

*Migrating artifacts from IBM Workload  
Deployer to IBM PureApplication  
System*



**Note**

Before using this information and the product it supports, read the information in "Notices" on page 29.

---

## Contents

<b>1 Migrating from IBM Workload Deployer to IBM PureApplication System . . . .</b>	<b>1</b>
Considerations for migrating various types of artifacts . . . . .	2
Considerations for the migration environment . . . .	4
Prerequisites . . . . .	6
<b>2 Installing the command line interface (CLI) . . . . .</b>	<b>7</b>
<b>3 Installing the migration tool . . . . .</b>	<b>9</b>
<b>4 Performing the export process . . . .</b>	<b>13</b>
<b>5 Preparing for the import process . . .</b>	<b>17</b>
<b>6 Updating pattern type information . . .</b>	<b>21</b>
<b>7 Performing the import process . . . .</b>	<b>23</b>
<b>8 Additional tasks after importing artifacts . . . . .</b>	<b>27</b>
<b>Notices . . . . .</b>	<b>29</b>



---

# 1 Migrating from IBM Workload Deployer to IBM PureApplication System

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 PureApplication™ System.

## 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 you want to migrate the patterns and virtual images used to deploy the workload instances to IBM PureApplication system Version 1.1.0.2 and later.

**Restriction:** The OS Image for AIX® used in IBM Workload Deployer uses the mksysb format, which is incompatible with IBM PureApplication System W1700. Workloads based on this image cannot be migrated to IBM PureApplication System W1700.

This migration solution is not a full migration where all artifacts and running instances are migrated to PureApplication System. With this solution, only the patterns and dependent artifacts used to deploy the workloads in IBM Workload Deployer are migrated. Existing workloads that were previously deployed using IBM Workload Deployer must continue running under that environment.

---

## 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 PureApplication system environment.

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

- Emergency fixes
- Add-ons
- Script packages
- Virtual images
- Virtual system patterns
- Pattern types (import only)
- Virtual application patterns
- Shared services

Pattern type artifacts cannot be automatically exported from IBM Workload Deployer. The migration tool exports only general information about each pattern type to a JSON object file.

To import a pattern type, you must manually copy the pattern type archive file to the filesystem local to the migration tool, or the remote location must be accessible by IBM PureApplication System. You must then edit these JSON files manually to specify the path of the pattern type archive file.

The OS Image for AIX used in IBM Workload Deployer uses the mkysb format, which is incompatible with IBM PureApplication System W1700. Workloads based on this image can be exported successfully, but they are not supported on import to IBM PureApplication System W1700. During the import operation an error message will be issued and the image will be skipped. You can, however, recreate these workloads in the IBM PureApplication System W1700 environment by using one of the virtual images provided. See “Recreating AIX workloads” on page 18 for more information.

There are significant differences between the security models in IBM Workload Deployer and in IBM PureApplication System. As a result you cannot migrate users, user groups, and the ownership and access control list (ACL) of the artifacts.

All of the imported artifacts in IBM PureApplication System are owned by the PureApplication migration user specified in the migration property file. You cannot change ownership of these artifacts after they are imported, so you must carefully consider which PureApplication System user account is used to perform the migration.

To help identify the list of artifacts migrated to PureApplication System, you might create a new user account for the migration. To perform the migration, this user account must be granted the following Workload Management permissions:

- Create new patterns
- Create new catalog content

After the migration process completes, you can sign in to the PureApplication System and view the list of migrated artifacts that are owned by this user account. You must manually create the users and user groups in the PureApplication System and grant them access to the migrated artifacts.

Environment profiles and instances are not migrated because IBM PureApplication System cannot be used to manage the existing IBM Workload Deployer cloud environments.

---

## Considerations for the migration environment

The migration process can be performed in several different environments.

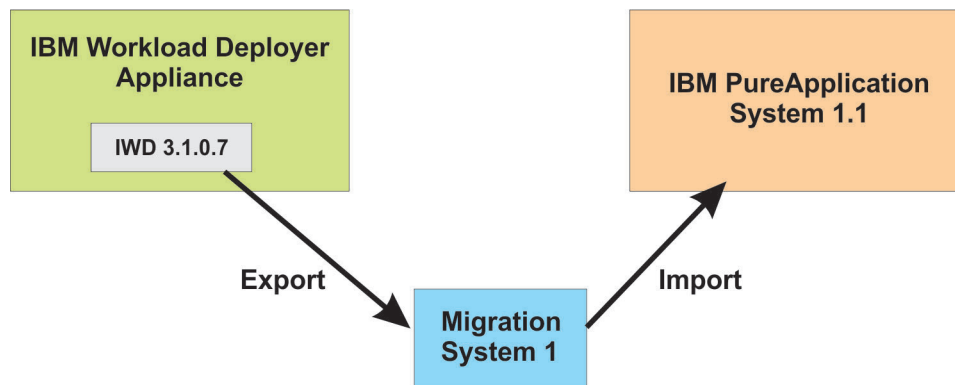
You can perform this migration using one of the following types of migration environments:

- **Environment 1:** A single migration system
- **Environment 2:** Two migration systems

### Environment 1: A single migration system

In this environment, the migration tool is installed on a separate system from both the IBM Workload Deployer appliance and IBM PureApplication system. You also must download and install the command line interface (CLI) from the IBM Workload Deployer appliance.

*Figure 1. Migration tool installed on a separate migration system with network connection to IBM Workload Deployer appliance and PureApplication System.*



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 IBM PureApplication System. 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 PureApplication System.

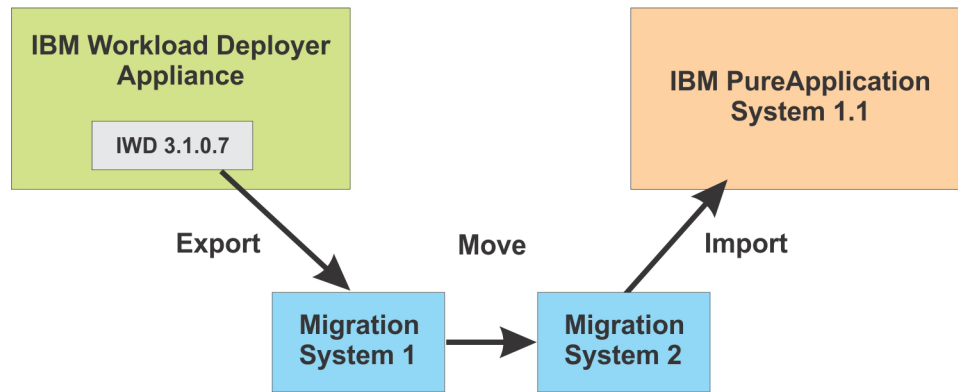
Follow the usual documented procedure for downloading the command line interface tool from the IBM Workload Deployer appliance.

### Environment 2: Two migration systems

Use this environment only if a single migration system cannot be configured to connect to both IBM Workload Deployer and IBM PureApplication System. In this environment, both the migration tool and the IBM Workload Deployer command line interface are installed on both migration systems.



Figure 2. Migration tool installed on two separate migration systems with no network connection between the IBM Workload Deployer appliance and IBM PureApplication System.



Both of these 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.

One migration system must have network access to IBM Workload Deployer, and is used to perform the export of the artifacts from the IBM Workload Deployer appliance. The second migration system must have network access to the IBM PureApplication System, and is used to perform the import of the migrated artifacts.

After the export process completes, you must manually copy the exported artifact data to the second migration system and then run the migration tool on this second system to perform the import process into IBM PureApplication System.

---

## 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 PureApplication System Version 1.1.0.2 or later installed on the same or a different network than IBM Workload Deployer.
- 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 migration systems must be running a supported Red Hat Enterprise Linux operating system, with sufficient storage available to perform the migration.
- You will need to provide pattern types that are not included with the IBM Workload Deployer default data.

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

- The migration might take many hours to complete, depending on the network and the amount of data to migrate.
- 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.

---

## 2 Installing the command line interface (CLI)

Before performing the migration, download and install the command line interface tool to your migration systems.

The migration tool uses the command line interface (CLI) from IBM Workload Deployer to export artifacts from IBM Workload Deployer, and to import the artifacts to IBM PureApplication System. Download the version of the CLI from the IBM Workload Deployer 3.1.0.7 appliance.

To download the CLI, from the Welcome page in the IBM Workload Deployer console, click **Download Tooling** and select **Download command line tool**.

Figure 3. Download the CLI from IBM Workload Deployer.



The downloaded CLI is in the form of a zip archive. Complete the following steps to install the CLI on each of the migration systems:

1. Copy the CLI archive file to any target destination on the migration system.
2. Unpack the contents of the CLI archive by issuing the following command:  
`unzip deployer.cli*.zip`
3. Note the target destination where the CLI is unpacked. You will need to specify this path during the installation of the migration tool.



---

## 3 Installing the migration tool

Depending on your migration environment, you can 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_PureApp.tar.gz`. Regardless of which migration environment (one or two migration systems), copy this file to each migration system, at a location where you plan to run the tool. You will then unpack this compressed file and configure it to perform the migration.

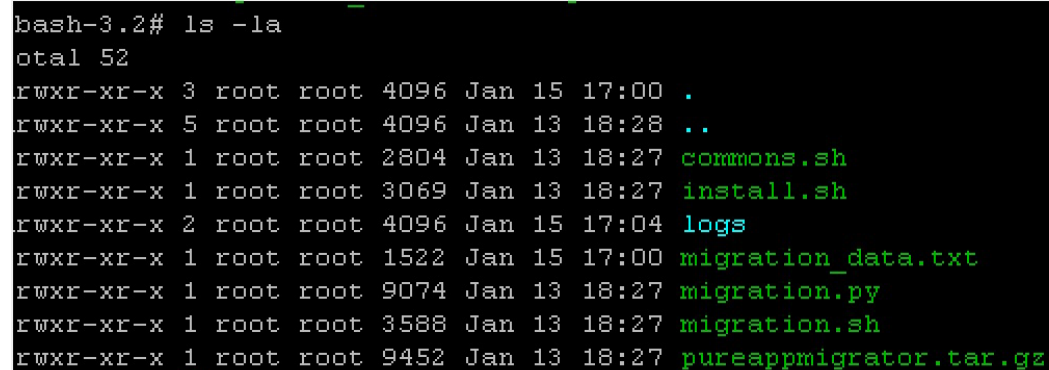
### Unpacking the migration tool

After copying the `MigrationTool_PureApp.tar.gz` file to the intended location on the migration system, complete the following steps on each migration system:

1. Unpack the contents of the file by issuing the following command:  

```
tar -xzf MigrationTool_PureApp.tar.gz
```
2. Verify that the `migration_tool` directory has been created.
3. Change the directory location to the `migration_tool` directory.
4. Your `migration_tool` directory should now look similar to the following example:

Figure 4. Contents of the `migration_tool` directory.



```
bash-3.2# ls -la
total 52
-rwxr-xr-x 3 root root 4096 Jan 15 17:00 .
-rwxr-xr-x 5 root root 4096 Jan 13 18:28 ..
-rwxr-xr-x 1 root root 2804 Jan 13 18:27 commons.sh
-rwxr-xr-x 1 root root 3069 Jan 13 18:27 install.sh
-rwxr-xr-x 2 root root 4096 Jan 15 17:04 logs
-rwxr-xr-x 1 root root 1522 Jan 15 17:00 migration_data.txt
-rwxr-xr-x 1 root root 9074 Jan 13 18:27 migration.py
-rwxr-xr-x 1 root root 3588 Jan 13 18:27 migration.sh
-rwxr-xr-x 1 root root 9452 Jan 13 18:27 pureappmigrator.tar.gz
```

### 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 using your preferred text editor. The default contents are as follows:

```
#####
# Migration properties
#####
IWD_CLI="/opt/ibm/cli/deployer.cli"
MIGRATION_DEST="/root/migration"

#####
# IWD Appliance properties
#####
```

```
# IWD_3107_HOSTNAME=<IP or Hostname of IWD 3107>
# IWD_3107_USER=<admin user>
# IWD_3107_PASSWORD=<admin password>

#####
# PureApplication System properties
#####
# PUREAPP_HOSTNAME=<IP or Hostname of PureApp>
# PUREAPP_USER=<pureapp user>
# PUREAPP_PASSWORD=<pureapp password>
#####
```

The first two parameters are required properties of the migration tool:

#### **IWD\_CLI**

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

#### **MIGRATION\_DEST**

The location where all the artifacts are temporarily stored during the migration process.

If you are running the migration tool in the single migration system environment, remove the comment (# symbol in column 1) from all of the parameters in both the IWD Appliance properties and PureApplication System properties sections. The installation process will install the migration tool for exporting from IBM Workload Deployer as well as for importing to IBM PureApplication System.

If you are running the migration tool in the two migration system environment, edit this migration\_data.txt file on both systems and complete the following steps:

- On *Migration System 1*, remove the comment symbol only from the parameters in the IWD Appliance properties section. The installation process will install the migration tool for exporting artifact data from the IBM Workload Deployer appliance to *Migration System 1*.
- On *Migration System 2*, remove the comment symbol only from the parameters in the PureApplication System properties section. The installation process will install the migration tool for importing artifact data from *Migration System 2* to IBM PureApplication System.

The IWD Appliance properties section contains the following parameters that are used to connect to IBM Workload Deployer to export the artifacts:

#### **IWD\_3107\_HOSTNAME**

Specify the IP address or hostname of the IBM Workload Deployer version 3.1.0.7 appliance. This parameter is required for the installation process to run successfully.

#### **IWD\_3107\_USER**

Specify a valid user name to access the IBM Workload Deployer appliance. This user must have the necessary privileges to export the artifacts from IBM Workload Deployer. This parameter is required for the installation process to run successfully.

#### **IWD\_3107\_PASSWORD**

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

The PureApplication System properties section contains the following parameters that are used to connect to IBM PureApplication System to import the migrated artifacts:

#### **PUREAPP\_HOSTNAME**

Specify the IP address or hostname of the IBM PureApplication System. This parameter is required for the installation process to run successfully.

## **PUREAPP\_USER**

Specify the user name that is used to access IBM PureApplication System. This user must have the following Workload Management permissions to import artifacts into the PureApplication System:

- Create new patterns
- Create new catalog content

All imported artifacts will be owned by this user. This parameter is required for the installation process to run successfully.

## **PUREAPP\_PASSWORD**

Specify the password for the PureApplication System user. This parameter is optional in this file. If you do not specify the password, the migration tool prompts you to enter it from the keyboard.

## **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
```

The installation script searches the current directory for the `migration_data.txt` property file and uses those parameters as input to the process.

The installation script connects to IBM Workload Deployer and IBM PureApplication System as needed, and verifies the user access credentials.

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





---

## 4 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>]
```

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
- **efix**: migrate emergency fixes
- **script**: migrate script packages
- **addon**: migrate add-ons
- **virting**: 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.

The following examples show valid syntax for the artifact list:

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

Examples:

- `./migration.sh -m export`

This command performs an export operation on all available artifacts.

- `./migration.sh --mode export -a all`

This command performs an export operation on all available artifacts.

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

This command performs an export operation only on virtual application pattern artifacts.

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

This command performs an export operation only on script packages, add-ons, and virtual system pattern artifacts.

## 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:

- A connection is made to the IBM Workload Deployer appliance to verify the credentials.
- The amount of free space at the target location (specified by **MIGRATION\_DEST**) 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 log file in the directory `migration_tool/logs`.

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.

```
-bash-3.2# ls -ls
total 28
4 drwxr-xr-x 10 root root 4096 Dec 18 04:38 add_ons
4 drwxr-xr-x  3 root root 4096 Dec 18 04:59 emergency_fixes
4 drwxr-xr-x 10 root root 4096 Dec 18 04:59 pattern_types
4 drwxr-xr-x  5 root root 4096 Jan  2 22:23 script_packages
4 drwxr-xr-x 24 root root 4096 Dec 18 04:59 virtual_application_patterns
4 drwxr-xr-x  2 root root 4096 Jan 15 17:21 virtual_images
4 drwxr-xr-x  5 root root 4096 Dec 18 04:59 virtual_system_patterns
```

The directory names that are created include:

- add\_ons
- emergency\_fixes
- pattern\_types
- script\_packages
- virtual\_application\_patterns
- virtual\_images
- virtual\_system\_patterns

## Handling existing artifact directories

If an artifact 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 migration\_tool/logs/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.

---

## 5 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 PureApplication System environment.

Before you begin importing artifacts into the IBM PureApplication System 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 PureApplication System environment.

### Requirements for importing artifacts

Ensure that the following requirements are met:

- If you are migrating using the two migration system environment, you should have already installed the IBM Workload Deployer command line interface (CLI) and the migration tool on *Migration System 2* (the system from which you are importing artifacts into IBM PureApplication System), and you should have copied all exported artifacts to this migration system.

**Restriction:** The OS Image for AIX virtual image used in IBM Workload Deployer uses the mksysb format, which is incompatible with IBM PureApplication System W1700. You might have exported this image successfully with other images, but you cannot import workloads based on this image into IBM PureApplication System W1700.

You can, however, use an OS Image for AIX virtual image that is provided in IBM PureApplication System W1700 and customize it to use for importing virtual system patterns from IBM Workload Deployer that referenced the mksysb formatted image. See “Recreating AIX workloads” on page 18 for more information.

- 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 (`migration_data.txt`).
- Ensure that the parameters in the PureApplication System properties section of the migration property file are specified correctly to connect to the IBM PureApplication System.

### 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:

- Script packages and add-ons are not required to be imported before importing virtual images. The virtual image export archive contains the dependent script packages and add-ons, and these are imported automatically during the import of the virtual image.
- Import virtual images before importing virtual system patterns. If the virtual images are not available, the import of virtual system patterns fails.



- 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 PureApplication System:

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 PureApplication System:

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, manually update the pattern.json file for each pattern type to specify the location of the corresponding pattern type archive file to be imported. For more information on this procedure, see chapter 6, “Updating pattern type information,” on page 21.

## Recreating AIX workloads

If you had workloads in IBM Workload Deployer that were based on the OS Image for AIX, you might have included that image and the associated virtual system patterns and artifacts in the migration export process. Because that virtual image is in mksysb format, it is not compatible with the IBM PureApplication System W1700 environment, and any attempts to import these components will fail.

IBM PureApplication System W1700, however, provides several compatible versions of the OS Image for AIX Systems virtual image. You can use one of these images to manually recreate your AIX workload from IBM Workload Deployer. Follow this general procedure:

1. Select one of the virtual images provided with IBM PureApplication System W1700. For more information about the available virtual images, see the following link in the IBM PureApplication

System W1700 information center: [http://pic.dhe.ibm.com/infocenter/psappsys/v1r1m0/topic/com.ibm.puresystems.appsys.1700.doc/iwd/pcc\\_preload\\_aibase.html](http://pic.dhe.ibm.com/infocenter/psappsys/v1r1m0/topic/com.ibm.puresystems.appsys.1700.doc/iwd/pcc_preload_aibase.html)

2. Extend, customize (as needed), and then recapture the virtual image. For more information, see the following link in the information center: [http://pic.dhe.ibm.com/infocenter/psappsys/v1r1m0/topic/com.ibm.puresystems.appsys.1700.doc/iwd/pct\\_extend\\_vi.html](http://pic.dhe.ibm.com/infocenter/psappsys/v1r1m0/topic/com.ibm.puresystems.appsys.1700.doc/iwd/pct_extend_vi.html)
3. Before you can import your virtual system patterns and artifacts that you exported from IBM Workload Deployer, you need to manually update the information in each virtual system pattern that references the virtual image:
  - a. Navigate to the target location where the exported artifact data is located. This location is specified in the MIGRATION\_DEST parameter in the migration property file. The default target location is /root/migration.
  - b. Change to the virtual\_system\_patterns directory.
  - c. List the contents of the virtual\_system\_patterns directory to display the subdirectory for each pattern that was exported from IBM Workload Deployer.
  - d. Change directory to the pattern to be imported.
  - e. Open the patterns.json file. The contents look similar to the following example:

```
"references": {
  "virtual_images": [
    {
      "advancedoptionsaccepted": "F",
      "build": "118",
      "name": "IBM Workload Deployer Image for x86 Systems",
      "version": "1.0.0.2"
    }
  ]
}
```
  - f. Update the name, version, and build that is used to uniquely identify the virtual image in IBM PureApplication System W1700. You can find this information in the **Virtual Images** page of the IBM PureApplication System W1700 Workload Console. The build of a virtual image is displayed with the label **Image reference number**. The following example data is for AIX version 6.1:  
Name: IBM OS Image for AIX Systems  
Version: 1.0.0.4  
Image reference number: 103
  - g. Save your changes. Repeat for each virtual system pattern that references the virtual image.
4. Proceed with the import process to migrate the virtual system patterns and artifacts into the IBM PureApplication System W1700 environment.
5. After the migration process completes successfully, you can deploy the patterns.

## Additional tasks after importing artifacts

After you perform the import operation, you will need to perform additional manual steps to grant access permissions to the imported artifacts before you have a fully functioning IBM PureApplication System environment.

For more information on this task, see chapter 8, “Additional tasks after importing artifacts,” on page 27.





---

## 6 Updating pattern type information

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

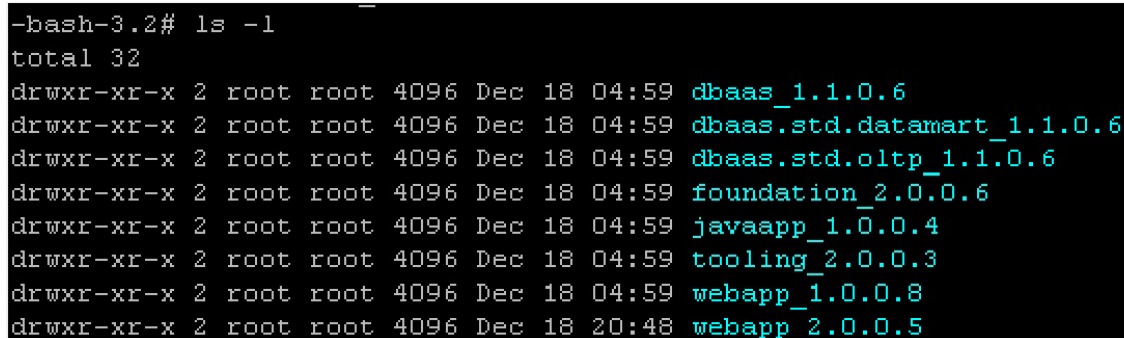
The migration tool is not able to export the pattern type archives because this operation is not supported by IBM Workload Deployer. The migration tool exports only general metadata information about the pattern types.

To import your pattern types to the IBM PureApplication System, you must edit the `pattern.json` file for each pattern type and specify the location of the pattern type archive file.

To update your pattern type information for importing into IBM PureApplication System, complete the following steps for each pattern type on the system from which the artifact data will be imported:

1. Navigate to the target location where the exported artifact data is located. This location is specified in the **MIGRATION\_DEST** parameter in the migration property file. The default target location is `/root/migration`.
2. Change to 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.



```
-bash-3.2# ls -l
total 32
drwxr-xr-x 2 root root 4096 Dec 18 04:59 dbaas_1.1.0.6
drwxr-xr-x 2 root root 4096 Dec 18 04:59 dbaas.std.datamart_1.1.0.6
drwxr-xr-x 2 root root 4096 Dec 18 04:59 dbaas.std.oltp_1.1.0.6
drwxr-xr-x 2 root root 4096 Dec 18 04:59 foundation_2.0.0.6
drwxr-xr-x 2 root root 4096 Dec 18 04:59 javaapp_1.0.0.4
drwxr-xr-x 2 root root 4096 Dec 18 04:59 tooling_2.0.0.3
drwxr-xr-x 2 root root 4096 Dec 18 04:59 webapp_1.0.0.8
drwxr-xr-x 2 root root 4096 Dec 18 20:48 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 PureApplication System is able to connect to the specified location, because the PureApplication System connects directly to the remote system to download the archive file.

Repeat this procedure for each pattern type to be imported.

**Note:** If you attempt to import a virtual application pattern but the associated pattern type has not already been imported into IBM PureApplication System, the import of the pattern will fail.

---

## 7 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 PureApplication System environment.

### Import command syntax

To run the migration tool for importing artifacts into IBM PureApplication System, 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>]
```

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
- **efix**: migrate emergency fixes
- **script**: migrate script packages
- **addon**: migrate add-ons
- **virting**: 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 vsys,script,vapp,virting
--artifacts "vapp,vsys,addon"
-a 'ptype,vapp'
```

Examples:

- `./migration.sh -m import`

This command performs an import operation on all available artifacts.

- `./migration.sh --mode import -a all`

This command performs an import operation on all available artifacts.

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

This command performs an import operation only on virtual application pattern artifacts.

- `./migration.sh -m import -a "script,addon,vsys"`

This command performs an import operation only on script packages, add-ons, and virtual system pattern artifacts.

## Performing the import operation

The import migration operation imports the 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.

```
-bash-3.2# ./migration.sh -m import -a "all"
-----
Migration tool started
-----
Verifying PureApp credentials...
Artifacts to migrate: all
Writing output to : /root/migratepure/migration_tool/logs/import_01-10_190956.log
```

The import operation performs the following tasks:

- A connection is made to the IBM PureApplication System to verify the credentials.
- Based on the specified list of artifacts, the migration tool imports the artifacts from the target location into IBM PureApplication System. Output log information is written to a log file in the `migration_tools/logs` directory. Check this log to verify that all artifacts are imported successfully.

## Handling duplicate artifacts

As you import artifacts into the IBM PureApplication System environment, you might encounter a situation where some artifacts already exist. This situation might occur if the artifacts are included in the default data, or if the import process is run multiple times.

If the import operation discovers an artifact that already exists in the IBM PureApplication System 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 you can be informed that 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 PureApplication System environment:

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

```
Importing artifacts from /root/migration
-----

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

[1] [2014-01-14 22:17:21.424000 UTC] Processing directory: /root/migration/script_packages/1
.
[1] Importing Script Package: test ...
[1] Script Package successfully imported.

[2] [2014-01-14 22:17:21.539999 UTC] Processing directory: /root/migration/script_packages/2
.
[2] Importing Script Package: Migration SP2 ...
[2] Script package test2 already exists. Skipping

[3] [2014-01-14 22:17:22.430999 UTC] Processing directory: /root/migration/script_packages/3
.
[3] Importing Script Package: AMC Cleanup Script Package ...
[3] Script package AMC Cleanup Script Package already exists. Skipping

Import of Script Packages completed
-----
```

In this example, the import operation is attempting to import three 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 that uniquely identifies 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/<script_package_dir>` directory, and edit the `script.json` file for the *Migration SP2* script package, which might look similar to the following example:

```
[
  {
    "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.



---

## 8 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 PureApplication System environment.

To complete the migration process, you need to grant access to the migrated artifacts.

### Granting access to the migrated artifacts

There are significant differences between the security model in IBM Workload Deployer and IBM PureApplication System. As a result, users, user groups, and the ownership and access control list (ACL) of the artifacts cannot be migrated. All the imported artifacts in IBM PureApplication System will be owned by the PureApplication migration user specified in the migration property file. You cannot change ownership of these artifacts after they are imported, but you can grant users access to them.

Complete the following steps to grant user access to the migrated artifacts:

1. Logon to the IBM PureApplication System, using the PureApplication System migration user specified in the migration property file. All the migrated artifacts will be owned by this user.
2. From the Workload Console, select the type of artifact (such as virtual images, script packages, or add-ons) to grant access.
3. Select the artifact for which you want to grant user access, and in the **Access granted to** field, type the name of each user to access the selected artifact.





---

## 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](http://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](http://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<sup>™</sup> 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