application server.
2. Copy the loader files from the installation media to the directory that you created on the admin application server.
   a) Go to the `OP_<version>_Main/OP_<version>_Configuration/Modules/Upgrade/loader-data/<version>_loader_data/loaderdata/VRM/` directory.
   b) Copy the following files to the directory that you created in step 1.

   ```
   dv-VRM-Vendor-Manager-op-config.xml
   dv-app-string-keys-VRM-Vendor-Manager-op-config.xml
   ```

   c) Go to the `OP_<version>_Main/OP_<version>_Configuration/Modules/Upgrade/loader-data/<version>_loader_data/loaderdata/Dashboards/locales/` directory.
   d) Copy the following files to the directory that you created in step 1.

   ```
   /en_GB/dv-app-string-keys-VRM-Vendor-Manager-en_GB-op-config.xml
   /en_US/dv-app-string-keys-VRM-Vendor-Manager-en_US-op-config.xml
   /es_ES/dv-app-string-keys-VRM-Vendor-Manager-es_ES-op-config.xml
   /fr_FR/dv-app-string-keys-VRM-Vendor-Manager-fr_FR-op-config.xml
   /de_DE/dv-app-string-keys-VRM-Vendor-Manager-de_DE-op-config.xml
   /it_IT/dv-app-string-keys-VRM-Vendor-Manager-it_IT-op-config.xml
   /ja_JP/dv-app-string-keys-VRM-Vendor-Manager-ja_JP-op-config.xml
   /pt_BR/dv-app-string-keys-VRM-Vendor-Manager-pt_BR-op-config.xml
   ```

```
/zh_CN/dv-app-string-keys-VRM-Vendor-Manager-zh_CN-op-config.xml
/zh_TW/dv-app-string-keys-VRM-Vendor-Manager-zh_TW-op-config.xml
```

3. Open a command line.

   If you are using Microsoft Windows, open a command prompt with the **Run as Administrator** option.
4. Go to the *<OP_HOME>*/bin directory.
5. Run the following commands to load the files.

   Replace *<loader-file-path>* with the directory that you created in step 1.

```
ObjectManager.cmd|sh l c <OpenPages Administrator user> <OpenPages Administrator password>
 <loader-file-path> dv-VRM-Vendor-Manager
ObjectManager.cmd|sh l c <OpenPages Administrator user> <OpenPages Administrator password>
 <loader-file-path> dv-app-string-keys-VRM-Vendor-Manager
ObjectManager.cmd|sh l c <OpenPages Administrator user> <OpenPages Administrator password>
 <loader-file-path> dv-app-string-keys-VRM-Vendor-Manager-en_GB
ObjectManager.cmd|sh l c <OpenPages Administrator user> <OpenPages Administrator password>
 <loader-file-path> dv-app-string-keys-VRM-Vendor-Manager-en_US
ObjectManager.cmd|sh l c <OpenPages Administrator user> <OpenPages Administrator password>
 <loader-file-path> dv-app-string-keys-VRM-Vendor-Manager-es_ES
ObjectManager.cmd|sh l c <OpenPages Administrator user> <OpenPages Administrator password>
 <loader-file-path> dv-app-string-keys-VRM-Vendor-Manager-fr_FR
ObjectManager.cmd|sh l c <OpenPages Administrator user> <OpenPages Administrator password>
 <loader-file-path> dv-app-string-keys-VRM-Vendor-Manager-de_DE
ObjectManager.cmd|sh l c <OpenPages Administrator user> <OpenPages Administrator password>
 <loader-file-path> dv-app-string-keys-VRM-Vendor-Manager-it_IT
ObjectManager.cmd|sh l c <OpenPages Administrator user> <OpenPages Administrator password>
 <loader-file-path> dv-app-string-keys-VRM-Vendor-Manager-ja_JP
ObjectManager.cmd|sh l c <OpenPages Administrator user> <OpenPages Administrator password>
 <loader-file-path> dv-app-string-keys-VRM-Vendor-Manager-pt_BR
ObjectManager.cmd|sh l c <OpenPages Administrator user> <OpenPages Administrator password>
 <loader-file-path> dv-app-string-keys-VRM-Vendor-Manager-zh_CN
ObjectManager.cmd|sh l c <OpenPages Administrator user> <OpenPages Administrator password>
 <loader-file-path> dv-app-string-keys-VRM-Vendor-Manager-zh_TW
```

   If you encounter any errors, review the log file, *<loader-file-path>*/ObjectManager.log.

Loading the sample workflows

If you upgraded or migrated from 8.0.0.2 or later and you want to use the sample workflows, you need to load them.

About this task

When you upgrade or migrate from 8.0.0.2 or later, the sample workflows are not loaded automatically. Your system might have workflows with the same names as the samples. Or your environment might not have all of the object types, field groups, or fields that the sample workflows require.

Analyze the sample workflow files, and then load the workflows that you want to use.

Important: If you load a sample workflow that has the same name as a workflow in your environment, your workflow will be overwritten.

Procedure

1. Log in to the admin application server as a user with administrative privileges.
2. Copy the OP_*<version>*_Main/OP_*<version>*_Configuration/Modules/Upgrade/loader-data/8200_loader_data/loaderdata/workflows directory in the installation media to the following directory on the admin application server:

   <OP_HOME>/addon_module/loaderdata/
3. Locate the loader file for the sample workflow that you want to load.

   For example, if you want to load the Finding workflow, locate the sample-workflow-Finding-op-config.xml file.
4. Analyze the file. Verify that your environment has all of the object types, fields, and field groups that are required by the workflow.

5. Load the sample workflow:
   a) Go to the *<OP_HOME>*/bin directory.
   b) Run the following command:

```
ObjectManager.cmd|.sh l c <OpenPages Administrator user>
 <OpenPages Administrator password> <OP_HOME>/addon_module/loaderdata/workflows
 <loader_file>
```

   For example, to load the Finding workflow, run the following command:

```
ObjectManager.cmd|.sh l c <OpenPages Administrator user>
   <OpenPages Administrator password> <OP_HOME>/addon_module/loaderdata/workflows
   sample-workflow-Finding
```

   c) After the loading process is complete, review the ObjectManager log.
6. Repeat steps 3-5 for each sample workflow that you want to load.
7. Log in to IBM OpenPages with Watson.
8. Click > **Solution Configuration** > **Workflows**.
9. Review the sample workflows that you loaded.

Updating the reporting schema

Update the reporting schema.

Procedure

1. Log in to OpenPages as a user with administrative privileges.
2. Enable System Admin Mode (SAM). Click , and then click **Enable System Admin Mode**.
3. Click > **System Configuration** > **Reporting Schema**.
4. Click **Update**.
5. Click **Refresh** until the process is 100% complete.
6. Disable SAM. Click , and then click **Disable System Admin Mode**.

Regenerating the reporting framework

After you upgrade IBM OpenPages with Watson, you might need to regenerate the reporting framework.

Version 8.2 adds a number of new system fields and object types. If you plan to use the new capabilities and want to be able to access the new fields and object types in reports, regenerate the reporting framework.

You also need to regenerate the reporting framework if any of the following cases applies to you:

- You use more than one URL to access OpenPages. When you regenerate the framework, select **Framework Model**, **Custom Query Subjects**, and **All Models**.
- You added new fields and you want to use the new fields in reports.

Regenerate the reporting framework after you complete all other installation and upgrade tasks.

For more information, see Generating the reporting framework.

Additional tasks for upgrades

You might want to complete additional tasks for an OpenPages upgrade.

Performing a silent upgrade

You can upgrade from the command line.

Before you begin

Complete the following preparation tasks:

- "Review new features and fixes" on page 11
- "Backing up your environment" on page 11
- "Upgrade prerequisite software " on page 14
- "Verifying servers before you upgrade" on page 21
- "Upgrade the OpenPages database (Oracle)" on page 23
- "Preparing the installation server" on page 29
- If you installed the agent software on remote servers manually, ensure that the agent software is updated and that the agents are running.

Procedure

1. Log on to the installation server computer as the user who installed the installation server.

   Alternatively, you can log in as any user who has full permissions on the installation server directories and who can run `Node.js`.
2. Go to the *<Installation_server_home>*`/src/deployment/`*<deployment name>* directory.
3. Edit the `deploy.properties` file.
   a) Change the value of the `task` property to `upgrade`.
   b) Change the value of the `maintenance_version` property to `8.2`.
   c) Update the value of the `install_db` property. Set it to `done`.

      For more information, see the *IBM OpenPages with Watson Installation and Deployment Guide*.
   d) Save and close the file.
4. Run the silent installation from the command line.
   a) Open a command prompt or open a shell window as an administrator.
   b) Go to the *<Installation_server_home>* directory.
   c) Run the following command:

      ```
      npm run silent <deployment name> acceptLicense
      ```

      Note: Do not close the command prompt or shell window until after the process completes.
5. Check the logs to ensure that the installation is successful.

Rolling back an upgrade

If you backed up your environment before you upgraded and you did not uninstall IBM WebSphere Application Server Network Deployment, you can roll back the upgrade.

Before you begin

To roll back an upgrade, you need the following backup files:

- The backup directories that you created before you upgraded.
- The backup of the `openpages-storage` directory that you created before you upgraded.
- The database backup that you created before you upgraded.
- Any other backup files that you created before the upgrade, such as custom reports, JSPs, and so on.

You also need IBM WebSphere Application Server Network Deployment and the version of Cognos Analytics that you used prior to the upgrade.

About this task

The following procedure applies to in-place upgrades, not migration upgrades.

In these steps, the backup directories use the name `OpenPages8001Backup`. Specify the name that you chose when you created the backup directories.

Procedure

1. Stop all servers:
   - OpenPages application servers (admin and non-admin)
   - IBM Cognos servers (active and standby)
   - OpenPages Framework Model Generator service
   - OpenPages search server

   For more information, see the *IBM OpenPages with Watson Installation and Deployment Guide*.
2. For application servers:
   a) If you are using Microsoft Windows, delete the Windows services for OpenPages 8.2.
   b) If you are using vertical cluster members, run the following command on each of them:

   ```
   sc delete <server name>
   ```

   c) Restore the backup directories that were created before the upgrade.
3. For the database server:
   a. If the database software was upgraded, reinstate the previous database software version.
   b. Restore the database using the database backup that was created before the database upgrade.
4. For all other servers, restore the backup directories that were created before the upgrade.

   After all servers are restored and all third-party products are at the versions required by the previous OpenPages deployment, the previous OpenPages deployment works without further actions.
5. Restore the backup of the `openpages-storage` directory that you created before the upgrade.
6. Restore any other backup files that you created before the upgrade, such as custom reports, JSPs, and so on.
7. Start all servers:
   - OpenPages application servers
   - IBM Cognos servers
   - OpenPages Framework Model Generator service
   - OpenPages search server

   For more information, see the *IBM OpenPages with Watson Installation and Deployment Guide*.

Manually loading the configuration data for an upgrade

When you upgrade IBM OpenPages with Watson to version 8.2, IBM OpenPages with Watson automatically loads the application data and enables user access to the standard Cognos Analytics reports. In limited situations, you can manually upgrade the loader configuration data.

Before you begin

IBM OpenPages with Watson must be installed.

The OpenPages services must be running.

About this task

If the upgrade loader files that are executed during the upgrade have errors, you can correct the issues that caused the errors and then run the upgrade loader files manually.

Procedure

1. Log on to the OpenPages admin application server as a user with administrative privileges.
2. Go to the *<OP_HOME>*/installer/maintenance/upgrade-8.2/addon_module/loaderdata directory.
3. Make a backup copy of the schema_loader_properties.sh|.bat file.
4. Open the original schema_loader_properties file in a text editor.
5. In the following line, update the password for the OpenPages Super Administrator to clear text.

   ```
   SET OPXUserName=<Super_Administrator_user_name>
   SET OPXUserPassword=********
   ```

   The default user name is OpenPagesAdministrator.

   The password for the OPXUserName user is masked by asterisks (***). Replace the mask with clear text.
6. Save and close the file.
7. Go to the *<OP_HOME>*/bin directory.
8. Edit the ObjectManager.properties file and update the following settings as shown:

   ```
   configuration.manager.vendor.mode=true
   configuration.manager.force.update.object.strings=false
   configuration.manager.force.update.application.strings=false
   configuration.manager.disable.triggers=true
   ```
9. Save and close the file.
10. Depending on your upgrade path, run the scripts in the order that is listed:

Upgrade path	Windows files to run
7.4 or 8.0.0.1 to 8.2	• openpages-op800x-to-8100-loader-data.bat • openpages-op810x-to-8200-loader-data.bat • op-sysviews-loader.bat • op-sampleWorkflows-upgrade-loader.bat
8.0.0.2 or a later 8.0.0.x fix pack to 8.2	• openpages-op800x-to-8100-loader-data.bat • openpages-op810x-to-8200-loader-data.bat • op-sysviews-loader.bat
8.1.x to 8.2	• openpages-op810x-to-8200-loader-data.bat • op-sysviews-loader.bat

Upgrade path	Linux files to run
7.4 or 8.0.0.1 to 8.2	• openpages-op800x-to-8100-loader-data.sh • openpages-op810x-to-8200-loader-data.sh • op-sysviews-loader.sh • op-sampleWorkflows-upgrade-loader.sh

Upgrade path	Linux files to run
8.0.0.2 or a later 8.0.0.x fix pack to 8.2	• `openpages-op800x-to-8100-loader-data.sh` • `openpages-op810x-to-8200-loader-data.sh` • `op-sysviews-loader.sh`
8.1.x to 8.2	• `openpages-op810x-to-8200-loader-data.sh` • `op-sysviews-loader.sh`

11. Go to the *<OP_HOME>*`/bin` directory.
12. Edit the `ObjectManager.properties` file and update the following settings as shown:

```
configuration.manager.vendor.mode=false
configuration.manager.disable.triggers=false
```

13. Save and close the file.
14. Go to the *<OP_HOME>*`/installer/maintenance/upgrade-8.2/addon_module/loaderdata` directory.
15. Open the `schema_loader_properties` file in a text editor.
16. In the following line, hide the clear text password for the OpenPages Super Administrator by changing it to asterisks (***).

```
SET OPXUserPassword=********
```

17. Save and close the file.
18. Restart the OpenPages services.

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service. This document may describe products, services, or features that are not included in the Program or license entitlement that you have purchased.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Location Code FT0
550 King Street
Littleton, MA

01460-1250
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

This Software Offering does not use cookies or other technologies to collect personally identifiable information.

Copyright

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written.

These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

Trademarks

IBM, the IBM logo and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information."

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

The registered trademark Linux is used pursuant to a sublicense from the Linux Foundation, the exclusive licensee of Linus Torvalds, owner of the mark on a worldwide basis.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Red Hat®, JBoss®, OpenShift®, Fedora®, Hibernate®, Ansible®, CloudForm, RHCA®, RHCE®, RHCSA®, Ceph®, and Gluster® are trademarks or registered trademarks of Red Hat, Inc. or its subsidiaries in the United States and other countries.

UNIX is a registered trademark of The Open Group in the United States and other countries.