Migrating artifacts from IBM Workload Deployer to IBM SmartCloud Orchestrator

Note

Before using this information and the product it supports, read the information in “Notices” on page 33.

Contents

1 Migrating from IBM Workload Deployer to IBM SmartCloud Orchestrator

This document describes the process for migrating virtual application patterns, virtual system patterns, shared services, and other artifacts from the IBM® Workload Deployer environment to IBM SmartCloud® Orchestrator.

Overview

You should use this procedure if you have an existing IBM Workload Deployer Version 3.1.0.x 2U appliance managing an existing VMware, or PowerVM® cloud and instances running in the cloud, and you want to migrate these artifacts to IBM SmartCloud Orchestrator Version 2.3. For this procedure, you install IBM SmartCloud Orchestrator Version 2.3 in the same network as IBM Workload Deployer, perform the migration, and then use IBM SmartCloud Orchestrator Version 2.3 to manage new deployments in the cloud environment.

This migration solution is not a full migration where all artifacts and running instances are migrated to SmartCloud Orchestrator. With this solution, IBM SmartCloud Orchestrator can deploy only new instances of existing IBM Workload Deployer artifacts (such as virtual system patterns, virtual application patterns, and shared services) into a new cloud. Existing workloads that were previously deployed using IBM Workload Deployer must continue running under that environment. The architecture of IBM SmartCloud Orchestrator is significantly different than that of IBM Workload Deployer. As a result, some parts of this procedure are automated, while others require manual steps to complete.

Considerations for migrating various types of artifacts

The migration process requires handling a variety of artifacts in different ways.

Generally, the migration tool performs the migration of various artifacts in two steps:

1. Artifacts are exported from the IBM Workload Deployer environment and stored on a temporary system.
2. Artifacts are imported from the temporary system into the IBM SmartCloud Orchestrator environment.

The following types of artifacts are handled by the migration tool:

- Virtual images
- Virtual system patterns
- Script packages
- Add-ons
- Virtual application patterns
- User groups
- Users

Pattern type artifacts cannot be automatically exported from IBM Workload Deployer. You must specify the location of the pattern types that you want to migrate. As the migration tool exports artifacts, the tool exports only general information about pattern types to the target JSON object files. You will need to edit these JSON files manually to specify the path to include with the pattern type.

Emergency Fixes can be exported from IBM Workload Deployer, but they are not supported in IBM SmartCloud Orchestrator Version 2.3.

Artifacts related to networking are handled by OpenStack. Due to the variety and complexity of network configurations, you must use OpenStack console tools to manually set up a working connection between OpenStack and hypervisors. However, in general you will be moving to new hypervisors, so this is not considered to be part of the migration process.

User Groups and Users are stored differently in IBM SmartCloud Orchestrator than in IBM Workload Deployer. IBM SmartCloud Orchestrator uses Keystone, which manages all users and *tenants*. A tenant is an OpenStack object, and represents a group of users. In IBM SmartCloud Orchestrator, tenants generally replace user groups, but these two types of objects are not equivalent.

During the migration process, the migration tool creates a default domain, *IWD_Default_Domain*, and all users are added to this domain. A *domain* is also specific to OpenStack, and is used to group *projects*. The migration tool also creates a default project, *IWD_Default_Project*, which replaces the *Everyone* group in IBM Workload Deployer. All migrated users are added to this project.

Other user groups are handled by the migration tool, transforming them into IBM SmartCloud Orchestrator projects by the same name, and stored in the *IWD_Default_Domain* domain.

IBM SmartCloud Orchestrator has only 3 roles that are enabled for users in projects:

- Admin
- Member
- Catalog Editor

Because IBM Workload Deployer supports many types of roles, all users are migrated to IBM SmartCloud Orchestrator and assigned the roles of Member and Catalog Editor. The typical administrative user in IBM Workload Deployer, *cbadmin* is not migrated, because there is already a default *admin* user in IBM

SmartCloud Orchestrator. If LDAP is used in IBM Workload Deployer, users can be migrated into the Keystone environment and all users will be available with their existing passwords. If LDAP is not used, migrated users will have their passwords set to a default value.

Migrating environment profiles is not useful, because you are moving to a new cloud environment. Even if you intended to use the same cloud, cloud group names are generated by OpenStack, so you would still have to manually choose which cloud should be associated with which environment profile.

Migration of virtual system instances, virtual application instances, or shared services is not supported. You should plan to deploy the same patterns again in IBM SmartCloud Orchestrator, and then perform suitable data migration for components running in a particular instance. For example, IBM DB2® and IBM Business Process Manager can migrate data from one deployed environment to another. Shared services should be handled the same as virtual application instances.

Virtual appliances are not migrated, because the are not supported in IBM SmartCloud Orchestrator.

Important: Because of inherent differences between the architectures of IBM Workload Deployer and IBM SmartCloud Orchestrator, some types of artifacts that are migrated from IBM Workload Deployer, such as add-ons, script packages, and pattern types, and others, might not work properly if they are incompatible when used with other artifacts native to the IBM SmartCloud Orchestrator environment.

For example, if you migrate an OVA image that includes the `scp-cloud-init` component, a deployment in IBM SmartCloud Orchestrator does not work because the OVA image must include the Virtual System Activation Engine to deploy the image successfully.

As another example, suppose that a script from the **Default add disk** add-on was used in IBM Workload Deployer to create a custom add-on. This custom add-on, after being migrated to IBM SmartCloud Orchestrator, cannot be used to create an additional disk on a virtual machine, because it is not compatible with the architecture in IBM SmartCloud Orchestrator. In this situation, you must instead use the equivalent default add disk add-on provided with IBM SmartCloud Orchestrator. This limitation applies to other default add-ons migrated from IBM Workload Deployer.

Considerations for the migration environment

The migration process can be performed in several different environments.

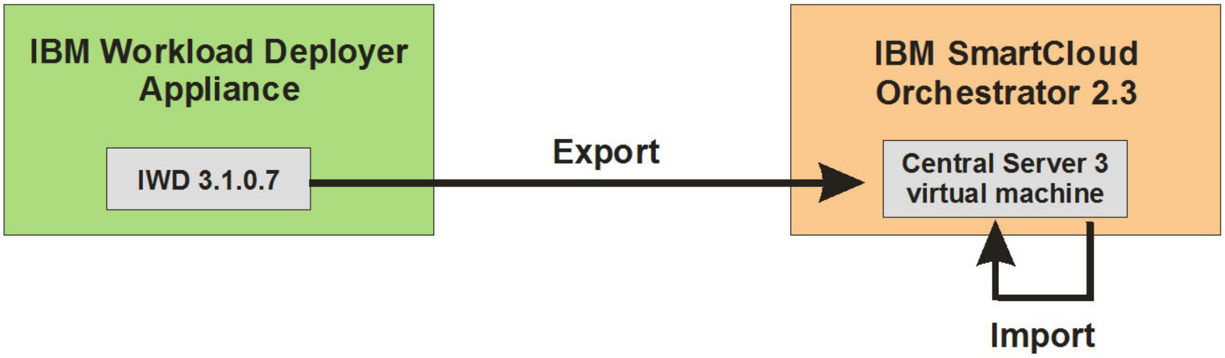
You can perform this migration using one of the following types of migration environments:

- **Environment 1:** Directly between IBM Workload Deployer and IBM SmartCloud Orchestrator
- **Environment 2:** Indirectly by using one separate migration system
- **Environment 3:** Indirectly by using two separate migration systems

Environment 1: Directly between IBM Workload Deployer and IBM SmartCloud Orchestrator

In this environment, the migration tool is installed in the same IBM SmartCloud Orchestrator virtual machine (for example, Central Server 3) where IBM Workload Deployer is already installed.

Figure 1. Migration tool installed on the IBM SmartCloud Orchestrator virtual machine.

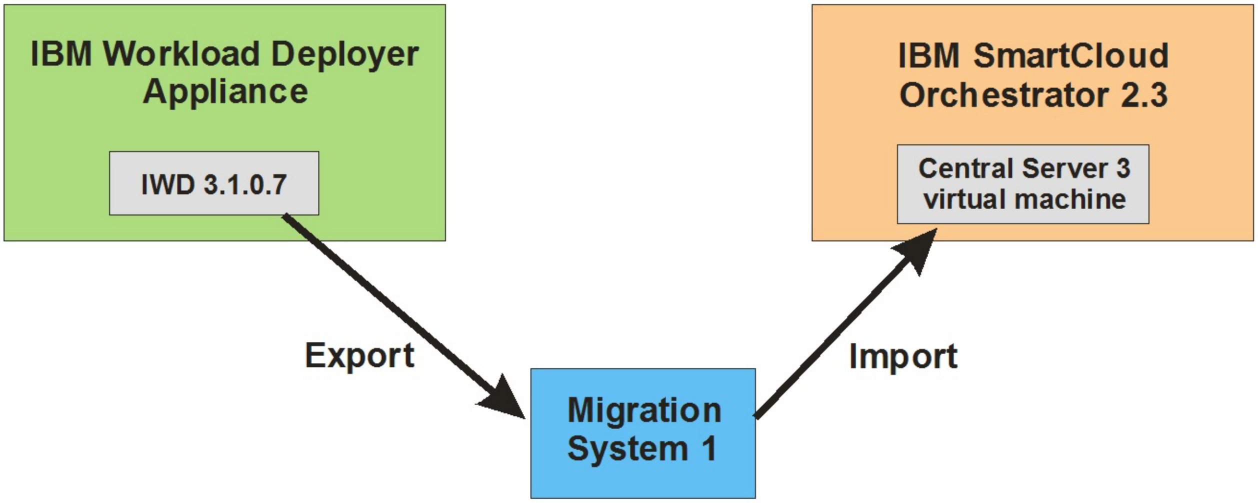

IBM Workload Deployer Appliance
IWD 3.1.0.7
Export
IBM SmartCloud Orchestrator 2.3
Central Server 3 virtual machine
Import

This environment can provide the fastest migration because the import process obtains its data from the same local host where the migration tool resides. In addition, the migration tool requires the IBM Workload Deployer command line interface to be available, and in this environment it is already installed on the IBM SmartCloud Orchestrator virtual machine.

Environment 2: Indirectly by using one separate migration system

In this environment, the migration tool is installed on a separate system from either the IBM Workload Deployer appliance or IBM SmartCloud Orchestrator. You also must download and install a version of IBM Workload Deployer command line interface on this separate system.

Figure 2. Migration tool installed on a separate migration system with network connection to IBM Workload Deployer appliance and SmartCloud Orchestrator.


IBM Workload Deployer Appliance
IWD 3.1.0.7
IBM SmartCloud Orchestrator 2.3
Central Server 3 virtual machine
Export
Import
Migration System 1

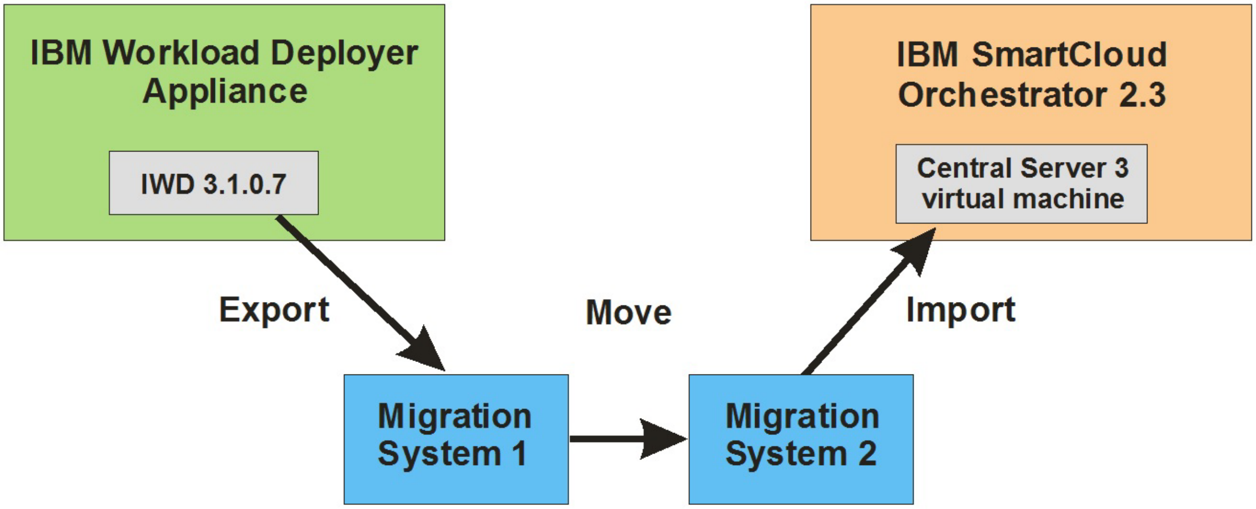
This separate migration system must be running a supported Linux operating system, and the migration tool is installed on this system. The migration system must have network access to both the IBM Workload Deployer appliance and to the IBM SmartCloud Orchestrator virtual machine. Artifacts are first exported from the IBM Workload Deployer appliance and stored on this migration system, and then the artifacts are imported from this system to IBM SmartCloud Orchestrator.

You need to download a version of the IBM Workload Deployer command line interface to this system. The actual version you install is not important, because the required libraries for specific versions of the command line interface are downloaded as needed from either the IBM Workload Deployer appliance or from the IBM SmartCloud Orchestrator virtual machine when the migration system connects to them for the first time.

Environment 3: Indirectly by using two separate migration systems

In this environment, you might have network issues that prevent you from establishing a network connection between the systems. In this situation the migration tool (and the IBM Workload Deployer command line interface) is installed on two separate systems from either the IBM Workload Deployer appliance or IBM SmartCloud Orchestrator.

Figure 3. Migration tool installed on two separate migration systems with no network connection between the IBM Workload Deployer appliance and the IBM SmartCloud Orchestrator virtual machine.


IBM Workload Deployer Appliance
IWD 3.1.0.7
IBM SmartCloud Orchestrator 2.3
Central Server 3 virtual machine
Export
Move
Import
Migration System 1
Migration System 2

Both of these separate migration systems must be running a supported Linux operating system, and the migration tool is installed on both systems. The IBM Workload Deployer command line interface must also be installed on each system. The migration tool on Migration System 1 performs only the export process from the IBM Workload Deployer appliance. After the export process completes, you must move the artifact data to Migration System 2, and then run the migration tool on this second system to perform the import process into IBM SmartCloud Orchestrator.

Prerequisites

There are a number of prerequisites that must be met before you perform this migration process.

Before you begin this migration procedure, ensure that the following prerequisites are met:

- Your IBM Workload Deployer environment must be upgraded to Version 3.1.0.7 with the latest fix pack installed.
- You can have IBM SmartCloud Orchestrator Version 2.3 installed on the same or a different network than IBM Workload Deployer. You should have HTTP communication enabled between IBM Workload Deployer and IBM SmartCloud Orchestrator virtual machines.
- Because the size of some virtual images can be quite large, you should plan for sufficient temporary storage by attaching an additional disk or providing a network share where the artifacts will be stored during the migration process.
- Your environment must be running a supported Red Hat Enterprise Linux operating system, with sufficient storage available to perform the migration.
- You must set up a new cloud that will be managed by IBM SmartCloud Orchestrator.
- You will need to provide pattern types that are not included with the IBM Workload Deployer default data. IBM SmartCloud Orchestrator provides only the foundation pattern types.

In addition to these prerequisites, keep the following requirements in mind:

- The migration tool must run in a supported Red Hat Enterprise Linux operating system.
- The migration might take many hours depending on the network and the amount of data to migrate. To minimize the duration of the migration, the migration should only export data from IBM Workload Deployer that does not already exist in IBM SmartCloud Orchestrator .
- The data exported from IBM Workload Deployer might contain passwords or other sensitive information which are not encrypted. When the migration process finishes or completes unsuccessfully, ensure that you delete the exported data.
- The migration tool supports the export from IBM Workload Deployer and import into IBM SmartCloud Orchestrator the following items:
  - Virtual application patterns and shared services, and their dependent artifacts (such as plug-ins, pattern types, and virtual application templates). Pattern types cannot be exported; you must upload any pattern types that are not included in default data.
  - Virtual system patterns and their dependent artifacts (such as virtual images, script packages, and add-ons).

2 Installing the migration tool

Depending on your migration environment, you install and configure the migration tool in several different ways.

Copying the migration tool

The migration tool is provided in a compressed archive file, named `MigrationTool.tar.gz`. Depending on your migration environment, you copy this file to the location where you plan to run the tool. You will then unpack this compressed file and configure it to perform the migration.

If you are running the migration tool in **Environment 1**, copy the archive file to the virtual machine in IBM SmartCloud Orchestrator where IBM Workload Deployer is running (for example, *Central Server 3*).

If you are running the migration tool in **Environment 2**, copy the archive file to the separate *Migration System 1*.

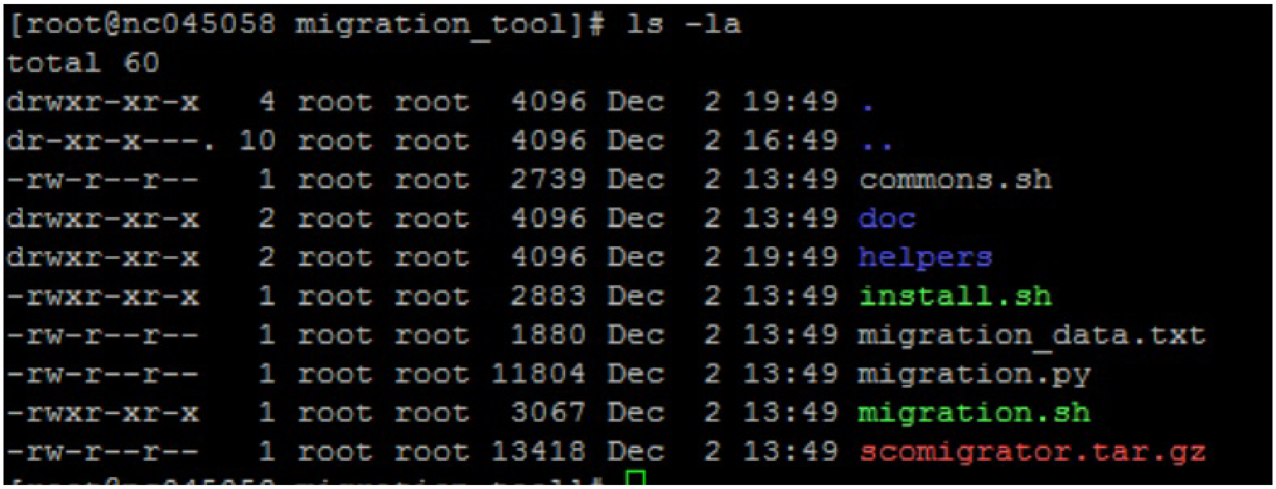
If you are running the migration tool in **Environment 3**, copy the archive file to both separate systems, *Migration System 1* and *Migration System 2*.

Unpacking the migration tool

After copying the `MigrationTool.tar.gz` file to the intended location, complete the following steps for each copy of the file:

1. Unpack the contents of the file by issuing the following command:
   ```
   tar -xzvf MigrationTool.tar.gz
   ```
2. Verify that the `migration_tool` directory has been created.
3. Change the directory location to the `migration_tool` directory.
4. Run the following command to enable the `install.sh` and `migration.sh` scripts for execution:
   ```
   chmod +x install.sh migration.sh
   ```
5. Your `migration_tool` directory should now look similar to the following example:

Figure 4. Contents of the migration_tool directory.



```
[root@nc045058 migration_tool]# ls -la
total 60
drwxr-xr-x   4 root root  4096 Dec  2 19:49 .
dr-xr-x---. 10 root root  4096 Dec  2 16:49 ..
-rw-r--r--   1 root root  2739 Dec  2 13:49 commons.sh
drwxr-xr-x   2 root root  4096 Dec  2 13:49 doc
drwxr-xr-x   2 root root  4096 Dec  2 19:49 helpers
-rwxr-xr-x   1 root root  2883 Dec  2 13:49 install.sh
-rw-r--r--   1 root root  1880 Dec  2 13:49 migration_data.txt
-rw-r--r--   1 root root 11804 Dec  2 13:49 migration.py
-rwxr-xr-x   1 root root  3067 Dec  2 13:49 migration.sh
-rw-r--r--   1 root root 13418 Dec  2 13:49 scomigrator.tar.gz
[root@nc045058 migration_tool]#
```

Configuring the migration tool parameters

Before starting the migration process, you need to configure a number of parameters defined in the migration_data.txt file.

In the migration_tool directory, open the migration_data.txt file. The default contents are as follows:

```
##########################################
# Required data
##########################################
IWD_CLI="/opt/ibm/cli/deployer.cli"
MIGRATION_DEST="/root/migration"

##########################################
# IWD Appliance data
##########################################
# IWD_3107_HOSTNAME=<IP of IWD 3107>
# IWD_3107_USER=<admin user>
# IWD_3107_PASSWORD=<admin password>

##########################################
# SCO data
##########################################
# SCO_HOSTNAME=<IP of SCO IWD - Central Server 3>
# SCO_USER=<admin user>
# SCO_PASSWORD=<admin password>
# GATEWAY_HOSTNAME=<IP of IAAS GATEWAY - Central Server 2>
# GATEWAY_PORT=9973
##########################################
```

The first two parameters are not commented out by default, and must always be provided:

IWD_CLI
The location of the IBM Workload Deployer command line interface on the system where the migration tool will be run.

MIGRATION_DEST
The location where all the artifacts will be temporarily stored during the migration process.

If you are running the migration tool in either **Environment 1** or **Environment 2**, remove the comment (# symbol in column 1) from all of the parameters in both the IWD Appliance data and SCO data sections. The installation process will install the migration tool for exporting from IBM Workload Deployer as well as for importing to IBM SmartCloud Orchestrator.

If you are running in **Environment 3** (using two migration systems), you need to edit this migration_data.txt file on both systems and complete the following steps:

- Edit the migration_data.txt on *Migration System 1*, and remove the comment symbol only from the parameters in the IWD Appliance data section. The installation process will install the migration tool for exporting artifact data from the IBM Workload Deployer appliance to *Migration System 1*.
- Edit the migration_data.txt on *Migration System 2*, and remove the comment symbol only from the parameters in the SCO data section. The installation process will install the migration tool for importing artifact data from *Migration System 2* to IBM SmartCloud Orchestrator.

If the migration tool is performing the export process, specify values for the parameters in the IWD Appliance data section of migration_data.txt as follows:

IWD_3107_HOSTNAME
Specify the IP address of the IBM Workload Deployer appliance system where IBM Workload Deployer version 3.1.0.7 is installed. This is a required parameter for the installation process to run successfully.

IWD_3107_USER
Specify a valid user name to access the IBM Workload Deployer appliance. This is a required parameter for the installation process to run successfully.

IWD_3107_PASSWORD
Specify a valid password to accompany the provided user name. This is an optional parameter in this file. If you do not specify the password, the migration tool prompts you to enter it from the keyboard.

If the migration tool is performing the import process, specify values for the parameters in the `SCO data` section of `migration_data.txt` as follows:

SCO_HOSTNAME
Specify the IP address of the virtual machine where IBM Workload Deployer is running in the IBM SmartCloud Orchestrator environment. Typically this is the IP address for *Central Server 3*. This is a required parameter for the installation process to run successfully.

SCO_USER
Specify the global SmartCloud Orchestrator administrator user name. This is a required parameter for the installation process to run successfully.

SCO_PASSWORD
Specify the global SmartCloud Orchestrator administrator password. This is an optional parameter in this file. If you do not specify the password, the migration tool prompts you to enter it from the keyboard.

GATEWAY_HOSTNAME
Specify the IP address of the virtual machine where the IAAS Gateway is set in the IBM SmartCloud Orchestrator environment. Typically this is the IP address for *Central Server 2*. This is a required parameter for the migration tool to import users and user groups.

GATEWAY_PORT
This parameter is defined with the default value *9973*. If this port number was changed during the installation of IBM SmartCloud Orchestrator, update this parameter to the correct port number. This is a required parameter for the migration tool to import users and user groups.

Running the `install.sh` installation script

On each system where the migration tool was unpacked and configured, run the `install.sh` script using the following format:

```
./install.sh [-f | --file <property file>]
```

The specification of the `-f` or `--file` attribute and the property file name is optional. If you do not specify this attribute, the installation script will search the current directory for the `migration_data.txt` property file.

The installation script connects to IBM Workload Deployer and IBM SmartCloud Orchestrator as needed, and verifies the user access credentials. If the version of the IBM Workload Deployer command line interface is different than the one supported by the IBM Workload Deployer we are connecting to, the required version of the command line interface is downloaded. This is standard functionality of the IBM Workload Deployer command line interface.

When verification of credentials completes successfully, the installation process extracts the `scomigrator.tar.gz` file (located in the `migration_tool` directory) to all available versions of the command line interface.

3 Performing the export process

After unpacking, configuring, and installing the migration tool, you can run the tool to export artifacts from your IBM Workload Deployer appliance.

Export command syntax

To run the migration tool for exporting artifacts from IBM Workload Deployer, use the `migration.sh` script, located in the `migration_tool` directory.

The general syntax of this script command and options is as follows:

```
./migration.sh -m|--mode <mode> [-a|--artifacts <artifact list> -f|--file <property file>]
```

You can specify the following options:

-m|--mode
The mode of migration. Valid values are *export* or *import*.

-a|--artifacts
The list of artifacts to be migrated. This is a list of artifact types, separated by commas, with no blank spaces in between. Valid values include:

- **all**: migrate all artifacts
- **user**: migrate users and user groups
- **script**: migrate script packages
- **addon**: migrate add-ons
- **virtimg**: migrate virtual images
- **vsys**: migrate virtual system patterns
- **vapp**: migrate virtual application patterns
- **ptype**: migrate pattern types

If you do not specify this option, the default value `all` is used.

You can optionally surround the list of options with single or double quotation marks, such as 'addon,vsys,script' or "vapp,ptype" and so on. Do not add blank spaces inside quotation marks.

Valid artifact list examples:

```
-a all
-a user,vsys,script,vapp,virtimg
-artifacts "vapp,vsys,addon"
-a 'ptype,vapp,user'
```

-f|--file
The name of the property file containing the migration process attributes. If you do not specify this option, the default value `migration_data.txt` is used.

Examples:

- `./migration.sh -m export`

  This command performs an export operation on all available artifacts, and uses the default property file, `migration_data.txt`.
- `./migration.sh --mode export -a all`

  This command performs an export operation on all available artifacts, and uses the default property file, `migration_data.txt`.
- `./migration.sh -m export -artifacts vapp`

This command performs an export operation only on virtual application pattern artifacts, and uses the default property file, `migration_data.txt`.

- `./migration.sh -m export -a "script,addon,vsys" -f migration_special.txt`

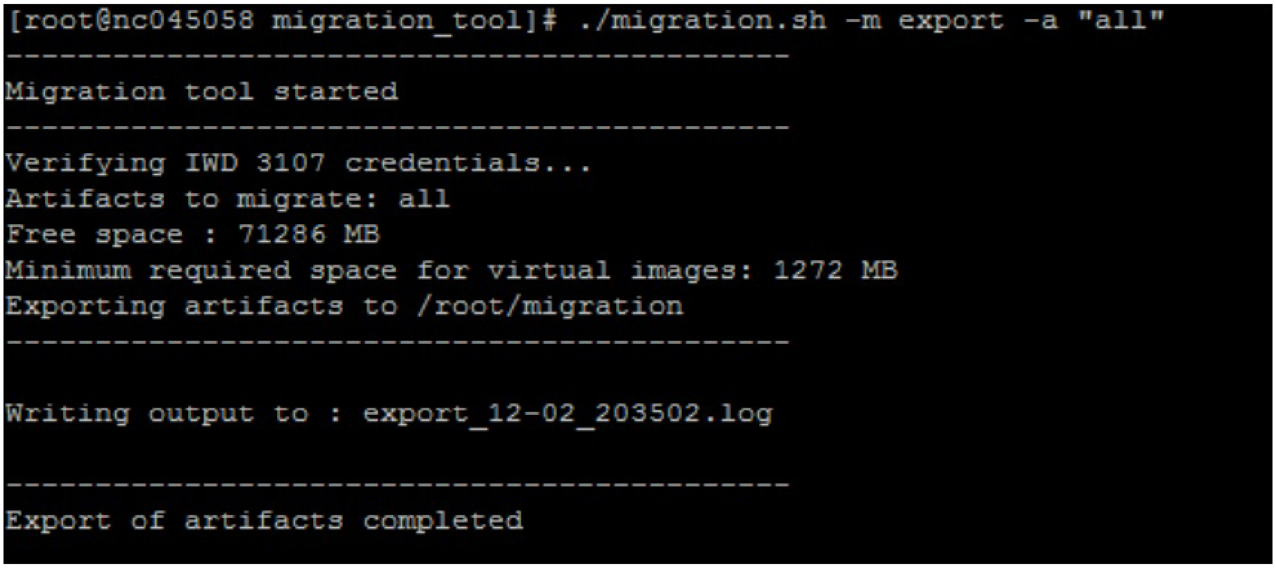
  This command performs an export operation only on script packages, add-ons, and virtual system pattern artifacts, and uses the property file, `migration_special.txt`.

Performing the export operation

The export migration operation exports all specified artifacts to the location specified in the **MIGRATION_DEST** parameter defined in the property file.

When you run the migration script to perform an export operation, the result is similar to the following example:

Figure 5. Example output when performing an export operation.



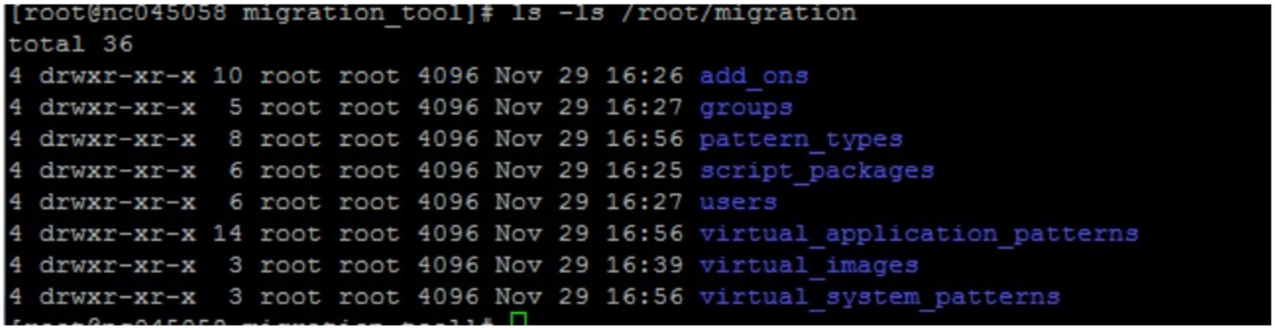
```
[root@nc045058 migration_tool]# ./migration.sh -m export -a "all"
-----------------------------------------
Migration tool started
-----------------------------------------
Verifying IWD 3107 credentials...
Artifacts to migrate: all
Free space : 71286 MB
Minimum required space for virtual images: 1272 MB
Exporting artifacts to /root/migration
-----------------------------------------

Writing output to : export_12-02_203502.log

-----------------------------------------
Export of artifacts completed
```

The export operation performs the following tasks:

- The connection credentials with the IBM Workload Deployer appliance are verified.
- The amount of free space at the specified target location is checked.
- If you are exporting virtual images, the total size of the virtual images to be exported is calculated and displayed as the minimum required space needed for the operation to complete successfully. If the size needed by the virtual images exceeds the available free space, the export operation stops.
- Based on the specified list of artifacts to export, the migration tool exports the artifacts to the target location. Output log information is written to a local log file.

As artifacts are exported, a directory for each type of artifact is created in the target location. The following example shows a listing (after the export operation completes) of the default target location `/root/migration`, after exporting all artifacts from IBM Workload Deployer:

Figure 6. Example of artifact directories created during an export operation.



```
[root@nc045058 migration_tool]# ls -ls /root/migration
total 36
4 drwxr-xr-x 10 root root 4096 Nov 29 16:26 add_ons
4 drwxr-xr-x  5 root root 4096 Nov 29 16:27 groups
4 drwxr-xr-x  8 root root 4096 Nov 29 16:56 pattern_types
4 drwxr-xr-x  6 root root 4096 Nov 29 16:25 script_packages
4 drwxr-xr-x  6 root root 4096 Nov 29 16:27 users
4 drwxr-xr-x 14 root root 4096 Nov 29 16:56 virtual_application_patterns
4 drwxr-xr-x  3 root root 4096 Nov 29 16:39 virtual_images
4 drwxr-xr-x  3 root root 4096 Nov 29 16:56 virtual_system_patterns
[root@nc045058 migration_tool]#
```

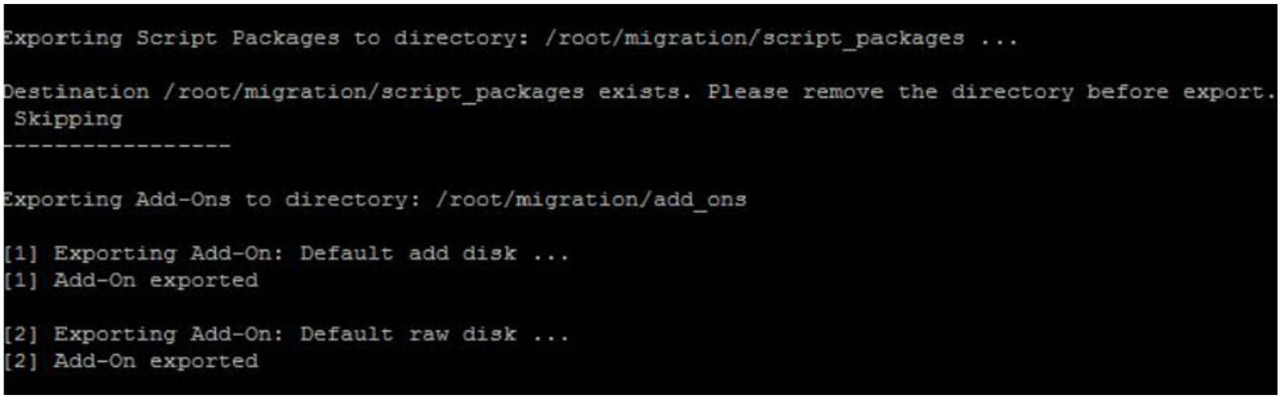
The directory names that are created include:

- `add_ons`
- `groups`
- `pattern_types`
- `script_packages`
- `users`
- `virtual_application_patterns`
- `virtual_images`
- `virtual_system_patterns`

Handling existing artifact directories

If a directory already exists, the migration tool logs an error message and this type of artifact is skipped and is not exported. The following example output shows the result when an export operation is attempted on script packages and add-ons, but the `script_packages` directory is already created:

Figure 7. Example output when script_packages directory is already created during an export operation.



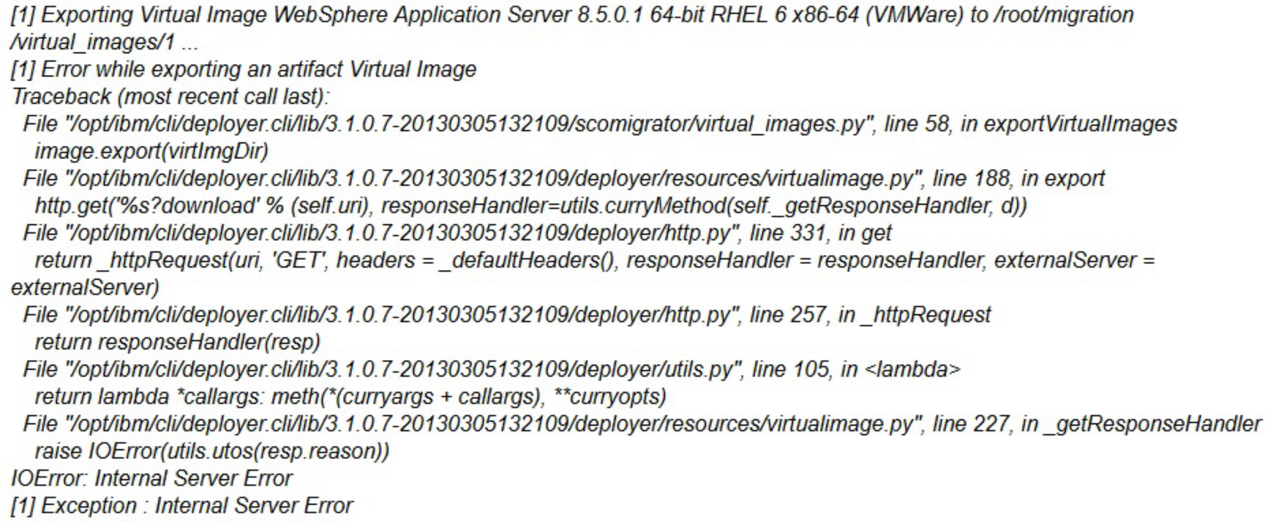
```
Exporting Script Packages to directory: /root/migration/script_packages ...

Destination /root/migration/script_packages exists. Please remove the directory before export.
 Skipping
-----------------

Exporting Add-Ons to directory: /root/migration/add_ons

[1] Exporting Add-On: Default add disk ...
[1] Add-On exported

[2] Exporting Add-On: Default raw disk ...
[2] Add-On exported
```

Handling an error during export

The export operation might fail if there are problems with certain artifacts. If the export of a particular artifact does not complete successfully, the error is logged in the `export_<timestamp>.log` file and the export operation continues with the next artifact.

The following example log output shows the resulting error message when an export operation fails for an OVA image file that has a corrupt file in the package:

Figure 8. Example log output for a failed export operation on an OVA image file.



```
[1] Exporting Virtual Image WebSphere Application Server 8.5.0.1 64-bit RHEL 6 x86-64 (VMWare) to /root/migration
/virtual_images/1 ...
[1] Error while exporting an artifact Virtual Image
 Traceback (most recent call last):
  File "/opt/ibm/cli/deployer.cli/lib/3.1.0.7-20130305132109/scomigrator/virtual_images.py", line 58, in exportVirtualImages
    image.export(virtImgDir)
  File "/opt/ibm/cli/deployer.cli/lib/3.1.0.7-20130305132109/deployer/resources/virtualimage.py", line 188, in export
    http.get('%s?download' % (self.uri), responseHandler=utils.curryMethod(self._getResponseHandler, d))
  File "/opt/ibm/cli/deployer.cli/lib/3.1.0.7-20130305132109/deployer/http.py", line 331, in get
    return _httpRequest(uri, 'GET', headers = _defaultHeaders(), responseHandler = responseHandler, externalServer =
externalServer)
  File "/opt/ibm/cli/deployer.cli/lib/3.1.0.7-20130305132109/deployer/http.py", line 257, in _httpRequest
    return responseHandler(resp)
  File "/opt/ibm/cli/deployer.cli/lib/3.1.0.7-20130305132109/deployer/utils.py", line 105, in <lambda>
    return lambda *callargs: meth(*(curryargs + callargs), **curryopts)
  File "/opt/ibm/cli/deployer.cli/lib/3.1.0.7-20130305132109/deployer/resources/virtualimage.py", line 227, in _getResponseHandler
    raise IOError(utils.utos(resp.reason))
 IOError: Internal Server Error
 [1] Exception : Internal Server Error
```

Examine the IBM Workload Deployer log files to diagnose the root cause of the image export failure.

4 Preparing for the import process

After exporting the various artifacts from IBM Workload Deployer. there are a number of tasks you need to complete and considerations you need to plan for before you can run the migration tool to import artifacts into your IBM SmartCloud Orchestrator environment.

Before you begin importing artifacts into the IBM SmartCloud Orchestrator environment, be sure to read the following topics and complete any necessary manual tasks, to ensure a successful import process.

- Ensure that the list of requirements is satisfied.
- Be aware that the order of importing some artifacts is important.
- Update pattern type information if needed.
- After the import operation completes, note that there are several additional manual steps to complete in your IBM SmartCloud Orchestrator environment.

Requirements for importing artifacts

Ensure that the following requirements are met:

- If you are migrating using **Environment 3** (two migration systems), you should have already installed the migration tool on both systems, and configured the migration tool on *Migration System 2* to import artifacts into IBM SmartCloud Orchestrator.
- All artifacts to be imported must be locally available to the migration tool in the location specified by the **MIGRATION_DEST** parameter in the migration property file (default file name `migration_data.txt`).
- Ensure that the system performing the migration can connect to IBM SmartCloud Orchestrator over HTTP protocol. This is required in order for the IBM Workload Deployer command line interface to import artifacts.
- Ensure that the parameters in the `SCO data` section of the migration tool property file are specified correctly to connect to *Central Server 2* and *Central Server 3* in IBM SmartCloud Orchestrator.

Considerations for the order of importing artifacts

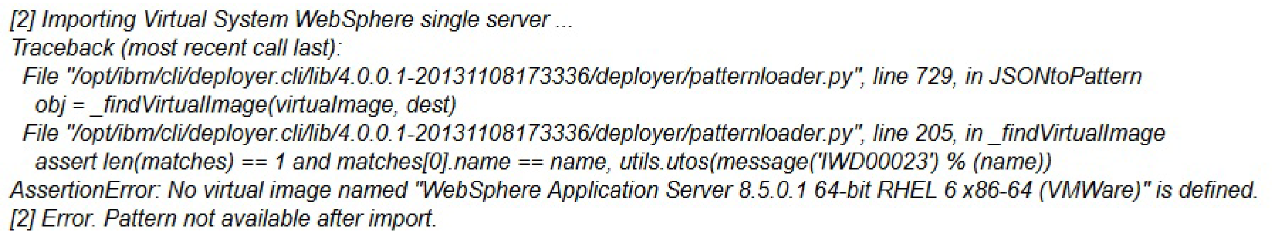
When you are importing several types of artifacts, be aware that the order of importing some artifacts is important. To minimize problems in this area, you should consider specifying the **-a all** option when performing the import operation, to let the migration tool control the order of importing artifacts correctly.

You can import artifacts one type at a time, by specifying only one type in the artifacts list, but if you do, keep the following limitations in mind:

- You should import users and user groups first before importing other types of artifacts. Without users and user groups, access control lists for other artifacts cannot be recreated.
- Be sure to import virtual images before importing virtual system patterns. If the virtual images are not available, the import of virtual system patterns fails.
- Be sure to import pattern types before importing virtual application patterns. If the pattern types are not available, the import of virtual application patterns fails.

The following example log output shows the resulting error message when an import operation fails on a virtual system pattern because it refers to a virtual image that has not yet been imported into IBM SmartCloud Orchestrator:

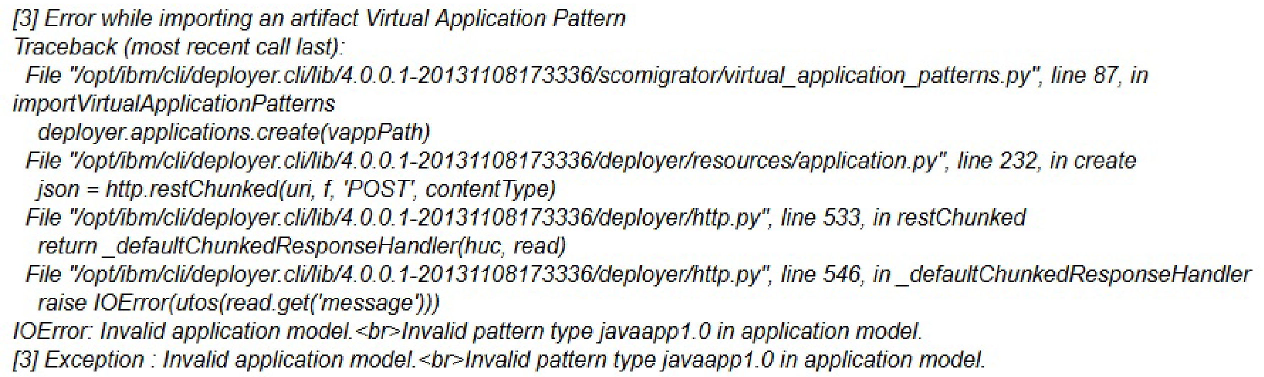
Figure 9. Example log output for a failed import operation on a virtual system pattern that refers to an image that is not already imported.



```
[2] Importing Virtual System WebSphere single server ...
Traceback (most recent call last):
  File "/opt/ibm/cli/deployer.cli/lib/4.0.0.1-20131108173336/deployer/patternloader.py", line 729, in JSONtoPattern
    obj = _findVirtualImage(virtualImage, dest)
  File "/opt/ibm/cli/deployer.cli/lib/4.0.0.1-20131108173336/deployer/patternloader.py", line 205, in _findVirtualImage
    assert len(matches) == 1 and matches[0].name == name, utils.utos(message('IWD00023') % (name))
AssertionError: No virtual image named "WebSphere Application Server 8.5.0.1 64-bit RHEL 6 x86-64 (VMWare)" is defined.
[2] Error. Pattern not available after import.
```

The following example log output shows the resulting error message when an import operation fails on a virtual application pattern because it refers to a pattern type that has not yet been imported into IBM SmartCloud Orchestrator:

Figure 10. Example log output for a failed import operation on a virtual application pattern that refers to a pattern type that is not already imported.



```
[3] Error while importing an artifact Virtual Application Pattern
Traceback (most recent call last):
  File "/opt/ibm/cli/deployer.cli/lib/4.0.0.1-20131108173336/scomigrator/virtual_application_patterns.py", line 87, in
importVirtualApplicationPatterns
    deployer.applications.create(vappPath)
  File "/opt/ibm/cli/deployer.cli/lib/4.0.0.1-20131108173336/deployer/resources/application.py", line 232, in create
    json = http.restChunked(uri, f, 'POST', contentType)
  File "/opt/ibm/cli/deployer.cli/lib/4.0.0.1-20131108173336/deployer/http.py", line 533, in restChunked
    return _defaultChunkedResponseHandler(huc, read)
  File "/opt/ibm/cli/deployer.cli/lib/4.0.0.1-20131108173336/deployer/http.py", line 546, in _defaultChunkedResponseHandler
    raise IOError(utos(read.get('message')))
IOError: Invalid application model.<br>Invalid pattern type javaapp1.0 in application model.
[3] Exception : Invalid application model.<br>Invalid pattern type javaapp1.0 in application model.
```

Updating pattern type information

If you are importing pattern types or virtual application patterns, manually update the `pattern.json` file for each pattern type to specify the location of the corresponding TGZ file to be imported. For more information on this procedure, see chapter 5, “Updating pattern type information,” on page 19.

Additional tasks after importing artifacts

After you perform the import operation, there are several additional manual steps you need to complete before you have a fully functioning IBM SmartCloud Orchestrator environment.

- Restore passwords for all migrated user accounts that were not stored with LDAP in IBM Workload Deployer.
- Restore the owner information for imported artifacts.
- Update access control lists for all artifacts.
- Add hypervisors and create network configurations as needed in IBM SmartCloud Orchestrator, by using the OpenStack console.
- Ensure that the checkout process is performed on all imported virtual images, to make them available for use in IBM SmartCloud Orchestrator.

For more information on these tasks, see chapter 7, “Additional tasks after importing artifacts,” on page 25.

5 Updating pattern type information

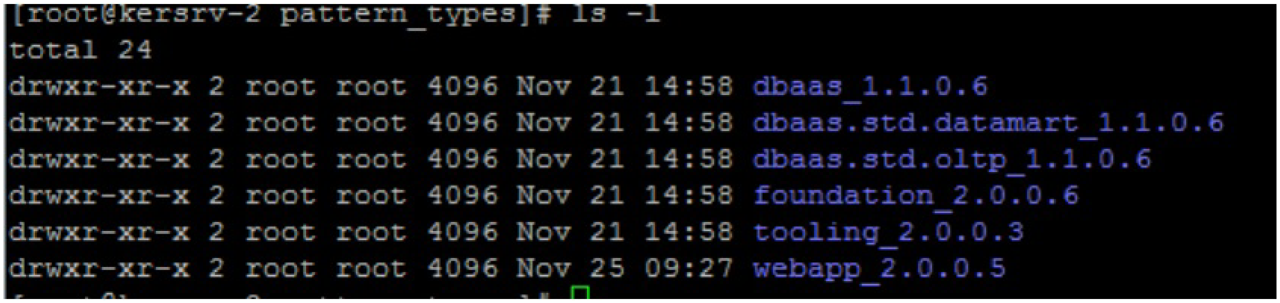
If you are importing pattern types or virtual application patterns, manually update the `pattern.json` file for each pattern type to specify the location of the corresponding TGZ file to be imported.

The migration tool is not able to export binaries of pattern types because this operation is not supported by the IBM Workload Deployer command line interface. When you specify to export pattern type artifacts, you are really only exporting general information about the pattern types. To include pattern types in your import to IBM SmartCloud Orchestrator, you must edit the `pattern.json` file for each pattern type and specify the location of the compressed file containing the actual pattern type data.

To update your pattern type information for importing into IBM SmartCloud Orchestrator, complete the following steps for each pattern type on the system where the exported artifact data is stored:

1. Navigate to the target location where the artifact data is located. This location is specified in the **MIGRATION_DEST** parameter in the migration property file (default file name `migration_data.txt`. The default target location is `/root/migration`.
2. Locate and open the `pattern_types` directory.
3. List the contents of the `pattern_types` directory to display the subdirectory for each pattern type that was exported from IBM Workload Deployer, similar to the following example:

Figure 11. Contents of the pattern_types directory.



```
[root@kersrv-2 pattern_types]# ls -l
total 24
drwxr-xr-x 2 root root 4096 Nov 21 14:58 dbaas_1.1.0.6
drwxr-xr-x 2 root root 4096 Nov 21 14:58 dbaas.std.datamart_1.1.0.6
drwxr-xr-x 2 root root 4096 Nov 21 14:58 dbaas.std.oltp_1.1.0.6
drwxr-xr-x 2 root root 4096 Nov 21 14:58 foundation_2.0.0.6
drwxr-xr-x 2 root root 4096 Nov 21 14:58 tooling_2.0.0.3
drwxr-xr-x 2 root root 4096 Nov 25 09:27 webapp_2.0.0.5
```

4. Change directory to the pattern type to be imported.
5. Open the `pattern.json` file, containing the description of the pattern type.

   The contents look similar to the following example:

   ```
   [
    {
      "file": ""
      "name": "Web Application Pattern Type",
      "shortname": "webapp",
      "status": "avail",
      "version": "2.0.0.5"
    }
   ]
   ```

   When the pattern type artifacts are exported, the `file` attribute is left empty. You need to edit this field and provide the location where the pattern type package is located.
6. Edit the `file` attribute and specify the name and location of the pattern type package file. Your entry should be similar to the following example:

   ```
   [
    {
      "file": "/root/migration/pattern_types/webapp_2.0.0.5/webapp-2.0.0.5.tgz"
      "name": "Web Application Pattern Type",
      "shortname": "webapp",
   ```

```
   "status": "avail",
   "version": "2.0.0.5"
 }
]
```

If the package file is at a remote location, ensure that the IBM SmartCloud Orchestrator virtual machine (*Central Server 3*) is able to connect to the specified location, because the import process is run directly by the server, not by the migration system.

Repeat this procedure for each pattern type to be imported.

Note: Be aware that if you attempt to import a virtual application pattern but the associated pattern type is not available, the import operation will fail.

6 Performing the import process

After exporting the various artifacts from IBM Workload Deployer and completing necessary manual tasks, you can run the migration tool to import artifacts into your IBM SmartCloud Orchestrator environment.

Import command syntax

To run the migration tool for importing artifacts into IBM SmartCloud Orchestrator, use the `migration.sh` script, located in the `migration_tool` directory.

The general syntax of this script command and options is as follows:

```
./migration.sh -m|--mode <mode> [-a|--artifacts <artifact list> -f|--file <property file>]
```

You can specify the following options:

-m|--mode
The mode of migration. Valid values are *export* or *import*.

-a|--artifacts
The list of artifacts to be migrated. This is a list of artifact types, separated by commas, with no blank spaces in between. Valid values include:

- **all**: migrate all artifacts
- **user**: migrate users and user groups
- **script**: migrate script packages
- **addon**: migrate add-ons
- **virtimg**: migrate virtual images
- **vsys**: migrate virtual system patterns
- **vapp**: migrate virtual application patterns
- **ptype**: migrate pattern types

If you do not specify this option, the default value `all` is used.

You can optionally surround the list of options with single or double quotation marks, such as 'addon,vsys,script' or "vapp,ptype" and so on. Do not add blank spaces inside quotation marks.

Valid artifact list examples:

```
-a all
-a user,vsys,script,vapp,virtimg
-artifacts "vapp,vsys,addon"
-a 'ptype,vapp,user'
```

-f|--file
The name of the property file containing the migration process attributes. If you do not specify this option, the default value `migration_data.txt` is used.

Examples:

- `./migration.sh -m import`

  This command performs an import operation on all available artifacts, and uses the default property file, `migration_data.txt`.
- `./migration.sh --mode import -a all`

  This command performs an import operation on all available artifacts, and uses the default property file, `migration_data.txt`.

- `./migration.sh -m import -artifacts vapp`

  This command performs an import operation only on virtual application pattern artifacts, and uses the default property file, `migration_data.txt`.
- `./migration.sh -m import -a "script,addon,vsys" -f migration_special.txt`

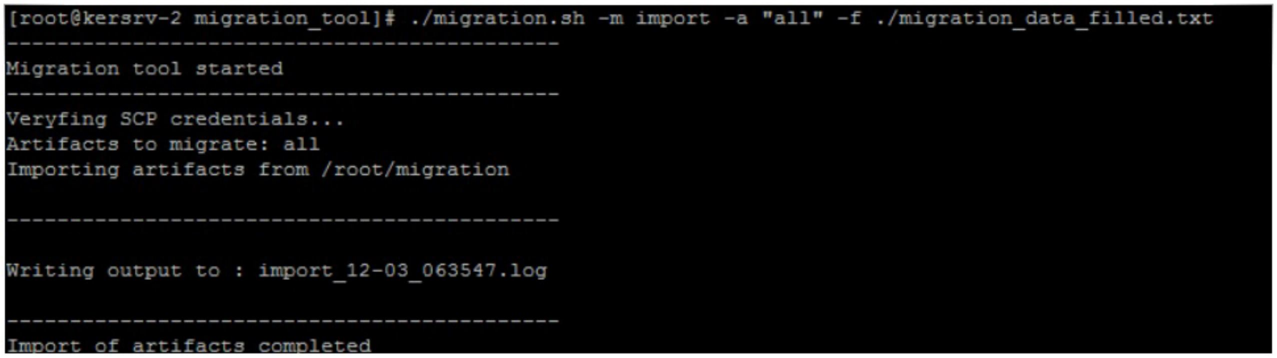
  This command performs an import operation only on script packages, add-ons, and virtual system pattern artifacts, and uses the property file, `migration_special.txt`.

Performing the import operation

The import migration operation imports all specified artifacts from the location specified in the **MIGRATION_DEST** parameter defined in the property file.

When you run the migration script to perform an import operation, the result is similar to the following example:

Figure 12. Example output when performing an import operation.



```
[root@kersrv-2 migration_tool]# ./migration.sh -m import -a "all" -f ./migration_data_filled.txt
-------------------------------------------
Migration tool started
-------------------------------------------
Veryfing SCP credentials...
Artifacts to migrate: all
Importing artifacts from /root/migration

-------------------------------------------

Writing output to : import_12-03_063547.log

-------------------------------------------
Import of artifacts completed
```

The import operation performs the following tasks:

- The connection credentials with the IBM SmartCloud Orchestrator virtual machine where IBM Workload Deployer is running (*Central Server 3*) are verified.
- Based on the specified list of artifacts to export, the migration tool imports the artifacts from the target location into IBM SmartCloud Orchestrator. Output log information is written to a local log file. Check this log to verify that all artifacts were imported successfully.

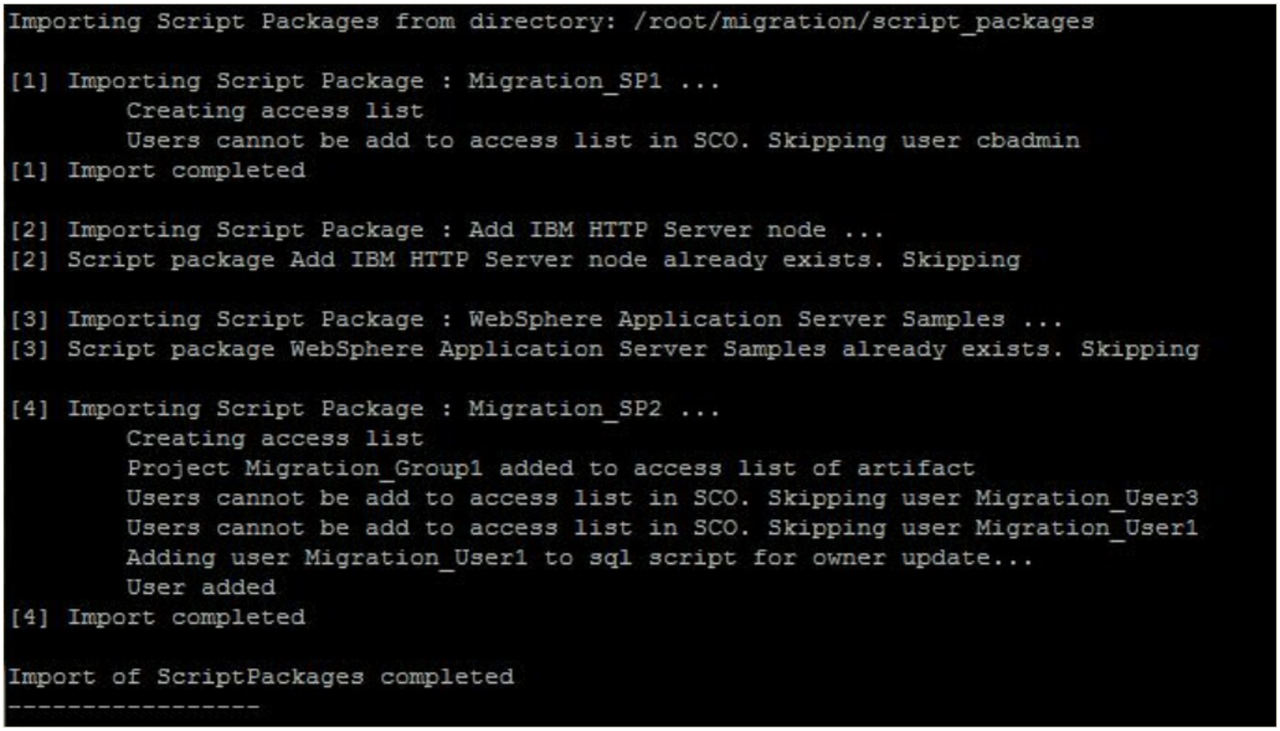
Handling duplicate artifacts

As you import artifacts into the IBM SmartCloud Orchestrator environment, you might encounter a situation where some artifacts already exist. This situation might occur when you are importing into an existing environment (not newly installed), or if the import process is run multiple times.

If the import operation discovers an artifact that already exists in the IBM SmartCloud Orchestrator environment, it is skipped from being imported, and the import operation continues. A message is written to the log file, so you can verify that all artifacts were imported or be aware if duplicates were encountered.

The following example output shows the partial log results when an import operation is attempted on script packages, but several script packages already exist in the IBM SmartCloud Orchestrator environment:

Figure 13. Example log output when script packages already exist and are skipped from being imported.



```
Importing Script Packages from directory: /root/migration/script_packages

[1] Importing Script Package : Migration_SP1 ...
        Creating access list
        Users cannot be add to access list in SCO. Skipping user cbadmin
[1] Import completed

[2] Importing Script Package : Add IBM HTTP Server node ...
[2] Script package Add IBM HTTP Server node already exists. Skipping

[3] Importing Script Package : WebSphere Application Server Samples ...
[3] Script package WebSphere Application Server Samples already exists. Skipping

[4] Importing Script Package : Migration_SP2 ...
        Creating access list
        Project Migration_Group1 added to access list of artifact
        Users cannot be add to access list in SCO. Skipping user Migration_User3
        Users cannot be add to access list in SCO. Skipping user Migration_User1
        Adding user Migration_User1 to sql script for owner update...
        User added
[4] Import completed

Import of ScriptPackages completed
-----------------
```

In this example, the import operation is attempting to import four script packages, but two of the script packages (the second and third) already exist. These script packages are skipped, and the import operation continues.

If you want to import the artifact anyway, you need to first change the name of the artifact in the corresponding JSON object file. For example, to change the name of the *Migration SP2* script package from the previous example, navigate to the `/root/migration/script_packages` directory, and edit the `script.json` file for the *Migration SP2* script package, which might look similar to the following example:

```
[
  {
    "acl": [
       {
          "name": "Migration_User1",
          "rights": 4,
          "type": "user"
       },
       {
          "name": "Migration_User3",
          "rights": 4,
          "type": "user"
       }
    ],
    "command": "sh \/tmp\/itm_test\/show_parameters",
    "commandargs": "",
    "environment": {
       "New_param": "6",
       "Test_Param2": ""
    },
    "execmode": 0,
    "file": "test2(1).zip",
    "location": "\/tmp\/itm_test",
    "log": "\/tmp\/itm_test",
"name": "Migration_SP2",
"owner": "Migration_User1",
"read_only": false,
"timeout": 120000
```

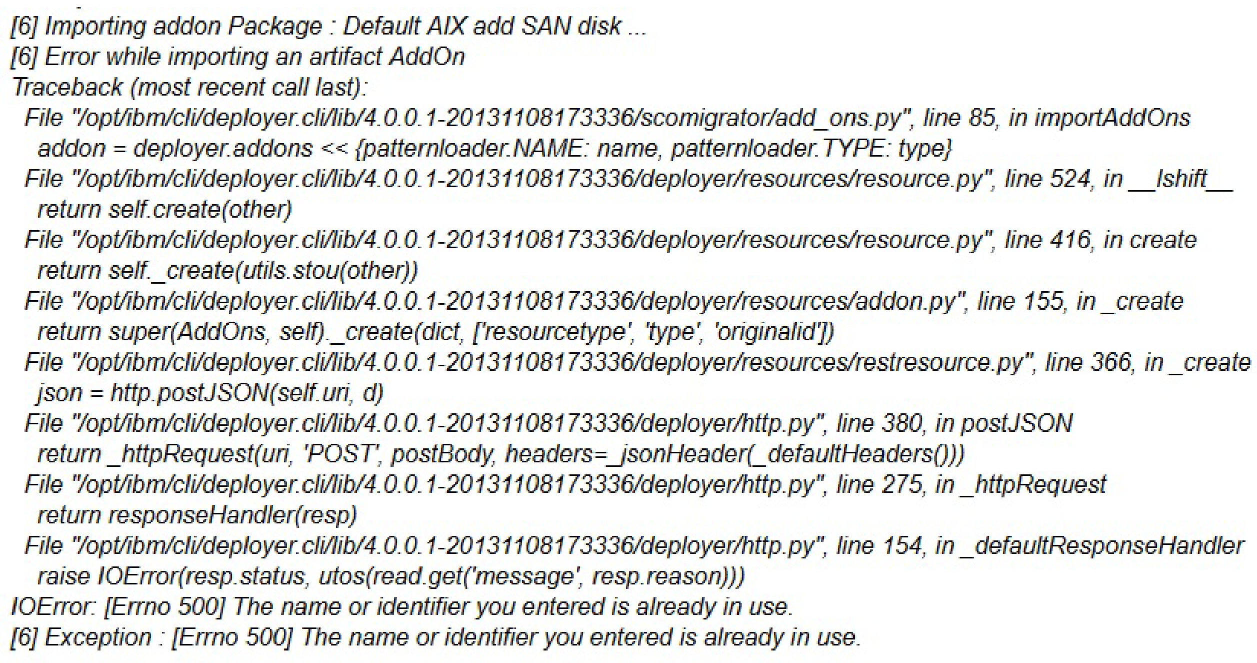
Change the contents of the name attribute to something unique, save your change and attempt the import operation again.

Limitation when importing Default AIX add SAN disk add-on

Due to a known limitation, the **Default AIX add SAN disk** add-on that is provided with IBM SmartCloud Orchestrator 2.3 is corrupted and cannot be recognized by the migration tool during an import operation. The add-on exists in the database, but when the migration tool checks for existing add-ons it does not detect this add-on.

As a result, the migration tool attempts to import the add-on, and the import operation fails. The following example log output shows the resulting error message when an import operation fails on the **Default AIX add SAN disk** add-on, claiming that the name of the artifact already exists in IBM SmartCloud Orchestrator:

Figure 14. Example log output for a failed import operation on a duplicate Default AIX® add SAN disk add-on.



```
[6] Importing addon Package : Default AIX add SAN disk ...
[6] Error while importing an artifact AddOn
Traceback (most recent call last):
  File "/opt/ibm/cli/deployer.cli/lib/4.0.0.1-20131108173336/scomigrator/add_ons.py", line 85, in importAddOns
    addon = deployer.addons << {patternloader.NAME: name, patternloader.TYPE: type}
  File "/opt/ibm/cli/deployer.cli/lib/4.0.0.1-20131108173336/deployer/resources/resource.py", line 524, in __lshift__
    return self.create(other)
  File "/opt/ibm/cli/deployer.cli/lib/4.0.0.1-20131108173336/deployer/resources/resource.py", line 416, in create
    return self._create(utils.stou(other))
  File "/opt/ibm/cli/deployer.cli/lib/4.0.0.1-20131108173336/deployer/resources/addon.py", line 155, in _create
    return super(AddOns, self)._create(dict, ['resourcetype', 'type', 'originalid'])
  File "/opt/ibm/cli/deployer.cli/lib/4.0.0.1-20131108173336/deployer/resources/restresource.py", line 366, in _create
    json = http.postJSON(self.uri, d)
  File "/opt/ibm/cli/deployer.cli/lib/4.0.0.1-20131108173336/deployer/http.py", line 380, in postJSON
    return _httpRequest(uri, 'POST', postBody, headers=_jsonHeader(_defaultHeaders()))
  File "/opt/ibm/cli/deployer.cli/lib/4.0.0.1-20131108173336/deployer/http.py", line 275, in _httpRequest
    return responseHandler(resp)
  File "/opt/ibm/cli/deployer.cli/lib/4.0.0.1-20131108173336/deployer/http.py", line 154, in _defaultResponseHandler
    raise IOError(resp.status, utos(read.get('message', resp.reason)))
IOError: [Errno 500] The name or identifier you entered is already in use.
[6] Exception : [Errno 500] The name or identifier you entered is already in use.
```

If you want to import the add-on anyway, you need to locate the add-on in the /root/migration/add_ons directory and edit the corresponding addon.json object file, changing the name of the add-on to a unique name before performing the import operation.

7 Additional tasks after importing artifacts

After you perform the import operation, there are several additional manual steps you need to complete before you have a fully functioning IBM SmartCloud Orchestrator environment.

See your IBM SmartCloud Orchestrator administrator for assistance in completing the following tasks:

- Restore passwords for all migrated user accounts that were not stored with LDAP in IBM Workload Deployer.
- Restore the owner information for imported artifacts.
- Update access control lists for all artifacts.
- Add hypervisors and create network configurations as needed in IBM SmartCloud Orchestrator, by using the OpenStack console.
- Ensure that the checkout process is performed on all imported virtual images, to make them available for use in IBM SmartCloud Orchestrator.

Restoring user account passwords

If user accounts that were exported from the IBM Workload Deployer appliance were not stored with LDAP, the migration process is unable to retrieve and migrate passwords for these user accounts. As a result, during the import process, passwords for these user accounts are set to a default value, *change_me*.

After migration, you will need to change these passwords manually.

Restoring ownership of artifacts

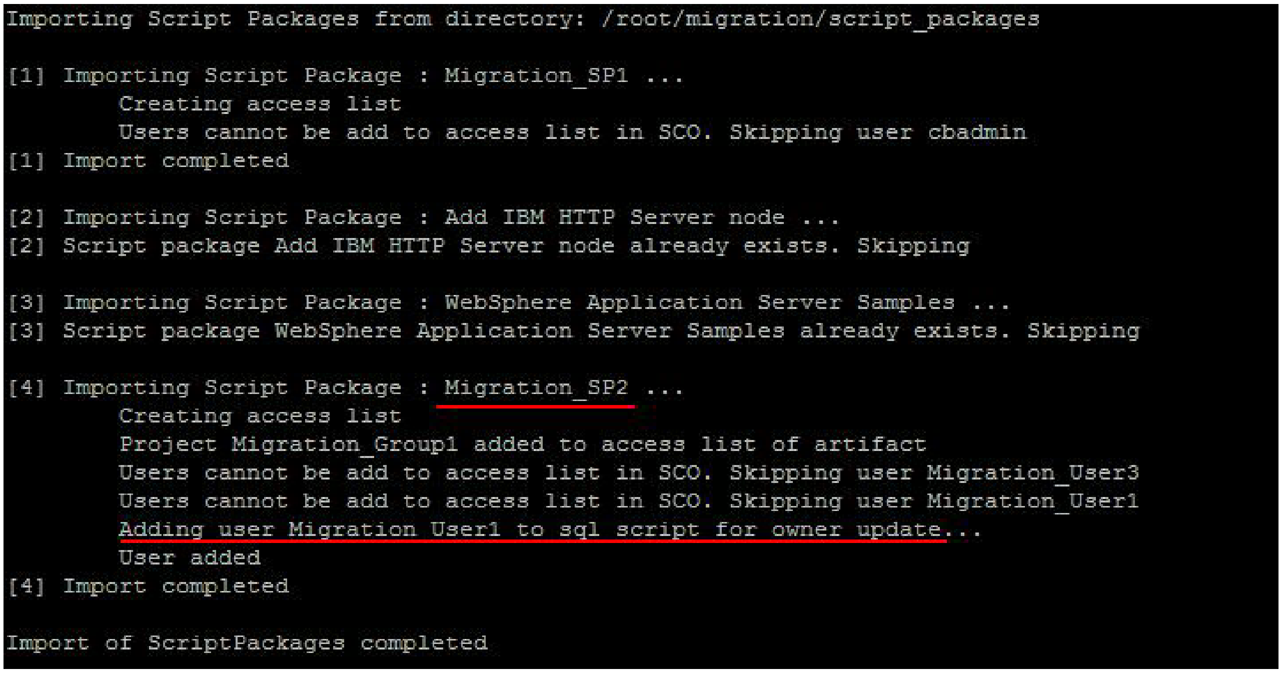
Because of differences between IBM Workload Deployer and IBM SmartCloud Orchestrator, there is no way to preserve the ownership of artifacts when they are migrated. By default all artifacts are imported and assigned to the *admin* owner, which is the default administrator of IBM SmartCloud Orchestrator.

To restore ownership of these artifacts to the original user name, the import migration process includes the creation of a special SQL file that you can copy to the *Central Server 1* virtual machine in IBM SmartCloud Orchestrator, where DB2 is running. This SQL file is stored in the same directory as the exported artifacts (default `/root/migration`), and is named in the following format: `migration_`*`mm-dd_nnnnn`*`.sql`, where *mm-dd_nnnnn* is a timestamp.

In addition, the migration tool provides a special script to run this SQL file, located in the `migration_tool/helpers` directory, named `updateIwdDatabase.sh`. You must also copy this script to the same *Central Server 1* virtual machine, and run it to reassign the original owner names to the artifacts.

The following example shows a partial log output after importing several script package artifacts. Of interest is the fourth script package named *Migration SP2*. During the creation of the access list, the *Migration_Group1* project is added. The process detects two users, *Migration_User3* and *Migration_User1*, but because we cannot add users directly to the access list in IBM SmartCloud Orchestrator, those user names are skipped. The owner of the artifact, *Migration_User1* (identified as the owner in the `script.json` file) is detected, however, and is added to the SQL file for later processing.

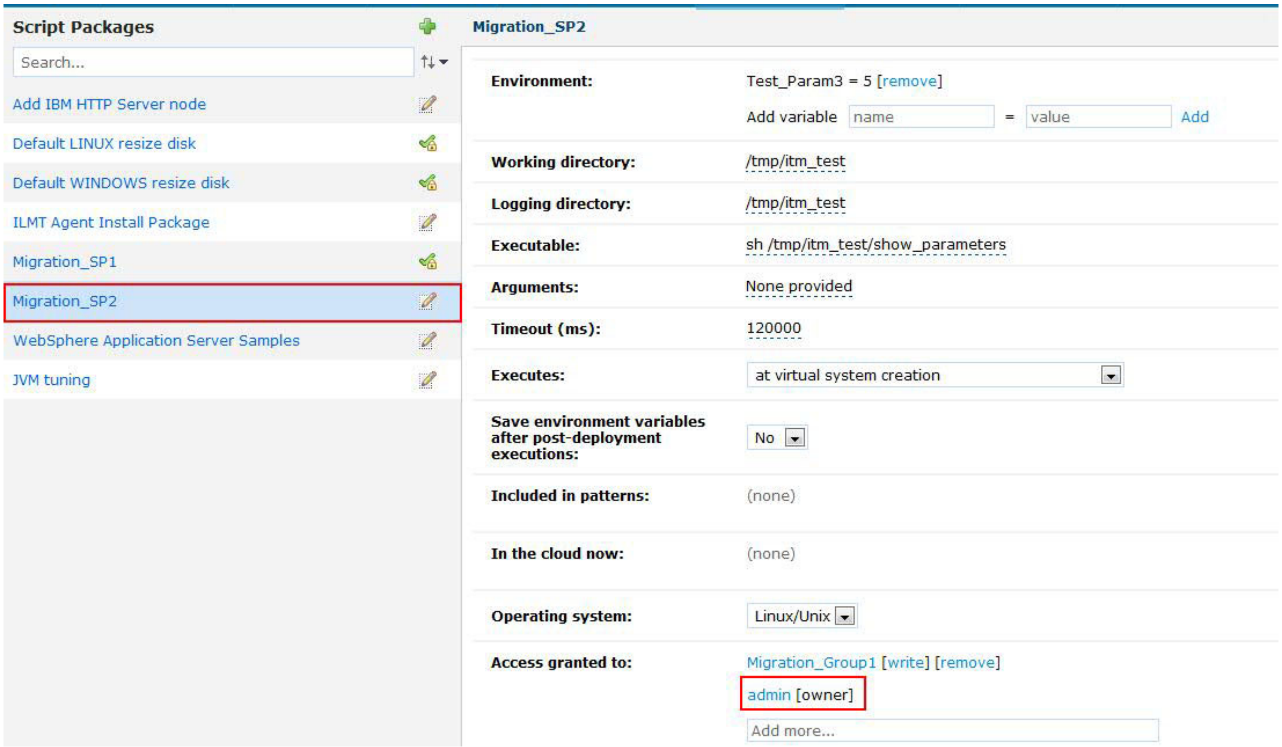
Figure 15. Example log output showing the import of the Migration_SP2 script package and assigning the artifact owner to the SQL file to include in the owner restore process.



```
Importing Script Packages from directory: /root/migration/script_packages

[1] Importing Script Package : Migration_SP1 ...
        Creating access list
        Users cannot be add to access list in SCO. Skipping user cbadmin
[1] Import completed

[2] Importing Script Package : Add IBM HTTP Server node ...
[2] Script package Add IBM HTTP Server node already exists. Skipping

[3] Importing Script Package : WebSphere Application Server Samples ...
[3] Script package WebSphere Application Server Samples already exists. Skipping

[4] Importing Script Package : Migration_SP2 ...
        Creating access list
        Project Migration_Group1 added to access list of artifact
        Users cannot be add to access list in SCO. Skipping user Migration_User3
        Users cannot be add to access list in SCO. Skipping user Migration_User1
        Adding user Migration_User1 to sql script for owner update...
        User added
[4] Import completed

Import of ScriptPackages completed
```

After the import process completes, when you view the details of this script package in the IBM SmartCloud Orchestrator script packages catalog, you see that the owner has been set to the default *admin* user:

Figure 16. IBM SmartCloud Orchestrator script package catalog details for the Migration SP2 script package



Script Packages

Search...

- Add IBM HTTP Server node
- Default LINUX resize disk
- Default WINDOWS resize disk
- ILMT Agent Install Package
- Migration_SP1
- Migration_SP2
- WebSphere Application Server Samples
- JVM tuning

Migration_SP2

Environment: Test_Param3 = 5 [remove]

Add variable name = value Add

Working directory: /tmp/itm_test

Logging directory: /tmp/itm_test

Executable: sh /tmp/itm_test/show_parameters

Arguments: None provided

Timeout (ms): 120000

Executes: at virtual system creation

Save environment variables after post-deployment executions: No

Included in patterns: (none)

In the cloud now: (none)

Operating system: Linux/Unix

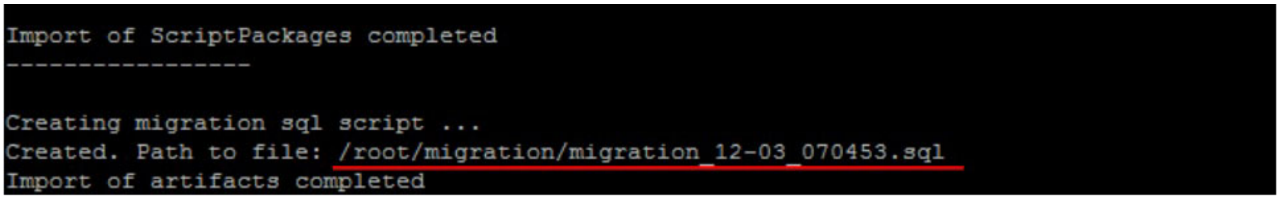
Access granted to: Migration_Group1 [write] [remove]

admin [owner]

Add more...

Looking back at the log output, you can see that a special SQL file has been generated:

Figure 17. The log file shows the name of the SQL file created during the import process.



```
Import of ScriptPackages completed
-----------------

Creating migration sql script ...
Created. Path to file: /root/migration/migration_12-03_070453.sql
Import of artifacts completed
```

In this example, you would complete the following steps:

1. Copy the `/root/migration/migration_12-03_070453.sql` file to the `/db2inst1` directory on the *Central Server 1* virtual machine.
2. Copy the `/migration_tool/helpers/updateIwdDatabase.sh` file to the `/db2inst1` directory on the *Central Server 1* virtual machine.
3. Log in to the *Central Server 1* virtual machine.

   ```
   ssh <Central Server 1 hostname>
   ```
4. Add execution rights to the `updateIwdDatabase.sh` script:

   ```
   chmod +x ./updateIwdDatabase.sh
   ```
5. Change the user to a database administrator, typically `db2inst1`.

   ```
   su - db2inst1
   ```
6. Run the `updateIwdDatabase.sh` script:

   ```
   ./updateIwdDatabase.sh -f migration_12-03_070453.sql
   ```

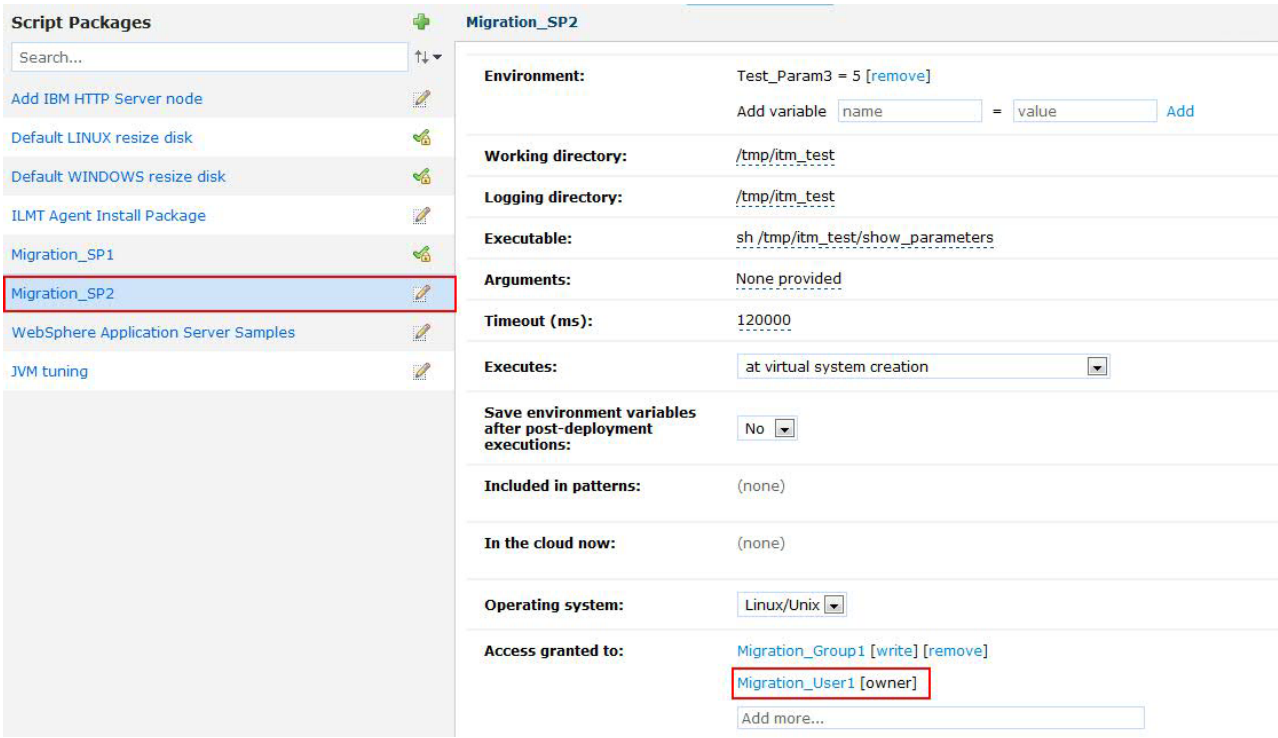
   The system response is similar to the following example:

   ```
   Updating IWD database. Using migration_12-03_070453.sql
   Logging to file UpdateIwdDatabase-1386073896.log
   Completed successfully
   ```

   The entire SQL file is treated as a database transaction, so if one SQL statement fails, all changes are rolled back.

Now return to the Script Packages catalog details for the *Migration_SP2* script package, and verify that the **Access granted to** field is updated to show the owner name as *Migration_User1*.

Figure 18. The script packages catalog details are updated to show the original owner name for the imported artifact.



Script Packages
Search...
Add IBM HTTP Server node
Default LINUX resize disk
Default WINDOWS resize disk
ILMT Agent Install Package
Migration_SP1
Migration_SP2
WebSphere Application Server Samples
JVM tuning

Migration_SP2

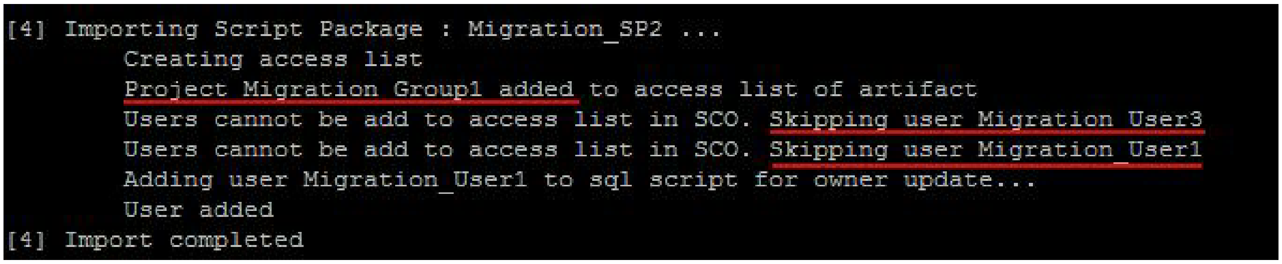
Environment:	Test_Param3 = 5 [remove] Add variable name = value Add
Working directory:	/tmp/itm_test
Logging directory:	/tmp/itm_test
Executable:	sh /tmp/itm_test/show_parameters
Arguments:	None provided
Timeout (ms):	120000
Executes:	at virtual system creation
Save environment variables after post-deployment executions:	No
Included in patterns:	(none)
In the cloud now:	(none)
Operating system:	Linux/Unix
Access granted to:	Migration_Group1 [write] [remove] Migration_User1 [owner] Add more...

Updating access control lists for artifacts

Access control lists in IBM SmartCloud Orchestrator are different from those in IBM Workload Deployer Version 3.1.0.7. In IBM SmartCloud Orchestrator, access control lists are defined on a *project* level. This means that you cannot grant specific rights directly for a single user, only to a whole project. Be aware that the access list for artifacts after migration does not include single user access. The IBM SmartCloud Orchestrator administrator will need to manage these user accounts and access control as needed.

In the previous example, during the import of the *Migration_SP2* script package, the project named *Migration_Group1* was added to the access control list for the script package. Two users that previously had access to the script package in IBM Workload Deployer, are also detected but are skipped because of the limitation on adding users directly to access control lists.

Figure 19. Example log output showing the import of the Migration_SP2 script package and the addition of the Migration_Group1 project to the access list for the script package.



```
[4] Importing Script Package : Migration_SP2 ...
        Creating access list
        Project Migration_Group1 added to access list of artifact
        Users cannot be add to access list in SCO. Skipping user Migration_User3
        Users cannot be add to access list in SCO. Skipping user Migration_User1
        Adding user Migration_User1 to sql script for owner update...
        User added
[4] Import completed
```

After the import process completes, the Script Packages catalog details for the *Migration_SP2* script package shows in the **Access granted to** field that the *Migration_Group1* project is granted project level access.

Adding hypervisors and create network configurations

In order to get a fully working IBM SmartCloud Orchestrator environment, you must add a hypervisor and create networks. You can perform network configuration only by using the OpenStack console. This operation should be performed manually by the IBM SmartCloud Orchestrator administrator, because the network configuration can be complicated and difficult. OpenStack is installed on Region Servers.

To configure OpenStack on the Region Server with VMware Vcenter, the administrator must complete several steps. Detailed documentation can be found at the following web site:

`http://pic.dhe.ibm.com/infocenter/tivihelp/v48r1/index.jsp/com.ibm.sco.doc_2.2/t_integrate_vcenter.html`

The migration tool provides a script that you can use for basic configuration with VMWare Vcenter. The script is named `configureOpenstack.sh` and is located in the `migration_tool/helpers` directory. Before running the script, be sure to read the IBM SmartCloud Orchestrator documentation regarding integration with VMWare Vcenter.

To add a hypervisor with this script, you need to define several parameters in the `configureOpenstack.txt` property file:

```
###########################################
# Parameters for Openstack configuration
# with existing VMWare VCenter
###########################################
# VCENTER_IP=<VCenter ip required for Openstack config>
# VCENTER_USER=<VCenter administrator user>
# VCENTER_PASSWORD=<VCenter administrator password>
# VCENTER_NETWORK_NAME=<Name of existing network on VCenter>
# VMWARE_CLOUD_NAME="VMWareCloud"
##########################################
```

The parameters are defined as follows:

VCENTER_IP
The IP address of the VMware Vcenter being used

VCENTER_USER
The user name of the Vcenter administrator.

VCENTER_PASSWORD
The password for the Vcenter administrator. If this is not specified in this property file, you are prompted to enter it during execution.

VCENTER_NETWORK_NAME
The network name that exists in the Vcenter. Take care to ensure this value is set correctly, or the configuration will fail.

VMWARE_CLOUD_NAME
The name for the new network being created. the default value is *VMWareCloud*, but you can change it to another value as needed.

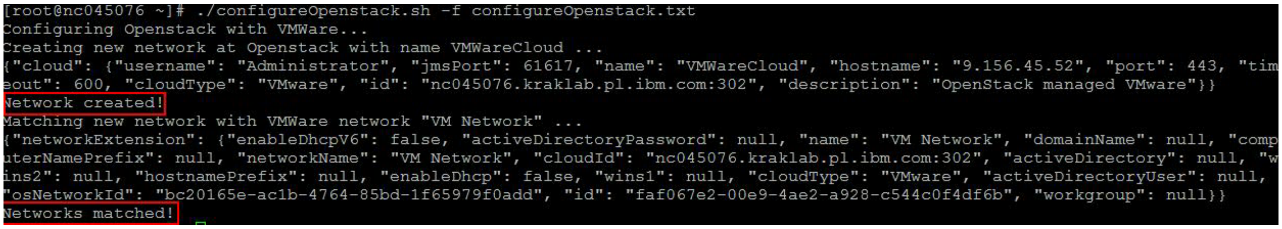
To configure on OpenStack, complete the following steps:

1. Copy `configureOpenstack.sh` and `configureOpenstack.txt` to the *Region Server* with OpenStack.
2. Set properties in `configureOpenstack.txt`.
3. Add execute rights for `configureOpenstack.sh`
4. Run `configureOpenstack.sh`:

   ```
   ./configureOpenstack.sh [-f|--file <property file>]
   ```

   If the file name is not specified, the default property file, `configureOpenstack.txt` is assumed to be used.

Figure 20. Example run of the `configureOpenstack.sh` *script.*



```
[root@nc045076 ~]# ./configureOpenstack.sh -f configureOpenstack.txt
Configuring Openstack with VMWare...
Creating new network at Openstack with name VMWareCloud ...
{"cloud": {"username": "Administrator", "jmsPort": 61617, "name": "VMWareCloud", "hostname": "9.156.45.52", "port": 443, "tim
eout": 600, "cloudType": "VMware", "id": "nc045076.kraklab.pl.ibm.com:302", "description": "OpenStack managed VMware"}}
Network created!
Matching new network with VMWare network "VM Network" ...
{"networkExtension": {"enableDhcpV6": false, "activeDirectoryPassword": null, "name": "VM Network", "domainName": null, "comp
uterNamePrefix": null, "networkName": "VM Network", "cloudId": "nc045076.kraklab.pl.ibm.com:302", "activeDirectory": null, "w
ins2": null, "hostnamePrefix": null, "enableDhcp": false, "wins1": null, "cloudType": "VMware", "activeDirectoryUser": null,
"osNetworkId": "bc20165e-ac1b-4764-85bd-1f65979f0add", "id": "faf067e2-00e9-4ae2-a928-c544c0f4df6b", "workgroup": null}}
Networks matched!
```

5. Verify that the new cloud group is available on IBM Workload Deployer.

Checking out virtual images after migration

Virtual images are stored differently in IBM SmartCloud Orchestrator than in IBM Workload Deployer. IBM SmartCloud Orchestrator includes a component called the Virtual Image Library, which is responsible for managing virtual images.

Before you can deploy patterns on the attached hypervisor, you must first *checkout* images in the Virtual Image Library. This operation requires a working configuration between OpenStack and the attached hypervisor, and an available datastore.

The checkout operation transfers imported images to hypervisor datastores and, based on them, creates a template for later use and stores it in the datastore. After this operation, the virtual image is available for deployment. The checkout operation is required to enable IBM SmartCloud Orchestrator to use the imported virtual images. Without it, pattern deployment will fail, even if the virtual image is present in the images list.

The migration tool provides a `checkout.sh` script that can perform this checkout operation automatically. The script is defined to checkout all images only to a single datastore. If you plan to checkout virtual images to multiple datastores, use the Virtual Image Library user interface instead to perform this checkout operation.

The `checkout.sh` script is located in the `migration_tool/helpers` directory. A property file is also required, and you can find the default `checkout_data.txt` property file, also in the `/helpers` directory.

To run the `checkout_sh` script, you need to define several parameters in the `checkout_data.txt` property file:

```
##########################################
# Required data
##########################################
IWD_CLI="/opt/ibm/cli/deployer.cli"
SCO_HOSTNAME=<IP of SCO IWD - Central Server 3>
SCO_USER=<admin user>
SCO_PASSWORD=<admin password>
GATEWAY_HOSTNAME=<IP of IAAS GATEWAY - Central Server 2>
GATEWAY_PORT=9973
VMWARE_REPOSITORY="<Operational repository name from Virtual Image Library>"
VMWARE_DATASTORE="<Datastore name related to Virtual Image Library Operational respository>"
##########################################
```

The parameters are defined as follows:

IWD_CLI
The location of the IBM Workload Deployer command line interface on the migration system.

SCO_HOSTNAME
Specify the IP address of the virtual machine where IBM Workload Deployer is running in the

IBM SmartCloud Orchestrator environment. Typically this is the IP address for *Central Server 3*. This is a required parameter for the checkout process to run successfully.

SCO_USER
Specify the global SmartCloud Orchestrator administrator user name. This is a required parameter for the checkout process to run successfully.

SCO_PASSWORD
Specify the global SmartCloud Orchestrator administrator password. This is an optional parameter in this file. If you do not specify the password, the migration tool prompts you to enter it from the keyboard.

GATEWAY_HOSTNAME
Specify the IP address of the virtual machine where the IAAS Gateway is set in the IBM SmartCloud Orchestrator environment. Typically this is the IP address for *Central Server 2*. This is a required parameter for the migration tool to import users and user groups.

GATEWAY_PORT
This parameter is defined with the default value *9973*. If this port number was changed during the installation of IBM SmartCloud Orchestrator, update this parameter to the correct port number. This is a required parameter for the migration tool to import users and user groups.

VMWARE_REPOSITORY
The repository name linked to the hypervisor in the Virtual Image Library. By default, this name is the same as the network name specified while configuring the OpenStack with Vcenter.

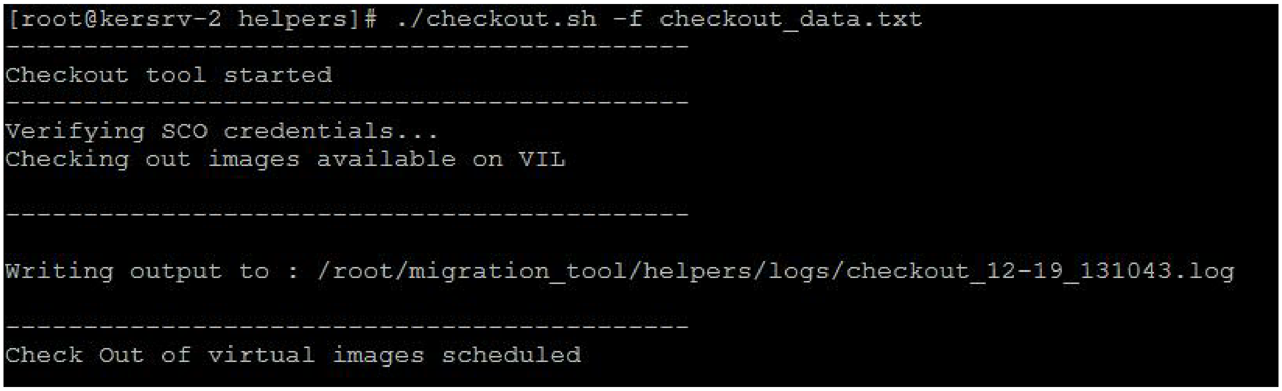
VMWARE_DATASTORE
The name of the target datastore to where virtual images are checked out.

To run the checkout operation, complete the following steps:

1. Change directory to `migration_tool/helpers`.
2. Add execute rights for the `checkout.sh` script:
   ```
   chmod +x ./checkout.sh
   ```
3. Set properties in `checkout_data.txt`.
4. Run `checkout.sh`:
   ```
   ./checkout.sh [-f|--file <property file>]
   ```
   If the file name is not specified, the default property file, `checkout_data.txt` is assumed to be used.

   The checkout operation schedules tasks for the Virtual Image Library. You can log in to the Virtual Image Library user interface to display the progress of the operation and verify that the tasks completed successfully.

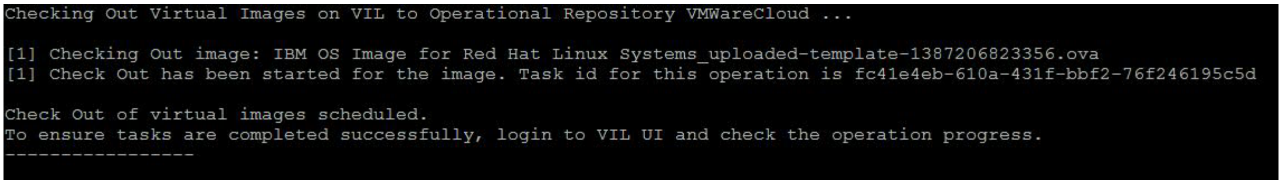
Figure 21. Example run of the `checkout.sh` script.



```
[root@kersrv-2 helpers]# ./checkout.sh -f checkout_data.txt
--------------------------------------------
Checkout tool started
--------------------------------------------
Verifying SCO credentials...
Checking out images available on VIL

--------------------------------------------

Writing output to : /root/migration_tool/helpers/logs/checkout_12-19_131043.log

--------------------------------------------
Check Out of virtual images scheduled
```

The checkout operation schedules the tasks to run asynchronously on the Virtual Image Library.

Figure 22. Example log of the `checkout.sh` operation.



```
Checking Out Virtual Images on VIL to Operational Repository VMWareCloud ...

[1] Checking Out image: IBM OS Image for Red Hat Linux Systems_uploaded-template-1387206823356.ova
[1] Check Out has been started for the image. Task id for this operation is fc41e4eb-610a-431f-bbf2-76f246195c5d

Check Out of virtual images scheduled.
To ensure tasks are completed successfully, login to VIL UI and check the operation progress.
-----------------
```

After completing all of these steps, your IBM SmartCloud Orchestrator administrator should have a working environment with migrated artifacts for IBM Workload Deployer 3.1.0.7.

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.

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.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 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
2Z4A/101
11400 Burnet Road
Austin, TX 78758 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.

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.

COPYRIGHT LICENSE:

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. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

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” at www.ibm.com/legal/copytrade.shtml.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

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

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

Other product and service names may be trademarks of IBM or other companies.

Printed in USA