

16.1.2



Open Service
Catalog Manager

VMware vSphere Integration (GlassFish)

November 2016

Trademarks

LINUX is a registered trademark of Linus Torvalds.

Microsoft and Windows are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

Open Service Catalog Manager is a registered trademark of FUJITSU LIMITED.

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

Apache Ant, Ant, and Apache are trademarks of The Apache Software Foundation.

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

VMware vSphere is a registered trademark of VMware in the United States and in other countries.

Other company names and product names are trademarks or registered trademarks of their respective owners.

Copyright FUJITSU LIMITED 2016

High Risk Activity

The Customer acknowledges and agrees that the Product is designed, developed and manufactured as contemplated for general use, including without limitation, general office use, personal use, household use, and ordinary industrial use, but is not designed, developed and manufactured as contemplated for use accompanying fatal risks or dangers that, unless extremely high safety is secured, could lead directly to death, personal injury, severe physical damage or other loss (hereinafter "High Safety Required Use"), including without limitation, nuclear reaction control in nuclear facility, aircraft flight control, air traffic control, mass transport control, medical life support system, missile launch control in weapon system. The Customer shall not use the Product without securing the sufficient safety required for the High Safety Required Use. In addition, FUJITSU (or other affiliate's name) shall not be liable against the Customer and/or any third party for any claims or damages arising in connection with the High Safety Required Use of the Product.

Export Restrictions

Exportation/release of this document may require necessary procedures in accordance with the regulations of your resident country and/or US export control laws.

Contents

| | | |
|------------|---|-----------|
| | About this Manual..... | 5 |
| 1 | Introduction..... | 8 |
| 1.1 | Components Involved in the vSphere Integration..... | 8 |
| 1.2 | Usage Scenarios..... | 9 |
| 2 | Installing the vSphere Integration Software..... | 10 |
| 2.1 | Prerequisites and Preparation..... | 10 |
| 2.1.1 | OSCM and vSphere..... | 10 |
| 2.1.2 | Hardware and Operating Systems..... | 10 |
| 2.1.3 | Java and Ant..... | 10 |
| 2.1.4 | Application Server..... | 11 |
| 2.1.5 | Relational Database..... | 11 |
| 2.1.6 | Mail Server..... | 12 |
| 2.2 | Installation..... | 12 |
| 2.2.1 | Preparing the Software and Setup Utilities..... | 12 |
| 2.2.2 | Configuring the vSphere Integration..... | 13 |
| 2.2.3 | Setting up the Databases..... | 15 |
| 2.2.4 | Setting up the Application Server Resources..... | 17 |
| 2.2.5 | Exchanging Certificates..... | 18 |
| 2.3 | Installing the VMware Service Controller in an Existing APP Environment..... | 19 |
| 2.4 | Update Installation..... | 21 |
| 3 | Creating and Publishing Services..... | 25 |
| 3.1 | Prerequisites and Preparation..... | 25 |
| 3.2 | Creating Technical Services..... | 25 |
| 3.3 | Creating and Publishing Marketable Services..... | 26 |
| 4 | Using vSphere Services in OSCM..... | 27 |
| 4.1 | Subscribing to Services..... | 27 |
| 4.2 | Modifying Subscription Parameters..... | 27 |
| 4.3 | Executing Service Operations..... | 28 |
| 4.4 | Terminating Subscriptions..... | 28 |

| | | |
|-----------------|--|-----------|
| 5 | Administrating the vSphere Integration..... | 29 |
| 5.1 | Controlling the Provisioning Process..... | 29 |
| 5.2 | Handling Problems in the Provisioning Process..... | 29 |
| 5.3 | Handling Communication Problems Between APP and OSCM..... | 30 |
| 5.4 | Backup and Recovery..... | 31 |
| 5.5 | Updating Configuration Settings..... | 31 |
| 5.6 | Adapting the Log Configuration..... | 33 |
| 6 | Uninstallation..... | 35 |
| | Appendix A Configuration Settings..... | 36 |
| A.1 | GlassFish Configuration Settings..... | 36 |
| A.2 | APP Database Configuration Settings..... | 38 |
| A.3 | APP Configuration Settings..... | 39 |
| A.4 | Controller Database Configuration Settings..... | 41 |
| A.5 | vSphere Configuration..... | 41 |
| A.6 | Controller Configuration Settings..... | 45 |
| | Appendix B Service Parameters and Operations..... | 48 |
| Glossary | | 57 |

About this Manual

This manual describes the integration of VMware vSphere with Open Service Catalog Manager (OSCM).

This manual is structured as follows:

| Chapter | Description |
|---|---|
| <i>Introduction</i> on page 8 | Provides an overview of the OSCM vSphere integration, the components involved, and the supported usage scenarios. |
| <i>Installing the vSphere Integration Software</i> on page 10 | Describes how to prepare and carry out the installation of the vSphere integration software. |
| <i>Creating and Publishing Services</i> on page 25 | Describes how to create and publish services for vSphere in OSCM. |
| <i>Using vSphere Services in OSCM</i> on page 27 | Describes how to provision, modify, and deprovision virtual machines in vSphere through services in OSCM. |
| <i>Administering the vSphere Integration</i> on page 29 | Describes administration tasks related to the OSCM vSphere integration. |
| <i>Uninstallation</i> on page 35 | Describes how to uninstall the OSCM vSphere integration software. |

Readers of this Manual

This manual is intended for operators who want to offer virtual machines controlled by vSphere through services on a marketplace provided by OSCM. It assumes that you have access to an existing OSCM installation and to a vSphere virtualization platform. In addition, you should have basic knowledge of vSphere and you should be familiar with the concepts and administration of OSCM.

Notational Conventions

This manual uses the following notational conventions:

| | |
|-------------------------------|--|
| Add | The names of graphical user interface elements like menu options are shown in boldface. |
| <code>init</code> | System names, for example command names and text that is entered from the keyboard, are shown in Courier font. |
| <code><variable></code> | Variables for which values must be entered are enclosed in angle brackets. |
| <code>[option]</code> | Optional items, for example optional command parameters, are enclosed in square brackets. |
| <code>one two</code> | Alternative entries are separated by a vertical bar. |
| <code>{one two}</code> | Mandatory entries with alternatives are enclosed in curly brackets. |

Abbreviations

This manual uses the following abbreviations:

| | |
|-------------|--|
| APP | Asynchronous Provisioning Platform |
| OSCM | Open Service Catalog Manager |
| DBMS | Database Management System |
| IaaS | Infrastructure as a Service |
| IdP | SAML Identity Provider |
| SAML | Security Assertion Markup Language |
| STS | Security Token Service |
| VM | Virtual Machine |
| WSDL | Web Services Description Language |
| WSIT | Web Services Interoperability Technologies |

Available Documentation

The following documentation on OSCM is available:

- *Overview*: A PDF manual introducing OSCM. It is written for everybody interested in OSCM and does not require any special knowledge.
- *Online Help*: Online help pages describing how to work with the administration portal of OSCM. The online help is intended for and available to everybody working with the administration portal.
- *Installation Guide (GlassFish)*: A PDF manual describing how to install and uninstall OSCM. It is intended for operators who set up and maintain OSCM in their environment.
- *Operator's Guide*: A PDF manual for operators describing how to administrate and maintain OSCM.
- *Technology Provider's Guide*: A PDF manual for technology providers describing how to prepare applications for usage in a SaaS model and how to integrate them with OSCM.
- *Supplier's Guide*: A PDF manual for suppliers describing how to define and manage service offerings for applications that have been integrated with OSCM.
- *Reseller's Guide*: A PDF manual for resellers describing how to prepare, offer, and sell services defined by suppliers.
- *Broker's Guide*: A PDF manual for brokers describing how to support suppliers in establishing relationships to customers by offering their services on a marketplace.
- *Marketplace Owner's Guide*: A PDF manual for marketplace owners describing how to administrate and customize marketplaces in OSCM.
- *Developer's Guide*: A PDF manual for application developers describing the public Web services and application programming interfaces of OSCM and how to integrate applications and external systems with OSCM.
- *ServerView Resource Orchestrator Integration (GlassFish)*: A PDF manual for operators describing how to offer and use virtual platforms and servers controlled by FUJITSU ServerView Resource Orchestrator through services in OSCM.

- *Amazon Web Services Integration (GlassFish)*: A PDF manual for operators describing how to offer and use virtual servers controlled by the Amazon Elastic Compute Cloud Web service through services in OSCM.
- *OpenStack Integration (GlassFish)*: A PDF manual for operators describing how to offer and use virtual systems controlled by OpenStack through services in OSCM.
- *VMware vSphere Integration (GlassFish)*: A PDF manual for operators describing how to offer and use virtual machines provisioned on a VMware vSphere server through services in OSCM.
- Javadoc and YAML documentation for the public Web services and application programming interfaces of OSCM and additional resources and utilities for application developers.

1 Introduction

Open Service Catalog Manager (OSCM) is a set of services which provide all business-related functions and features required for turning on-premise applications and tools into 'as a Service' (aaS) offerings and using them in the Cloud. This includes ready-to-use account and subscription management, online service provisioning, billing and payment services, and reporting facilities.

VMware vSphere is a server and infrastructure virtualization platform that allows for running multiple operating systems and applications on a single server, the vSphere server. It gives you on-demand access to fully configured virtual machines (VMs) that are provisioned by cloning VM templates provided in vSphere. The provisioning of computing resources in the Cloud allows you to rapidly scale and flex your infrastructure to support new business initiatives or roll out services.

The integration of OSCM and vSphere provides for an Infrastructure as a Service (IaaS) solution that leverages the features of both products: Through services, which are published on a marketplace in OSCM, users can request and use VMs on a vSphere server. The usage costs can be calculated and charged by means of the OSCM billing and payment services.

The vSphere integration package provided with OSCM includes all the components required for connecting an existing OSCM installation with vSphere. This manual describes how to deploy this package and how to create and use services for vSphere on a OSCM marketplace.

1.1 Components Involved in the vSphere Integration

The following picture provides an overview of the main components involved in the integration of OSCM and vSphere:



In OSCM, customer subscriptions are managed by means of the **Subscription service**. When a customer creates, modifies, or terminates a subscription for a VM in vSphere, the Subscription service asynchronously triggers the corresponding actions in vSphere through the **Asynchronous Provisioning Platform (APP)** and the **VMware service controller**: VMs are created, modified, or deleted in vSphere.

APP is a framework which provides a provisioning service, an operation service, as well as functions, data persistence, and notification features which are required for integrating applications with OSCM in asynchronous mode. The actual communication with the applications is carried

out by service controllers. APP and the VMware service controller are the main components that make up the vSphere integration software.

In vSphere, the basic configuration of VMs for typical usage scenarios is defined by the vSphere administrators in so-called **VM templates**. A template defines, for example, the operating system used by the VM to be provisioned. For every template, any number of technical services can be created in OSCM. On the other hand, a technical service can reference several templates, and a user may choose a template when subscribing to a corresponding service. The technology provider organization is responsible for knowing and providing the correct VM template names. The vSphere API is contacted by the VMware service controller for triggering actions in vSphere.

1.2 Usage Scenarios

The OSCM vSphere integration supports the following usage scenarios:

- **Provisioning of a VM:** When a customer subscribes to a corresponding service on a OSCM marketplace, vSphere is triggered to create a VM based on a specific VM template.
- **Vertical scale-up and scale-down:** By changing the corresponding parameters at an existing subscription, customers can trigger vSphere to add resources (disks, CPU, RAM, etc.) to a provisioned VM. In the same way, they can request vSphere to remove resources from a VM.
- **Starting and stopping a VM:** A customer can explicitly start and stop a VM on vSphere by executing a service operation at the corresponding subscription.
- **Creating a snapshot and restoring a VM:** A customer can explicitly create a snapshot of a VM on vSphere and restore it by executing a service operation at the corresponding subscription.
- **Deletion of a VM:** When a customer terminates a subscription for a VM, vSphere is triggered to delete it. The subscription is terminated independent of whether the deletion is successful.

In vSphere, the VMs created for OSCM subscriptions are managed in the same way as other VMs. They can be viewed and monitored with the standard vSphere management tools.

For more details on the usage scenarios, refer to *Using vSphere Services in OSCM* on page 27.

2 Installing the vSphere Integration Software

The following sections describe how to install and configure the vSphere integration software as well as the preparations you need to take beforehand.

Installing the vSphere integration software consists of installing APP, registering the VMware service controller and setting up a database for the controller.

If you already have a working APP installation in your environment, proceed as described in *Installing the VMware Service Controller in an Existing APP Environment* on page 19.

2.1 Prerequisites and Preparation

The following sections describe the prerequisites that must be fulfilled and the preparations you need to take before installing and deploying the vSphere integration software.

2.1.1 OSCM and vSphere

- You must have access to a fully functional OSCM installation. You can install the vSphere integration software in the same environment or on a different server.
- You must have access to OSCM as an administrator and as a technology manager of an organization that has at least the technology provider role.
- You must have access to a fully functional vSphere installation via the URL of its vCenter. The vSphere integration requires the following vSphere API version: v5.5.
- In vSphere, a technical user must exist for accessing the vSphere API. This user must have the following permissions in vSphere:
 - Virtual machine/*: A set of privileges that allow the user to interact with a VM's console, insert media, and perform power operations.
 - Resource: Assign VM to resource pool
 - Datastore: Allocate space
 - Network: Assign network

For details on vCenter, users, and their privileges in vSphere, refer to the vSphere documentation.

2.1.2 Hardware and Operating Systems

The vSphere integration software as a Java application does not rely on specific hardware or operating systems. It can be deployed on any platform supported by the application server and the database management system.

2.1.3 Java and Ant

The vSphere integration software requires a Java Development Kit (JDK), version 7, 64 bit. Deployment with JDK 7, Update 45 has been tested and is recommended.

Due to a CORBA library change which is incompatible with Oracle GlassFish Server version 3.1.2.2, deployment with JDK 7, Update 55 and higher is not supported.

In order to be able to execute the installation scripts, you need to install the Apache Ant 1.8 (or higher) open source software. In the subsequent sections, `<ANT_HOME>` is the installation directory of Apache Ant.

2.1.4 Application Server

The vSphere integration software must be deployed on an application server compatible with Java EE version 6. The following application server is supported:

Oracle GlassFish Server, version 3.1.2.2.

You can deploy the vSphere integration software on the application server you use for OSCM. Alternatively, you can use a separate application server installation.

Note: Before installing GlassFish, make sure that the `JAVA_HOME` environment variable points to a Java Development Kit (JDK), version 7, 64 bit.

Proceed as follows:

1. Install the application server as described in its documentation, and configure it as required by your environment.

Note: Make sure that the path of the GlassFish installation directory does not contain blanks.

2. After you have configured GlassFish, make a backup copy of the GlassFish installation.
3. Make sure that GlassFish is running in a JDK 7 environment. Also, make sure that no other applications (e.g. Tomcat) are running on your GlassFish ports.

The installation of the vSphere integration software creates the `app-domain` domain in your application server. If required, you can change the domain name in the `glassfish.properties` file before starting the installation.

In the subsequent sections, `<GLASSFISH_HOME>` is the installation directory of GlassFish.

2.1.5 Relational Database

The vSphere integration software stores its data in a relational database. An additional relational database is used to store the vSphere configuration for use by the VMware service controller. The following database management system (DBMS) is supported:

PostgreSQL, version 9.1.12.

Install the DBMS as described in its documentation.

If required, you can use a separate machine for the vSphere integration databases.

Setup and Configuration

Edit the file

`<postgres_dir>/data/postgresql.conf`

as follows (`<postgres_dir>` is the PostgreSQL installation directory):

1. Set the `max_prepared_transactions` property value to 50.
2. Set the `max_connections` property value to 210.

This property determines the maximum number of concurrent connections to the database server.

Note the following: This setting is used in combination with the JDBC pool size settings for the domains on your application server. If you change the JDBC pool size, you might need to adapt the `max_connections` setting. Refer to the *OSCM Operator's Guide*, section *Tuning Performance*, for details.

3. Set the `listen_addresses` property value:

Specify the IP addresses of all application servers on which the database server is to listen for connections from client applications. If you use the entry '*', which corresponds to all available IP addresses, you must be aware of possible security holes.

4. Save the file.

If you use a server name in all configuration files instead of `localhost` during installation, edit the file

`<postgres_dir>/data/pg_hba.conf`

as follows (`<postgres_dir>` is the PostgreSQL installation directory):

1. Add the IP address of the application server that is to host the vSphere integration software.

For example:

```
host all all 123.123.12.1/32 md5
```

Also add the application server's IPv6 address.

For example:

```
host all all fe80::cdfb:b6ed:9b38:cf17/128 md5
```

There are authentication methods other than `md5`. For details, refer to the PostgreSQL documentation.

2. Save the file.

Restart your PostgreSQL server for the changes to take effect.

2.1.6 Mail Server

To inform users about relevant issues, the vSphere integration software requires a mail server in its environment. You can use any mail server that supports SMTP.

The settings for addressing the mail server are defined in the `glassfish.properties` file of the vSphere integration package.

2.2 Installation

The installation of the vSphere integration software consists of the following main steps:

- *Preparing the Software and Setup Utilities* on page 12
- *Configuring the vSphere Integration* on page 13
- *Setting up the Databases* on page 15
- *Setting up the Application Server Resources* on page 17
- *Exchanging Certificates* on page 18

2.2.1 Preparing the Software and Setup Utilities

The vSphere integration software and setup utilities are provided in the vSphere integration package, `oscm-vmware-install-pack.zip`. The contents of the package need to be made available in your environment as follows:

Extract the contents of the `oscm-vmware-install-pack.zip` package to a separate temporary directory on the system from where you want to install and deploy the vSphere integration software.

In the following sections, this directory is referred to as `<install_pack_dir>`.

After extraction, the following directories are available:

- **databases/app_db**
Configuration files for setting up the database used by the vSphere integration software for storing its own data.
- **databases/controller_db**
 - Configuration file (`db.properties`) for setting up the database for storing the vSphere configuration for use by the VMware service controller.
 - `/csv:`
Files (`*.csv`) for specifying the vSphere configuration.
- **doc**
The *VMware vSphere Integration* guide (this manual).
- **domains/app_domain**
 - Configuration file (`glassfish.properties`) for setting up the application server resources for the domain to which the vSphere integration software is to be deployed.
 - `applications:` The APP application (`oscm-app.ear`) and the VMware service controller (`oscm-app-vmware.ear`).
- **install**
XML files that support you in setting up the databases and the application server resources for APP.
- **samples**
Technical service sample and a sample script file for changing the Linux root password.

2.2.2 Configuring the vSphere Integration

The vSphere integration software and setup utilities require a number of settings. These settings are provided in the following subdirectories and files of `<install_pack_dir>`:

- **databases/app_db**
 - `db.properties:` Settings for the APP database setup and access.
 - `configsettings.properties:` Configuration settings for APP.
The initial installation stores these settings in the `bssapp` database, where you can change them later, if required. An update installation only adds new settings to the database, but does not overwrite existing ones.
 - `configsettings_controller.properties:` Configuration settings for the VMware service controller.
The initial installation stores these settings in the `bssapp` database. You can change them later, if required.
The `configsettings_controller.properties` file specifies the organization ID and user credentials for accessing OSCM. For security reasons, it is recommended that you delete the file as soon as you have successfully installed and configured the vSphere integration software.

- **databases/controller_db**

- `db.properties`: Settings for the VMware service controller database setup and access.
- `csv/*: .csv` files for specifying the vSphere configuration.

The initial installation stores these settings in the `vmware` database. Usually, you do not need to change them once they have been imported into the database.

The `vcenter.csv` file specifies the URL and user credentials for accessing the vSphere server. For security reasons, it is recommended that you delete the file as soon as you have successfully installed and configured the vSphere integration software.

- **domains/app_domain**

The configuration settings for setting up the application server domain to which APP is deployed are provided in the following file:

`glassfish.properties`

Additional configuration files contained in other subdirectories are used internally and must not be changed.

For details on the configuration settings, refer to *Configuration Settings* on page 36. For details on updating the configuration settings, refer to *Updating Configuration Settings* on page 31.

You need to adapt the settings in the files above to your environment. In particular, server names, ports, paths, user IDs, and passwords require adaptation.

Proceed as follows to view and adjust the configuration settings:

1. Open each of the configuration files listed above with an editor.
2. Check the settings in each file and adapt them to your environment.
3. Save the files to their original location in `<install_pack_dir>/<subdirectory>`. For future reference, it is a good idea to create a backup of the files.

Observe the following configuration issues:

- The specified ports are suggestions and work with the default settings used in the files.
- If you install everything on the local system, use either the server name or `localhost` in all configuration files for all URLs that need to be resolved by APP.

The `APP_BASE_URL` setting in the `configsettings.properties` file for the `app-domain` domain must be resolved by clients. They always require that the server name be specified.

Specify the `APP_BASE_URL` setting as follows:

`APP_BASE_URL=http://<host>:<port>/oscm-app`

If you have changed the `glassfish.domain.portbase` setting in the `glassfish.properties` file, you must change the port here accordingly.

Configuration for SAML_SP authentication mode:

If the OSCM installation you want to work with is configured for SAML_SP authentication mode, Web service calls to it are secured and authenticated by a Security Token Service (STS). This is a Web service that issues security tokens as defined in the WS-Security/WS-Trust specification. The STS is usually provided by the Identity Provider (IdP) system in use (for example Active Directory Federation Service, Cloudminder, or OpenAM).

To use an STS for Web service calls, you must perform the following steps before installing the vSphere integration software:

1. From the IdP or OSCM operator, obtain a metadata exchange file in WSDL format generated with and for the IdP system in use. The metadata includes namespace information required for connecting to the STS.
2. Save the metadata exchange file to the following file, overwriting the existing empty file:
`<install_pack_dir>/domains/app_domain/wsit/STSService.xml`
3. Open the `STSService.xml` file and retrieve the value of `targetNamespace`, for example:
`http://schemas.microsoft.com/ws/2008/06/identity/securitytokenservice`
4. Open the following file:
`<install_pack_dir>/domains/app_domain/wsit/wsit-client.xml`
5. Replace the placeholder in the `namespace` tag of the `wsit-client.xml` file with the `targetNamespace` value copied from the `STSService.xml` file.
6. Close and save the `wsit-client.xml` file to its original location.
 During the installation process, an `OSCM-wsit.jar` file is created containing the `STSService.xml` file as well as the `wsit-client.xml` file; the `.jar` file is then copied to `<install_pack_dir>/domains/app_domain/lib`.
7. Make sure that you enter correct values for the SAML_SP authentication mode in the `configsettings.properties` file in `<install_pack_dir>/databases/app_db`:
 - `BSS_AUTH_MODE=SAML_SP`
 - `BSS_STS_WEBSERVICE_URL=https://<server>:<port>/{SERVICE}/STS`
 - `BSS_STS_WEBSERVICE_WSDL_URL= https://<server>:<port>/oscm/v1.9/{SERVICE}/STS?wsdl`
 - `APP_KEYSTORE_PASSWORD=changeit`
 - `APP_TRUSTSTORE_PASSWORD=changeit`
8. In the `configsettings_controller.properties` located in `<install_pack_dir>/databases/app_db`, set the `BSS_USER_SSO` property to `true`.

2.2.3 Setting up the Databases

The vSphere integration software requires and stores its data in the `bssapp` PostgreSQL database. Another PostgreSQL database, `vmware`, is used to store the vSphere configuration for use by the VMware service controller. The vSphere configuration must be imported into the `vmware` database.

The databases are created by executing installation scripts (`build-db.xml` and `build-db-import.xml`). They are initialized with the appropriate schema and settings.

Proceed as follows:

1. Make sure that the database server is running.
2. Open the command prompt (Windows) or a terminal session (UNIX/Linux).
3. Set the following environment variable for your current session:
`DB_INTERPRETER`: The absolute path and name of the `psql` executable of PostgreSQL. The executable is usually located in the `bin` subdirectory of the PostgreSQL installation directory.

Example (Unix/Linux):

```
export DB_INTERPRETER="/opt/PostgreSQL/9.1/bin/psql"
```

Example (Windows):

```
set DB_INTERPRETER="C:\Program Files\PostgreSQL\9.1\bin\psql"
```

4. Create the databases by executing the `build-db.xml` file in `<install_pack_dir>/install` as follows:

```
<ANT_HOME>/bin/ant -f build-db.xml initDB
```

If you set an ID or password other than `postgres` for the PostgreSQL user account (`postgres`) when installing the database management system, you have to specify the ID or password with the call to the `build-db.xml` file as follows:

```
<ANT_HOME>/bin/ant -f build-db.xml initDB
-Ddb.admin.user=<user ID> -Ddb.admin.pwd=<password>
```

Note: It may be required to enclose the `-Ddb.admin.user=<user ID>` and `-Ddb.admin.pwd=<password>` in double or single quotes depending on the operating system.

5. Import the `.csv` files for the VMware service controller database into the controller database by executing the `build-db-import.xml` file in `<install_pack_dir>/install` as follows:

```
<ANT_HOME>/bin/ant -f build-db-import.xml importAll
-Ddb.dir=controller_db
```

Instead of importing all `.csv` file in one step, you can split the import by executing the `build-db-import.xml` file in the following order:

```
<ANT_HOME>/bin/ant -f build-db-import.xml importVCenter
-Ddb.dir=controller_db
<ANT_HOME>/bin/ant -f build-db-import.xml importCluster
-Ddb.dir=controller_db
<ANT_HOME>/bin/ant -f build-db-import.xml importVLAN
-Ddb.dir=controller_db
<ANT_HOME>/bin/ant -f build-db-import.xml importDomain
-Ddb.dir=controller_db
<ANT_HOME>/bin/ant -f build-db-import.xml importIPPool
-Ddb.dir=controller_db
```

Note: It may be required to enclose `-Ddb.dir=controller_db` in double or single quotes depending on the operating system.

If the setup of the databases fails with errors, proceed as follows:

1. Check and correct the configuration files.
2. Execute the `build-db.xml` file in `<install_pack_dir>/install` as follows:

```
<ANT_HOME>/bin/ant -f build-db.xml DROP.dbsAndUsers
```

3. Repeat the setup.

2.2.4 Setting up the Application Server Resources

The vSphere integration software requires specific settings and resources in the application server, such as mail settings or a data source.

Proceed as follows to create the resources and make the required settings in the application server:

1. Open the command prompt (Windows) or a terminal session (UNIX/Linux).
2. Execute the `build-glassfish.xml` file in `<install_pack_dir>/install` as follows:

```
<ANT_HOME>/bin/ant -f build-glassfish.xml SETUP
```

This has the following results:

- The `app-domain` domain is created and started.
 - The settings and resources for APP are created in the application server.
 - APP (`oscm-app.ear`) is deployed to the `app-domain` domain.
 - The VMware service controller (`oscm-app-vmware.ear`) is deployed to the `app-domain` domain.
 - A data source for the VMware service controller is created in the application server.
3. Depending on your environment, it may be required to define a proxy server for the `app-domain` domain in the **JVM Options** of the application server. The VMware service controller can address the corresponding system via the proxy server.

To define a proxy server, specify the following JVM options:

- `-Dhttps.proxyHost`
- `-Dhttps.proxyPort`

If authentication is required, specify the following additional settings:

- `-Dhttps.proxyUser`
- `-Dhttps.proxyPassword`

For all direct communication, you need to bypass the proxy server. Specify the hosts which are to be addressed directly and not through the proxy server in the following setting:

- `-Dhttp.nonProxyHosts`

For example, APP must not use the configured proxy for Web service calls to OSCM:

```
-Dhttp.nonProxyHosts=localhost|127.0.0.1|myServer*
```

where `myServer` is the host on which OSCM is running.

In case several service controllers are to run in the same APP domain, and only one of them is to communicate via a proxy server, you need to exclude the target systems of the other service controllers, for example, as follows:

```
-Dhttps.proxyHost=proxy.intern.myserver.com
-Dhttps.proxyPort=8081
-Dhttp.nonProxyHosts=myServer.com|localhost|127.0.0.1|
http://10.140.18.112*|http://myServer.com:8880/templates/*|
https://ror-demo.myServer.com:8014/cfmapi/endpoint*
```

After having configured the proxy server, restart the `app-domain`.

If the setup of the application server domain fails with errors, proceed as follows:

1. Stop the `app-domain` domain.

2. Delete the `app-domain` domain.
3. Repeat the setup.

2.2.5 Exchanging Certificates

For secure communication of the vSphere integration software with OSCM and the vSphere server, you need to exchange the corresponding certificates.

You need to:

- Import the certificates of OSCM and vSphere into the truststore of the `app-domain` application server domain of the vSphere integration software.
- Export the certificate of the `app-domain` domain and import it into the `bes-domain` application server domain of OSCM.

Proceed as follows:

1. Obtain a `.crt` file with the certificate of OSCM from the OSCM operator.

The `.crt` file can be created, for example, by executing the following command at the command prompt (Windows) or in a terminal session (UNIX/Linux) on the application server where OSCM is deployed:

```
<AppServerJRE>/bin/keytool -export -rfc -alias slas
-file ctmgbs.crt -storepass <password> -keystore
<GLASSFISH_HOME>/glassfish/domains/bes-domain/config/keystore.jks
```

2. Obtain a `.cer` file with the certificate of vSphere.

The `.cer` file can be created, for example, in the following way:

Open the URL of the vSphere API (<https://<server>:<port>/sdk/vimService>) in a Web browser. View the certificate of the page and export it to a file using the standard procedures of the Web browser.

3. Import the certificates of OSCM and vSphere into the truststore of the `app-domain` application server domain.

To import the OSCM certificate from the `.crt` file you created, you can use, for example, the following command at the command prompt (Windows) or in a terminal session (UNIX/Linux) on the application server:

```
<AppServerJRE>/bin/keytool -import -trustcacerts -alias <alias>
-file <filename>.crt -storepass <password> -keystore
<GLASSFISH_HOME>/glassfish/domains/app-domain/config/cacerts.jks
```

To import the vSphere certificate from the `.cer` file you created, you can use, for example, the following command at the command prompt (Windows) or in a terminal session (UNIX/Linux) on the application server:

```
<AppServerJRE>/bin/keytool -import -trustcacerts -alias <alias>
-file <filename>.cer -storepass <password> -keystore
<GLASSFISH_HOME>/glassfish/domains/app-domain/config/cacerts.jks
```

4. Create a `.crt` file with the certificate of the `app-domain` domain in which you have deployed the vSphere integration software.

The `.crt` file can be created, for example, by executing the following command at the command prompt (Windows) or in a terminal session (UNIX/Linux) on the application server:

```
<AppServerJRE>/bin/keytool -export -rfc -alias slas
-file ctmgapp.crt -storepass <password> -keystore
<GLASSFISH_HOME>/glassfish/domains/app-domain/config/keystore.jks
```

5. Import the certificate of the `app-domain` domain into the `bes-domain` application server domain of OSCM.

To do this, you can use, for example, the following command at the command prompt (Windows) or in a terminal session (UNIX/Linux) on the application server:

```
<AppServerJRE>/bin/keytool -import -trustcacerts -alias ctmgapp
-file ctmgapp.crt -storepass <password> -keystore
<GLASSFISH_HOME>/glassfish/domains/bes-domain/config/cacerts.jks
```

6. If the vSphere integration software and OSCM are configured for SAML_SP authentication mode, obtain the relevant certificates from the IdP system and import them into the truststore of the `app-domain` domain.

For example, when using Microsoft Active Directory as the IdP, you need to obtain and import the service communications and token-signing certificates.

7. Stop and restart the `app-domain` and the `bes-domain` domains for the certificates to become effective.

2.3 Installing the VMware Service Controller in an Existing APP Environment

If you already have a working APP installation in your environment, you can use it for the vSphere integration and simply register the VMware service controller in it. In addition, you need to setup the VMware service controller database. Proceed as follows:

1. Check the prerequisites described in *OSCM and vSphere* on page 10.
2. Specify the settings for the VMware service controller database setup and access in the `db.properties` file located in the `<install_pack_dir>/databases/controller_db` directory.
3. To set up the data source for the VMware service controller database in the application server, copy the following file:

```
<install_pack_dir>/domains/app_domain/installer/resources-extension-template.xml
to <install_pack_dir>/domains/app_domain/installer/resources-extension.xml
```
4. Edit the `resources-extension.xml` file and replace the following placeholders with the values defined in the `db.properties` file of the VMware service controller database. The corresponding property in the `db.properties` file is shown in parenthesis.
 - `@SERVER.NAME@ (db.host)`: The name of the host the VMware service controller database is running on. Example: `localhost`
 - `@PORT.NUMBER@ (db.port)`: The database port. Example: `5432`
 - `@DATABASE.NAME@ (db.name)`: The name of the database. Example: `vmware`
 - `@DB.USER@ (db.user)`: The name of the user to connect to the database. Example: `vmwareuser`

- @DB.PASSWORD@(db.pwd): The password of the user to connect to the database. Example: vmwareuser

5. Specify the vSphere configuration in the .csv files located in the <install_pack_dir>/databases/controller_db/csv directory.
6. Create the VMware service controller database (vmware) by executing the build-db.xml file in <install_pack_dir>/install as follows:

```
<ANT_HOME>/bin/ant -f build-db.xml CREATE.dbAndUser
-Ddb.dir=controller_db
```

Note: It may be required to enclose -Ddb.dir=controller_db in double or single quotes depending on the operating system.

7. Update the schema of the VMware service controller database (vmware) by executing the build-db.xml file in <install_pack_dir>/install as follows:

```
<ANT_HOME>/bin/ant -f build-db.xml UPDATE.dbSchema
-Ddb.dir=controller_db
```

Note: It may be required to enclose -Ddb.dir=controller_db in double or single quotes depending on the operating system.

8. Add the data source for the vmware database to the app-domain domain as follows:

```
<GLASSFISH_HOME>/bin/asadmin -p 8848 add-resources
<install_pack_dir>/domains/app_domain/installer/resources-extension.xml
```

9. Import the .csv files into the vmware database by executing the build-db-import.xml file in <install_pack_dir>/install as follows:

```
<ANT_HOME>/bin/ant -f build-db-import.xml importAll
-Ddb.dir=controller_db
```

Instead of importing all .csv files in one step, you can split the import by executing the build-db-import.xml file in the following order:

```
<ANT_HOME>/bin/ant -f build-db-import.xml importVCenter
-Ddb.dir=controller_db
<ANT_HOME>/bin/ant -f build-db-import.xml importCluster
-Ddb.dir=controller_db
<ANT_HOME>/bin/ant -f build-db-import.xml importVLAN
-Ddb.dir=controller_db
<ANT_HOME>/bin/ant -f build-db-import.xml importDomain
-Ddb.dir=controller_db
<ANT_HOME>/bin/ant -f build-db-import.xml importIPPool
-Ddb.dir=controller_db
```

Note: It may be required to enclose -Ddb.dir=controller_db in double or single quotes depending on the operating system.

Usually, you do not need to change the settings once they have been imported into the database.

The `vcenter.csv` file specifies the URL and user credentials for accessing the vSphere server. For security reasons, it is recommended that you delete the file as soon as you have successfully installed and configured the VMware integration software.

10. Deploy the VMware service controller to the `app-domain` domain. To do this, you use the GlassFish administration console, for example:

```
http://127.0.0.1:8848/
```

The `oscm-app-vmware.ear` file of the service controller is located in `<install_pack_dir>/domains/app_domain/applications`

11. Register the VMware service controller as follows in APP:

1. In a Web browser, access the base URL of APP, for example:

```
http://127.0.0.1:8880/oscm-app
```

2. Log in with the ID and password of the user and organization defined in the `configsettings.properties` file of APP (`BSS_USER_ID` and `BSS_USER_PWD`).
3. Specify the controller ID (`ess.vmware`) and the technology provider organization responsible for the VMware service controller.
4. Click **Save Configuration** to save the settings.

12. Configure the VMware service controller following the instructions in *Updating Configuration Settings* on page 31.

13. Import the certificate of vSphere into the truststore of the `app-domain` application server domain:

1. Obtain a `.cer` file with the certificate of vSphere. The `.cer` file can be created, for example, in the following way:

Open the URL of the vSphere API (`https://<server>:<port>/sdk/vimService`) in a Web browser. View the certificate of the page and export it to a file using the standard procedures of the Web browser.

2. Import the vSphere certificate from the `.cer` file you created, for example, by executing the following command at the command prompt (Windows) or in a terminal session (UNIX/Linux) on the application server:

```
<AppServerJRE>/bin/keytool -import -trustcacerts -alias <alias>
-file <filename>.cer -storepass <password> -keystore
<GLASSFISH_HOME>/glassfish/domains/app-domain/config/cacerts.jks
```

14. Stop and restart the `app-domain` and the `bes-domain` domains for the certificate to become effective.

2.4 Update Installation

Before updating your installation of the vSphere integration software, read the *Release Notes* of the new release. They contain information on compatibility issues, changes and enhancements, and known restrictions.

Preparing the Update

Before you start with the update installation, carry out the following steps:

1. Make sure that all provisioning operations are complete. Follow the steps as described in *Handling Problems in the Provisioning Process* on page 29.

2. In the `app-domain` application server domain, disable or undeploy the following applications:

```
oscm-app
oscm-app-vmware
```

3. Check for `.glassfishStaleFiles` files in the `app-domain` domain. If there are any, delete them. The files are located in

```
app-domain/applications/<application name>/.glassfishStaleFiles
```

For example:

```
app-domain/applications/oscm-app/.glassfishStaleFiles
```

4. Set the following environment variable for your current session:

`DB_INTERPRETER`: The absolute path and name of the `psql` executable of PostgreSQL. The executable is usually located in the `bin` subdirectory of the PostgreSQL installation directory.

Example:

```
export DB_INTERPRETER="/opt/PostgreSQL/9.1/bin/psql"
```

5. Make sure that the `resources-extension.xml` file is available in

```
<install_pack_dir>/domains/app_domain/installer
```

and that the settings for the VMware service controller database setup and access match the ones in defined in the following file:

```
<install_pack_dir>/databases/controller_db/db.properties
```

If this is not the case, proceed as follows:

1. Copy the following file:

```
<install_pack_dir>/domains/app_domain/installer/resources-extension-template.xml
to <install_pack_dir>/domains/app_domain/installer/resources-extension.xml
```

2. Edit the `resources-extension.xml` file and replace the following placeholders with the values defined in the `db.properties` file of the VMware service controller database. The corresponding property in the `db.properties` file is shown in parenthesis.

- `@SERVER.NAME@ (db.host)`: The name of the host the VMware service controller database is running on. Default: `localhost`
The permitted length of host names depends on the operating system.
- `@PORT.NUMBER@ (db.port)`: The database port. Default: `5432`
- `@DATABASE.NAME@ (db.name)`: The name of the database. Default: `vmware`
- `@DB.USER@ (db.user)`: The name of the user to connect to the database. Default: `vmwareuser`
- `@DB.PASSWORD@ (db.pwd)`: The password of the user to connect to the database. Default: `vmwareuser`

Updating the `bssapp` Database

Proceed with updating the `bssapp` database as follows:

1. Check whether the file

```
postgresql-9.1-903.jdbc4.jar
```

is contained in the following directories of the application server:

- lib directory of the app-domain domain
- <GLASSFISH_HOME>/mq/lib/ext

If it is not, copy the file from the <install_pack_dir>/install/lib directory to the location where it is missing.

2. Create a backup of the `bssapp` database using the standard PostgreSQL commands. The database backup must be compatible with PostgreSQL 9.1.12.
3. Update the following configuration files so that the settings match your current installation:
 - `db.properties`
 - `configsettings.properties`
 - `configsettings_controller.properties`
4. Update the schema and configuration settings of the `bssapp` database by executing the `build-db.xml` file in <install_pack_dir>/install as follows:

```
<ANT_HOME>/bin/ant -f build-db.xml updateDatabase
```

Note: Make sure that Ant runs in a Java 7 runtime environment when calling the `build-db.xml` file.

Updating the VMware Database

If you want to update the vSphere configuration stored in the VMware service controller database, proceed as follows:

1. Create a backup of the database using the standard PostgreSQL commands. The database backup must be compatible with PostgreSQL 9.1.12.
2. Update the following configuration files so that the settings match your current installation:
 - `db.properties`
 - `databases/controller_db/csv/*.csv`
3. Delete the existing database tables by executing the `build-db-import.xml` file in <install_pack_dir>/install as follows:

```
<ANT_HOME>\bin\ant -f build-db-import.xml deleteAll
-Ddb.dir=controller_db
```

4. To update the vSphere configuration in the VMware service controller database, import the `.csv` files into the database by executing the `build-db-import.xml` file in <install_pack_dir>/install as follows:

```
<ANT_HOME>/bin/ant -f build-db-import.xml importAll
-Ddb.dir=controller_db
```

Instead of importing all `.csv` files in one step, you can split the import by executing the `build-db-import.xml` file in the following order:

```
<ANT_HOME>/bin/ant -f build-db-import.xml importVCenter
-Ddb.dir=controller_db
<ANT_HOME>/bin/ant -f build-db-import.xml importCluster
-Ddb.dir=controller_db
```

```
<ANT_HOME>/bin/ant -f build-db-import.xml importVLAN
-Ddb.dir=controller_db
<ANT_HOME>/bin/ant -f build-db-import.xml importDomain
-Ddb.dir=controller_db
<ANT_HOME>/bin/ant -f build-db-import.xml importIPPool
-Ddb.dir=controller_db
```

Note: It may be required to enclose `-Ddb.dir=controller_db` in double or single quotes depending on the operating system.

Updating the Application Server

After you have executed the preparation steps, redeploy or deploy the applications that make up the vSphere integration software in the `app-domain` domain:

1. `oscm-app`
2. `oscm-app-vmware`
3. Update the data source for the `vmware` database in the `app-domain` domain as follows:

```
<GLASSFISH_HOME>/bin/asadmin -p 8848 add-resources
<install_pack_dir>/domains/app_domain/installer/resources-extension.xml
```

4. Restart the `app-domain` domain.

Updating the Configuration for SAML_SP Mode

If you are running OSCM and the vSphere integration software in SAML_SP mode, and if the IdP metadata of your SSO environment have changed, you need to update your WSIT files accordingly:

1. Extract the `OSCM-wsit.jar` file into a separate directory.

The `.jar` file is located in

```
<GLASSFISH_HOME>/domains/app_domain/lib
```

The `OSCM-wsit.jar` file contains the following files:

```
wsit-client.xml
STSService.xml
```

2. Adapt the `.xml` files as required by your environment. For details, refer to *Configuring the vSphere Integration* on page 13.
3. Recreate the `OSCM-wsit.jar` file with the modified contents.
4. Stop the `app-domain` domain.
5. Copy the adapted `OSCM-wsit.jar` to the following directory:

```
<GLASSFISH_HOME>/domains/app_domain/lib
```
6. Restart the `app-domain` domain.

Note: If, for some reason, you recreate the `app-domain` domain, you also need to recreate the `OSCM-wsit.jar` in the `<GLASSFISH_HOME>/domains/app_domain/lib` directory.

3 Creating and Publishing Services

The following sections describe how to create and publish services in OSCM by means of which customers can request and use VMs on a vSphere server.

3.1 Prerequisites and Preparation

The following prerequisites must be fulfilled before you can create and publish services in OSCM:

- To create technical services for the vSphere integration in OSCM, you must have access to OSCM as a technology manager. You must be a member of the technology provider organization responsible for the VMware service controller as specified in the configuration settings for the installation.
- In vSphere, appropriate templates for the provisioning of VMs must exist, to which the technical services in OSCM can be mapped. The vSphere user specified in the configuration settings for the installation must have the necessary permissions to create and configure VMs based on these templates.
- To create marketable services for the vSphere integration in OSCM, you must have access to OSCM as a service manager of an organization with the supplier role. This may be the same organization as the technology provider organization or a different one.
- To publish your marketable services, you must have access to an appropriate marketplace in OSCM in your service manager role.

3.2 Creating Technical Services

The first step in providing OSCM services for vSphere is to create one or more technical services. Proceed as follows:

1. Define one or more technical services in an XML file.

The vSphere integration package, `oscm-vmware-install-pack.zip`, includes a technical service as a sample. Use the sample as a basis for defining your own technical services as required:

```
samples/TechnicalService_VMware.xml
```

In the technical service definition, be sure to specify:

- The asynchronous provisioning type
- The direct access type
- Service parameters which represent the VM templates defined in vSphere as well as the naming, location, and sizing parameters for the VMs to be provisioned. Service parameters also define the network setup in vSphere and the configuration of the VMs. For details on the supported service parameters, refer to *Service Parameters and Operations* on page 48.

Note: Make sure that you do not specify the `baseUrl` attribute in the technical service definition XML file. It specifies an application's remote interface and is not needed for providing OSCM services for vSphere.

2. Log in to the OSCM administration portal with your technology manager account.
3. Import the technical services you created and appoint one or more supplier organizations for them.

For details on these steps, refer to the *Technology Provider's Guide* and to the online help of OSCM.

3.3 Creating and Publishing Marketable Services

As soon as the technical services for the vSphere integration exist in OSCM, you can define and publish marketable services based on them. Your cost calculation for the services should include any external costs for operating the VMs.

Proceed as follows:

1. Log in to the OSCM administration portal with your service manager account.
2. Define one or more marketable services based on the technical services you created for vSphere.
3. Define price models for your marketable services.
4. Publish the services to a marketplace.

For details on these steps, refer to the *Supplier's Guide* and to the online help of OSCM.

4 Using vSphere Services in OSCM

The following sections describe how users can subscribe to and work with the services you have created for vSphere in OSCM. You will find details of the supported usage scenarios outlined in *Usage Scenarios* on page 9.

4.1 Subscribing to Services

Users of customer organizations can subscribe to the services you have created for vSphere on the marketplace where you have published them. This results in the provisioning of a VM in vSphere, as defined in the underlying technical service.

Depending on the definition of the service, the name of the VM is either generated automatically, or the customer can enter a name for it. The technical service may specify a prefix which is prepended to this name, as well as a pattern against which the name is checked before the provisioning operation is started. The name is used as the host name of the provisioned VM.

Depending on the parameters defined for the technical service, the customer can select or enter the number of CPUs, amount of RAM, disk size, additional data disks, and network settings for the VM as well as the vCenter, data center, and cluster in which the VM is to be provisioned.

The provisioning operations are carried out in asynchronous mode. As long as the provisioning is not complete, the status of the subscription is **pending**. The status changes to **ready** as soon as the provisioning has been finished successfully.

As soon as the provisioning is complete, the users assigned to the subscription can access the VM provided by vSphere using the IP address and initial password indicated in the subscription details on the marketplace in OSCM. The initial user ID for accessing the VM is `Administrator` for Microsoft Windows systems, `root` for Linux systems. It is strongly recommended that the initial password is changed after the first login.

4.2 Modifying Subscription Parameters

By modifying the corresponding service parameters at their subscriptions, customers can change the VMs provisioned in vSphere.

The following scenario is supported by the vSphere integration software:

Vertical scale-up and scale-down. By changing the corresponding service parameters, customers can trigger vSphere to add resources (disks, RAM, CPUs) to a VM. In the same way, they can request vSphere to remove resources from the VM.

The modifications are carried out in asynchronous mode. As long as the update is not complete, the status of the corresponding subscription is **pending update**. The status changes to **ready** as soon as the modification has been finished successfully.

Note: If the subscription was **suspended** before starting the modification, its status changes to **suspended update** as long as the operation is not complete.

In the technical service definition, you can specify an address to which emails are to be sent that notify service users or administrators of customer organizations about a successful modification of a VM.

4.3 Executing Service Operations

Customers can explicitly start and stop a VM in vSphere from OSCM as well as create and restore snapshots. To do this, they execute the appropriate service operation from the subscription for the VM:

- **Start:** Starts the VM if it was stopped.
- **Stop:** Stops the VM if it was started.
- **Restart:** Shuts down the VM in vSphere and starts it again.
- **Snapshot:** Creates a snapshot of the VM. Be aware that no snapshot versions are stored. When a new snapshot is created, the previous one is overwritten.
- **Restore:** Resets the VM to a previously created snapshot.

4.4 Terminating Subscriptions

A customer can at any time terminate a subscription for a VM in vSphere.

vSphere is triggered to delete the VM. The subscription is terminated independent of whether the deletion is successful. Note, however, that the VM name cannot be re-used before the deletion has been completed in vSphere.

5 Administrating the vSphere Integration

The following sections describe administration tasks you may need to perform in your role as an operator of the vSphere integration software:

- *Controlling the Provisioning Process* on page 29
- *Handling Problems in the Provisioning Process* on page 29
- *Handling Communication Problems Between APP and OSCM* on page 30
- *Backup and Recovery* on page 31
- *Updating Configuration Settings* on page 31
- *Adapting the Log Configuration* on page 33

5.1 Controlling the Provisioning Process

The vSphere integration provides you with the following feature for controlling the provisioning, modification, and deprovisioning of VMs:

In the definition of the technical services for vSphere, you can specify the `MAIL_FOR_COMPLETION` parameter. This is an address to which emails are to be sent in case manual steps are required to complete an operation.

If you specify this parameter, the VMware service controller interrupts the processing of each operation before its completion and waits for a notification about the execution of a manual action. This notification consists in opening the link given in the email.

Omit the `MAIL_FOR_COMPLETION` parameter if you do not want to interrupt the processing.

5.2 Handling Problems in the Provisioning Process

If the provisioning of a VM fails on the vSphere side or if there are problems in the communication between the participating systems, the corresponding subscription in OSCM remains pending. The VMware service controller informs the technology managers of its responsible technology provider organization by email of any incomplete operation in vSphere.

You can then take the appropriate actions to solve the problem in vSphere or in the communication. For example, you could remove an incomplete VM, or you could restore a missing connection.

After solving the problem, the vSphere integration components and OSCM need to be synchronized accordingly. You do this by triggering a corresponding action in the APP component. Proceed as follows:

1. Work as a technology manager of the technology provider organization responsible for the VMware service controller.
2. Invoke the instance status interface of APP for the VMware service controller by opening the following URL in a Web browser:

```
https://<server>:<port>/oscm-app/controller/?cid=ess.vmware
```

For example:

```
https://127.0.0.1:8881/oscm-app/controller/?cid=ess.vmware
```

The Web page shows all subscriptions for VMs, including detailed information such as the customer organization, the ID of the related VM instance, and the provisioning status.

3. Find the subscription for which you solved the problem in the most recent provisioning, modification, or delete operation.

4. In the **Action** column, select the action for the vSphere integration components to execute next. Possible actions are the following:
 - **RESUME** - to resume the processing of a provisioning operation in APP which was suspended.
 - **SUSPEND** - to suspend the processing of a provisioning operation in APP, for example when vSphere does not respond.
 - **UNLOCK** - to remove the lock for a VM instance in APP.
 - **DELETE** - to terminate the subscription in OSCM and remove the instance in APP, but keep the VM in vSphere for later use. The service manager role is required for this action.
 - **DEPROVISION** - to terminate the subscription in OSCM, remove the instance in APP, and delete the VM in vSphere. The service manager role is required for this action.
 - **ABORT_PENDING** - to abort a pending provisioning or modification operation in OSCM. OSCM is notified to roll back the changes made for the subscription and return it to its previous state. In vSphere, no actions are carried out.
 - **COMPLETE_PENDING** - to complete a pending provisioning or modification operation in OSCM. OSCM is notified to complete the changes for the subscription and set the subscription status to **ready** (or **suspended** if it was suspended before). This is possible only if the operations of the service controller are already completed.
5. Click **Execute** to invoke the selected action.

The instance status interface provides the following additional functionality that is useful for problem-solving purposes:

- You can display service instance details for each subscription by clicking the corresponding entry in the table. This displays all subscription-related information that is stored in the `bssapp` database.
- The **Run with timer** column indicates whether the timer for the interval at which APP polls the status of instances is running. You can reset the timer, if required. For details on the timer setting, refer to *Configuration Settings* on page 36.

5.3 Handling Communication Problems Between APP and OSCM

When the communication between APP and OSCM is no longer possible, for example, because OSCM is stopped, APP suspends the processing of requests. An internal flag is set in the APP database: `APP_SUSPEND=true`, and an email is sent to the address specified in the `APP_ADMIN_MAIL_ADDRESS` configuration setting.

Contact the OSCM operator to make sure that OSCM is up and running again correctly.

You then have the following possibilities to resume the processing of requests by APP:

1. Click the link provided in the email.
2. Log in to APP.

APP is restarted instantly. In the APP database, the `APP_SUSPEND` key is set to `false`.

As an alternative, you can proceed as follows:

1. In a Web browser, access the base URL of APP, for example:

`http://127.0.0.1:8880/oscm-app`

2. Log in with the ID and password of the user and organization defined in the `configsettings.properties` file of APP (`BSS_USER_ID` and `BSS_USER_PWD`).

A message is shown that APP has been suspended due to a communication problem with OSCM.

3. Click **Restart**.

APP is restarted instantly. In the APP database, the `APP_SUSPEND` key is set to `false`.

5.4 Backup and Recovery

The vSphere integration software does not offer integrated backup and recovery mechanisms. Use the standard file system, application server, and database mechanisms instead.

Backup

It is recommended that you create a regular backup of the following data according to the general guidelines of your organization:

- Databases (`bssapp` and `vmware`). The frequency of database backups depends on the amount of changes and the availability of time slots with low load. PostgreSQL supports database backups without previous shutdown. For details, refer to the PostgreSQL documentation.
- Certificates contained in the truststore of the `app-domain` domain (`cacerts.jks` file).
- Configuration files.

Note: When preparing for an update installation of the current vSphere integration software, always create a backup of the data mentioned above.

Recovery

If you need to recover your vSphere integration installation, the recommended procedure is as follows:

1. Restore the `bssapp` and `vmware` databases from the backup using the relevant PostgreSQL commands.
2. Stop the `app-domain` domain of the application server.
3. Restore the certificate truststore of the `app-domain` domain (`cacerts.jks` file) from the backup.
4. Start the `app-domain` domain.

5.5 Updating Configuration Settings

The vSphere integration software and setup utilities require a number of settings. In the installation, you adapted the settings, in particular server names, ports, paths, and user IDs, to your environment.

The configuration settings are provided in the following subdirectories and files of

`<install_pack_dir>`:

- `databases/app_db`
 - `db.properties`: Settings for the APP database setup and access.
 - `configsettings.properties`: Configuration settings for APP.

The initial installation stores these settings in the `bssapp` database, where you can change them later, if required. An update installation overwrites the settings. If you don't want existing settings to be overwritten, delete them from the properties file. In case that

mandatory settings are missing in the properties file and not yet stored in the database, an exception will occur.

- `configsettings_controller.properties`: Configuration settings for the VMware service controller.

The initial installation stores these settings in the `bssapp` database, where you can change them later, if required, by means of an update installation. Refer to *Update Installation* on page 21 for details.

- **databases/controller_db**

- `db.properties`: Settings for the VMware service controller database setup and access.
- `csv/*: .csv` files for specifying the vSphere configuration.

The initial import stores the vSphere configuration in the `vmware` database. If you want to update the installation, you need to delete the configuration settings and import the `.csv` files again. Refer to *Update Installation* on page 21 for details.

- **domains/app_domain**

The configuration settings for setting up the application server domain to which APP is deployed are provided in the following file:

```
glassfish.properties
```

For details on the configuration settings, refer to *Configuration Settings* on page 36.

If you need to change the settings, proceed as described in the following sections.

To update the configuration settings for database access:

1. Log in to the administration console of the application server.
2. Adapt the settings as required.

To update the configuration settings for the application server:

1. Open the `glassfish.properties` file located in `<install_pack_dir>/domains/app_domain` with an editor.
2. Check the settings in the file and adapt them to your environment if required.
3. Save the file to its original location in `<install_pack_dir>/domains/app_domain`.
4. Update the settings and resources in the application server by executing the `build-glassfish.xml` file in `<install_pack_dir>/install` as follows:

```
<ANT_HOME>/bin/ant -f build-glassfish.xml
SETUP.configureDomains
```

To update the configuration settings for APP:

1. Edit the content of the `configsettings.properties` file as required.
2. Execute the `build-db.xml` file in `<install_pack_dir>/install` as follows:

```
<ANT_HOME>/bin/ant -f build-db.xml
UPDATE.configSettings
```

To update the settings of the vSphere configuration:

1. In a Web browser, access the URL of the VMware service controller, for example:
`http://127.0.0.1:8880/oscm-app-vmware/vcenter.jsf`.

2. Log in with the ID and password of the user specified in the configuration settings for the VMware service controller (`BSS_USER_ID` and `BSS_USER_PWD`) or as another technology manager registered for the same organization.
3. You can change the following settings for accessing the vCenter:
 - **Connection URL:** The URL for accessing the vSphere API.
 - **User ID:** The ID of the technical user in vSphere for accessing the vSphere API.
 - **Password:** The password of the technical user in vSphere for accessing the vSphere API.
4. Save the settings.

If you want to change the technology provider organization responsible for the VMware service controller, you can use the Web interface of APP:

1. In a Web browser, access the base URL of APP, for example:
`http://127.0.0.1:8880/oscm-app`
2. Log in with the ID and password of the user specified for `BSS_USER_KEY` in the configuration settings for APP or as another administrator of the same organization.
3. Specify the technology provider organization for the VMware service controller, `ess.vmware`.
4. Save the settings.

5.6 Adapting the Log Configuration

The vSphere integration software records information and problems such as connection issues in the following log files on the application server:

- `<GLASSFISH_HOME>/domains/<DOMAIN_NAME>/logs/app-vmware.log`: Log of the VMware service controller
- `<GLASSFISH_HOME>/domains/<DOMAIN_NAME>/logs/app-core.log`: Log of the APP component

The logging is based on `log4j`. The default log level is `INFO`, which may not be sufficient depending on the circumstances. In such a case, you will need to adapt the log level in the configuration files. The following configuration files are of relevance:

- `<GLASSFISH_HOME>/domains/<DOMAIN_NAME>/config/log4j.ess.vmware.properties`: Log configuration of the VMware service controller
- `<GLASSFISH_HOME>/domains/<DOMAIN_NAME>/config/log4j.app.core.properties`: Log configuration of the APP component

Proceed as follows to adapt the log level:

1. Open the relevant configuration file.
2. Find the string `log4j.logger.org.oscm.app` in the configuration file.
3. Change the log level as desired to one of the following:
 - `ERROR` - designates error events that might still allow the application to continue running.
 - `WARN` - designates potentially harmful situations.
 - `INFO` - designates informational messages that highlight the progress of the application at coarse-grained level.
 - `DEBUG` - designates fine-grained informational events that are most useful to debug an application.

Example:

```
log4j.logger.org.oscm.app=INFO
```

Every 60 seconds, the vSphere integration software checks for changes in the log configuration. There is no need to restart the application.

6 Uninstallation

If you want to uninstall the vSphere integration software, take the following preparations:

- Back up resources and data you would like to keep. For details, refer to *Backup and Recovery* on page 31.
- In OSCM, delete the marketable services and technical services related to vSphere.

To uninstall the vSphere integration software:

1. Stop the `app-domain` domain in the application server.
2. Delete the `app-domain` domain.
3. Delete the `bssapp` and `vmware` databases in the database management system.
4. Uninstall the database management system and the application server if you no longer need them for other purposes.

For details on how to proceed, refer to the documentation of the database management system and the application server.

Appendix A: Configuration Settings

The configuration settings for the vSphere integration software are provided in the following files in subfolders of the directory to which you extracted the `oscm-vmware-install-pack.zip` file (`<install_pack_dir>`):

- `domains/app_domain/glassfish.properties`
- `databases/app_db/db.properties`
- `databases/app_db/configsettings.properties`
- `databases/controller_db/db.properties`
- `databases/controller_db/csv/*.csv`
- `databases/app_db/configsettings_controller.properties`

This appendix describes the settings in detail.

A.1 GlassFish Configuration Settings

The `glassfish.properties` file located in `<install_pack_dir>/domains/app_domain` contains the configuration settings for the GlassFish application server. The settings are required for configuring the domain where APP is deployed.

Below you find a detailed description of the settings.

GLASSFISH_HOME

The absolute path and name of the GlassFish installation directory.

JDBC_DRIVER_JAR_NAME

The name of the PostgreSQL JDBC driver jar file as available after installation.

Example: `postgresql-9.1-903.jdbc4.jar`

MAIL_HOST

The host name or IP address of your mail server.

MAIL_RESPONSE_ADDRESS

The email address used by the server as the sender of emails.

Example: `saas@yourcompany.com`

MAIL_PORT

The port of your mail server.

Default: `25`

MAIL_USE_AUTHENTICATION

Optional. Defines whether mails can be sent only to users authenticated against the SMTP mail system.

Allowed values: `true`, `false`

Default: `false`

MAIL_USER

Mandatory if `MAIL_USE_AUTHENTICATION=true`. Specifies the name of the user to be used for authentication against the SMTP mail system.

MAIL_PWD

Mandatory if `MAIL_USE_AUTHENTICATION=true`. Specifies the password of the user to be used for authentication against the SMTP mail system.

MAIL_TIMEOUT

Optional. The time interval in milliseconds for sending email messages, i.e. until a socket I/O timeout occurs.

Allowed values: Any value between 0 and 4924967296

Default: 30000

MAIL_CONNECTIONTIMEOUT

Optional. The time interval in milliseconds for establishing the SMTP connection, i.e. until a socket connection timeout occurs.

Allowed values: Any value between 0 and 4924967296

Default: 30000

glassfish.domain.portbase

Mandatory. The base number for all ports used by the domain of the APP application.

Example: 8800

glassfish.domain.portadmin

The administration port of the domain used for APP.

Example: 8848

glassfish.domain.name

The name of the domain where APP is deployed.

Example: app-domain

glassfish.domain.admin.user

The user name of the APP domain administrator.

Default: admin

glassfish.domain.admin.pwd

The password of the APP domain administrator.

Default: adminadmin

glassfish.domain.admin.master.pwd

Mandatory. The master password required for accessing the keystore and truststore files of the application server domain.

Default: changeit

glassfish.domain.stop.waitSeconds

Mandatory. The time in seconds the application server waits until a stop domain operation is executed.

Default: 60

glassfish.domain.start.maxWaitSeconds

Mandatory. The maximum time in seconds the application server waits until it checks whether a domain is started.

Default: 600

A.2 APP Database Configuration Settings

The `db.properties` file located in `<install_pack_dir>/databases/app_db` contains the configuration settings for APP database access. This configuration is used for the initial setup and schema updates.

db.driver.class

The Java class of the JDBC driver.

Default: `org.postgresql.Driver`

db.host

The database host.

Default: `localhost`

db.port

The database port.

Default: 5432

db.name

The name of the database.

Default: `bssapp`

db.user

The name of the user to connect to the database.

Default: `bssappuser`

db.pwd

The password of the user to connect to the database.

Default: `bssappuser`

db.type

The type of the database.

Default: `postgresql`

A.3 APP Configuration Settings

The `configsettings.properties` file located in `<install_pack_dir>/databases/app_db` contains the configuration settings for APP.

APP_BASE_URL

`APP_BASE_URL=http://<server>:<port>/oscm-app`

The URL used to access APP.

APP_TIMER_INTERVAL

`APP_TIMER_INTERVAL=15000`

The interval (in milliseconds) at which APP polls the status of instances. If you increase the value, provisioning takes longer. If you decrease it, more load is put on the system. We strongly recommend that you do not set a value of more than 180000 milliseconds (3 minutes), although the maximum value is much higher (922337203685477580).

If you do not specify this setting at all, the default value used is 15000.

APP_MAIL_RESOURCE

`APP_MAIL_RESOURCE=mail/APPMail`

The JNDI name of the GlassFish mail resource used to send emails.

The resource `mail/APPMail` is created during setup with the parameters defined in the `glassfish.properties` file. This setting needs to be changed only if you want to use a different mail resource.

APP_ADMIN_MAIL_ADDRESS

`APP_ADMIN_MAIL_ADDRESS=admin@example.com`

The email address to which email notifications are sent.

APP_KEYSTORE_PASSWORD

`APP_KEYSTORE_PASSWORD=changeit`

The password required to access the keystore of the domain used for APP in the application server.

APP_TRUSTSTORE_PASSWORD

`APP_TRUSTSTORE_PASSWORD=changeit`

The password required to access the truststore of the domain used for APP in the application server.

BSS_AUTH_MODE

`BSS_AUTH_MODE=INTERNAL`

Specifies whether OSCM is used for user authentication or whether it acts as a SAML service provider and allows for single sign-on. The setting must be identical to the `AUTH_MODE` setting in OSCM.

Possible values: `INTERNAL` (OSCM user authentication is used) or `SAML_SP` (OSCM acts as a SAML service provider).

Contact the OSCM platform operator for details on which value to set.

BSS_USER_KEY

`BSS_USER_KEY=<userKey>`

The user key for accessing OSCM.

Replace `<userKey>` with the user key which you receive with the confirmation email for your user account.

The user specified here must have the administrator role for your organization in OSCM. The user account is used for carrying out actions on behalf of APP in OSCM. In addition, the user is allowed to register service controllers in APP.

BSS_USER_ID

`BSS_USER_ID=<userId>`

The identifier of the user specified in `BSS_USER_KEY` for accessing OSCM.

Replace `<userId>` with the user ID.

BSS_USER_PWD

`BSS_USER_PWD=_crypt:<password>`

The password of the user specified in `BSS_USER_KEY` for accessing OSCM.

Replace `<password>` with the plain text password. The password is encrypted when it is stored in the database.

BSS_WEBSERVICE_URL

`BSS_WEBSERVICE_URL=https://<server>:<port>/{SERVICE}/BASIC`

Mandatory when `BSS_AUTH_MODE` is set to `INTERNAL` and basic authentication is used. The endpoint of the OSCM Web services to be used. The `{SERVICE}` placeholder must not be replaced.

BSS_WEBSERVICE_WSDL_URL

`BSS_WEBSERVICE_WSDL_URL=https://<server>:<port>/oscm/v1.9/{SERVICE}/BASIC?wsdl`

Mandatory when `BSS_AUTH_MODE` is set to `INTERNAL` and basic authentication is used. The URL specifying the version of the OSCM Web services to be used. The `{SERVICE}` placeholder must not be replaced.

BSS_STS_WEBSERVICE_URL

`BSS_STS_WEBSERVICE_URL=https://<server>:<port>/{SERVICE}/STS`

Mandatory when `BSS_AUTH_MODE` is set to `SAML_SP` and security token based authentication is used. The endpoint of the OSCM Web services to be used. The `{SERVICE}` placeholder must not be replaced.

BSS_STS_WEBSERVICE_WSDL_URL

`BSS_STS_WEBSERVICE_WSDL_URL=https://<server>:<port>/oscm/v1.9/{SERVICE}/STS?wsdl`

Mandatory when `BSS_AUTH_MODE` is set to `SAML_SP`. The URL specifying the version of the OSCM Web services to be used. The `{SERVICE}` placeholder must not be replaced.

A.4 Controller Database Configuration Settings

The `db.properties` file located in `<install_pack_dir>/databases/controller_db` contains the configuration settings for access to the VMware service controller database. This configuration is used for the initial setup and schema updates.

db.driver.class

The Java class of the JDBC driver.

Default: `org.postgresql.Driver`

db.host

The database host.

Default: `localhost`

db.port

The database port.

Default: `5432`

db.name

The name of the database.

Default: `vmware`

db.user

The name of the user to connect to the database.

Default: `vmwareuser`

db.pwd

The password of the user to connect to the database.

Default: `vmwareuser`

db.type

The type of the database.

Default: `postgresql`

A.5 vSphere Configuration

The `.csv` files located in `<install_pack_dir>/databases/controller_db/csv` contain the complete configuration of vSphere for use by the VMware service controller. The `.csv` files must be filled with the vSphere configuration of your environment before importing them to the VMware service controller database (`vmware`). The files must be UTF-8 encoded.

In vSphere, several entities are distinguished. They form the following hierarchy:



One vCenter can manage many data centers. Each data center may contain any number of clusters. One cluster may use several VLANs (Virtual Local Area Networks), each having assigned several IP addresses.

You need to specify the vSphere configuration in the `.csv` files so that the correct entities are used for the provisioning of VMs in vSphere. The following files define the vSphere configuration. They map the hierarchy depicted above:

- `vcenter.csv`: Defines the vSphere installation.
- `datacenter.csv`: Defines one or several data centers of the vSphere installation.
- `cluster.csv`: Defines one or several clusters managed by a data center.
- `vlan.csv`: Defines one or several VLANs for a cluster.
- `ippool.csv`: Defines the list of IP addresses associated with a VLAN.

Below you find a description of the headers in the individual files below which you need to enter the appropriate information. The individual entries must be separated by commas, and be placed into one line without line breaks. If the entries contain a comma or blanks, you need to enclose them in double quotes.

vCenter

vCenter represents the vSphere installation. The `vcenter.csv` file contains the following headers:

| Header | Description |
|--------|--|
| TKey | The technical identifier of the vCenter in the VMware service controller database. It is used internally for identifying the vCenter. The key must be unique and consist of integers only. |
| Name | The name of the vCenter running on the vSphere server. |

| Header | Description |
|------------|--|
| Identifier | The unique identifier of the vCenter. If specified, this identifier can be used as a placeholder in the naming scheme of the VMs to be provisioned (<code>INSTANCENAME</code> service parameter). |
| URL | The URL for accessing the vSphere API. |
| UserId | The ID of the technical user in vSphere for accessing the vSphere API. |
| Password | The password of the technical user in vSphere for accessing the vSphere API. |

Example:

```
TKey,Name,Identifier,URL,UserId,Password
1,myvc1,1,https://myvc1.com/sdk/vimService?wsdl,user,pw123
```

Data Center

Data centers are managed by a vCenter. For each vCenter, the `datacenter.csv` file may contain several entries for several data centers with the following headers:

| Header | Description |
|--------------|--|
| VCenter | The name of the vCenter managing the data center. |
| Datacenter | The name of the data center. |
| DatacenterId | The unique identifier of the data center. The <code>\${DC}</code> placeholder in the <code>INSTANCENAME</code> service parameter of a technical service definition is replaced by this ID when vSphere instance names are generated. |

Example:

```
VCenter,Datacenter,DatacenterId
myvc1,datacenter1,1
myvc1,datacenter2,2
```

Clusters

One data center can contain many clusters. Each VM is provisioned in a cluster. For one vCenter and each data center managed by this vCenter, the `cluster.csv` may contain several entries for several clusters with the following headers:

| Header | Description |
|-------------|--|
| VCenter | The name of the vCenter managing the data center where the cluster is located. |
| Datacenter | The name of the data center the cluster is associated with. |
| Clustername | The name of the cluster. |

Example:

```

VCenter,Datacenter,Clustername
myvc1,datacenter1,cluster1
myvc1,datacenter1,cluster2
myvc1,datacenter2,cluster3

```

VLAN

VLANs (Virtual Local Area Networks) enable a single physical LAN segment to be further subdivided so that groups of ports are isolated from one another as if they were on physically separated segments. VLANs are associated with clusters. The `vlan.csv` file contains the following headers for defining VLANs:

| Header | Description |
|------------|--|
| VCenter | The name of the vCenter managing the data center where the cluster is located. |
| Datacenter | The name of the data center the cluster is associated with. |
| Cluster | The name of the cluster this VLAN is assigned to. |
| Name | The name of this VLAN. |
| Gateway | The IP address of the gateway of this VLAN. |
| SubnetMask | The subnet mask. |
| DNSServer | The DNS server for the VLAN. |
| DNSSuffix | The DNS suffix for the VLAN. The suffix is used in DNS name registration and resolution. |
| Enabled | In order to associate a VLAN with a VM and to activate the VLAN, you need to set this parameter to <code>true</code> . |

Example:

```

VCenter,Datacenter,Cluster,Name,Gateway,SubnetMask,
  DNSServer,DNSSuffix,Enabled
myvc1,datacenter1,cluster1,VLAN1,10.10.1.1,255.255.251.0,
  10.10.1.11,mycompany1.com,true
myvc1,datacenter1,cluster1,VLAN2,10.10.1.2,255.255.252.0,
  10.10.1.22,mycompany2.com,true
myvc1,datacenter1,cluster1,VM Network,10.10.1.3,255.255.253.0,
  10.10.1.33,mycompany3.com,true

```

The above example is not complete, and it contains line breaks due to text formatting reasons. You need to specify additional values for all clusters defined in the `cluster.csv` file.

IP Pool

An IP pool is a list of IP addresses associated with a VLAN. The `ippool.csv` file contains the following headers for defining IP addresses:

| Header | Description |
|------------|--|
| VCenter | The name of the vCenter managing the data center where the cluster is located. |
| Datacenter | The name of the data center the cluster is associated with. |
| Cluster | The name of the cluster to which this IP address is assigned. |
| VLAN | The name of the VLAN this IP address is assigned to. |
| IPAddress | The IP address. |

Example:

```
VCenter,Datacenter,Cluster,VLAN,IPAddress
myvc1,datacenter1,cluster1,VLAN1,169.254.10.1
myvc1,datacenter1,cluster1,VLAN1,169.254.10.2
myvc1,datacenter1,cluster1,VLAN1,169.254.10.3
myvc1,datacenter1,cluster1,VLAN2,169.255.10.1
myvc1,datacenter1,cluster1,VLAN2,169.255.10.2
myvc1,datacenter1,cluster1,VLAN2,169.255.10.3
```

The above example is not complete. You need to specify additional values for all VLANs defined in the `vlan.csv` file.

A.6 Controller Configuration Settings

The `configsettings_controller.properties` file located in `<install_pack_dir>/databases/app_db` contains the configuration settings for the VMware service controller. This configuration is used for the initial setup and stored in the APP database.

CONTROLLER_ID

```
CONTROLLER_ID=ess.vmware
```

The identifier of the service controller.

BSS_ORGANIZATION_ID

```
BSS_ORGANIZATION_ID=<organizationID>
```

The ID of the organization in OSCM which is responsible for the service controller. The organization must have the technology provider role.

BSS_USER_KEY

```
BSS_USER_KEY=<userKey>
```

The user key for accessing OSCM.

Replace `<userKey>` with the user key which you receive with the confirmation email for your user account.

The user specified here must have the technology manager role in OSCM and belong to the organization specified in the `BSS_ORGANIZATION_ID` setting.

It is recommended that the user account is used only for carrying out actions on behalf of the service controller in OSCM.

BSS_USER_ID

`BSS_USER_ID=<userId>`

The identifier of the user specified in `BSS_USER_KEY` for accessing OSCM.

Replace `<userId>` with the user ID.

BSS_USER_PWD

`BSS_USER_PWD=_crypt:<password>`

The password of the user specified in `BSS_USER_KEY` for accessing OSCM.

Replace `<password>` with the plain text password. The password is encrypted when it is stored in the database.

BSS_USER_SSO

`BSS_USER_SSO=true|false`

This setting tells the VMware service controller whether OSCM is running in `SAML_SP` mode. By default, this setting is set to `false`. If OSCM uses SSO authentication, you must change the value to `true`.

READY_TIMEOUT

`READY_TIMEOUT=<organizationID>`

The time interval in milliseconds the VMware service controller waits for the VM instance to accept the next command in the provisioning process. When the specified timeout value is reached, the provisioning is aborted.

Allowed values: Any value between 0 and 4924967296

Default: 300000

USERID_<organizationId>

`USERID_<organizationId>=<userId>`

Optional. The identifier of a service manager of a supplier organization providing a service on a OSCM marketplace for provisioning VMs. This identifier must be specified in case a shell script with service parameter placeholders is to be executed during the provisioning process.

Replace `<organizationId>` and the `<userId>` with the corresponding IDs.

USERKEY_<organizationId>

`USERKEY_<organizationId>=<userKey>`

Optional. The user key of the service manager specified in `USERID_<organizationID>`. This key must be specified in case a shell script with service parameter placeholders is to be executed during the provisioning process.

Replace `<organizationId>` and the `<userKey>` with the organization ID and user key, respectively.

USERPWD_<organizationId>

`USERPWD_<organizationId>=_crypt:<userPwd>`

Optional. The password of the service manager specified in `USERID_<organizationID>`. This password must be specified in case a shell script with service parameter placeholders is to be executed during the provisioning process.

Replace `<organizationId>` and the `<userPwd>` with the organization ID and password, respectively. The `_crypt:` prefix ensures that the password will be encrypted before it is stored.

Appendix B: Service Parameters and Operations

The following sections describe the technical service parameters and service operations which are supported by the VMware service controller.

General Parameters

The following parameters are required for provisioning a VM:

TEMPLATENAME

Mandatory. The name of the VM template to be used for the provisioning of a new VM instance. The template is a master copy of a VM that can be used to create many clones. When you provision a VM, you create a copy of the template, including its settings, configured virtual devices, installed software, and other contents of its disks.

Once specified, the template name can no longer be changed. Be aware that template names are case-sensitive.

Example: `centos52-template`

ACCESS_INFO

Optional. A pattern for the access information for a VM after it has been provisioned. The access information is sent by email to the subscribing user as well as all users assigned to the subscription. It is displayed in the details of the subscription.

The instance-specific values of this parameter contain the actual access information for the VM as soon as the subscription has been created.

The pattern may contain any textual expression. In addition, the following placeholders are supported:

- `${IP}`: IP address of the VM.
- `${HOST}`: Host name of the VM.
- `${CPU}`: CPU count.
- `${MEM}`: Main memory.
- `${DISK}`: System and data disk size(s).

Example: `${HOST} (IP: ${IP})
 ${CPU} CPUs, ${MEM} RAM, ${DISK} HDD`

Default: `${HOST}, ${IP}`

MAIL_FOR_COMPLETION

Optional. The address to which emails are to be sent in case manual steps are required to complete an operation. If you specify this parameter, the service controller interrupts the provisioning of the VM and waits for a notification about the execution of a manual action. Omit this parameter if you do not want to interrupt the provisioning process.

Example: `info@company.com`

VM Naming Rule Parameters

The following parameters define the naming rules for the VMs to be provisioned:

INSTANCENAME

Mandatory. The name of the VM instance to be provisioned. This name is either generated automatically, or must be specified by customers when they subscribe to a corresponding service. The instance name is used as the host name of the VM in the network configuration. The permitted length of host names depends on the operating system.

The string given in `INSTANCENAME_PREFIX` is prepended to the name. The name including the prefix must match the pattern given in `INSTANCENAME_PATTERN`.

For the automatic generation of instance names, the following placeholders are supported. They are filled with the corresponding data from the VMware service controller database when the VM is provisioned. Refer to *vSphere Configuration* on page 41 for details.

- `${VC}`: Unique ID of the vCenter derived from the `vcenter` database table. Specifies the vSphere server on which the VM is provisioned.
- `${DC}`: Unique ID of the data center derived from the `vcenter` database table.
- `${ID3}`: Unique sequential number of 3 digits.
- `${ID6}`: Unique sequential number of 6 digits.
- `${ID8}`: Unique sequential number of 8 digits.
- `${ID10}`: Unique sequential number of 10 digits.
- `${ID12}`: Unique sequential number of 12 digits.
- `${IP3}`: The last octet (eight bit) of the VM's IP address, for example, 152 of the 192.16.20.152 IP address.

It is helpful to add some hint about the naming rules as default value.

Example: `Hostname ([a-z0-9]){6,8}`

INSTANCENAME_PATTERN

Optional. A regular expression specifying a pattern for the VM instance names entered by the users when they subscribe to a corresponding service. This parameter is useful if the `INSTANCENAME` parameter is configurable by the user subscribing to the service, i.e. the instance names are not generated automatically. If the names do not match the pattern, the subscription is rejected. Special characters must not be used.

Default: `^[a-zA-Z0-9][a-zA-Z0-9-]{0,61}[a-zA-Z0-9]$`

The pattern may contain any textual expression.

Example: `IT([a-z0-9]){6,8}`

INSTANCENAME_PREFIX

Optional. A string to be prepended to the VM instance names when they are generated or entered by the users when they subscribe to a corresponding service. The combination of prefix and instance name must match the `INSTANCENAME_PATTERN` parameter.

Example: `IT`

VM Location Parameters

The following parameters can be specified for defining where on the vSphere server the VMs are to be provisioned. The parameters can be used by suppliers for offering a list of options to the customers subscribing to a corresponding service.

TARGET_VCENTER_SERVER

Optional. The site where vSphere is installed. vCenter is the centralized management tool for the vSphere suite. Together with the `TARGET_DATACENTER` and `TARGET_CLUSTER` parameters, this parameter is used to specify the location of the provisioned VM instances. The vCenter must be contained in the `vmware` database after the import of the `vcenter.csv` configuration file.

TARGET_DATACENTER

Optional. The data center where the VM instances will be provisioned. Together with the `TARGET_VCENTER_SERVER` and `TARGET_CLUSTER` parameters, this parameter is used to determine the location of the provisioned VM instances. The data center must be contained in the `vmware` database after the import of the `datacenter.csv` configuration file.

TARGET_CLUSTER

Optional. The cluster where the VM instances will be provisioned. A cluster is a group of hosts. Together with the `TARGET_VCENTER_SERVER` and `TARGET_DATACENTER` parameters, this parameter is used to determine the location of the provisioned VM instances. The cluster must be contained in the `vmware` database after the import of the `cluster.csv` configuration file.

TARGET_HOST

Optional. The name of the host on which the VM instances will be provisioned. If not specified, the default vSphere load balancing mechanism will determine the host machine to be used. In this case, the target host will be the one with the least VMs running on it.

TARGET_STORAGE

Optional. The name of the disk where the files of the VM instances are to be stored. If not specified, the vSphere load balancing mechanism will determine the target storage. In this case, the target storage will be the one with the highest amount of available disk space.

Example: `vmwdev1`

TARGET_FOLDER

Optional. The name of the virtual folder where the VM instances will be provisioned. If not specified, the VMs are placed into the folder where the VM template from which they were cloned is located.

Example: `folder_${ORGID}`

`${ORGID}` is a placeholder for the organization of the subscribing user.

The actual value will be, for example, `folder_83d79a`.

VM Sizing Parameters

The following parameters define the size of the VMs to be provisioned. Make sure that users can only specify numbers that are supported by the VM template and the configuration in vSphere. This can be done, for example, by presenting the supported numbers as different parameter options for selection.

NUMBER_OF_CPU

Mandatory. The number of virtual CPUs the VM instance to be provisioned will use. The number of licensed CPUs on the host, the number of CPUs that the guest operating system supports, and the VM hardware version determine the number of virtual CPUs that you can use.

Default: 1

AMOUNT_OF_RAM

Mandatory. The amount of memory in MB allocated to each VM.

Default: 1024

DISK_SIZE

Mandatory. The operating system disk size in GB. This parameter can be used to adjust the system disk size. If the parameter is not specified, the same disk size as specified in the VM template is used.

Currently, only one virtual device is supported. If multiple disks are defined in the VM template, the first one only will be configured.

Default: 40

DATA_DISK_SIZE_#

Optional. Up to 4 additional data disks can be defined: Replace the hash sign (#) with a digit between 1 and 4. For each disk, this parameter is used for specifying the disk size in GB. Make sure that when a user modifies subscription parameters, the value for the disk size can only be increased.

When adding data disks, additional actions are required on the operating system where the VMs are provisioned: To enable disk usage, you need to format and mount them. To do so, you could use a script and provide the script's URL in the `SCRIPT_URL` service parameter.

Example: `DATA_DISK_SIZE_1="40"`

DATA_DISK_TARGET_#

Optional. The mount point of the data disk with the given number (Linux), or the disk drive (Microsoft Windows).

Example: `DATA_DISK_TARGET_1="D"`

DATA_DISK_TARGET_VALIDATION_#

Optional. A regular expression specifying a fixed part of the mount point or disk drive where the additional data disk is to be mounted. This parameter must not be configurable by the user subscribing to the service.

Example: `~/home/datadisk/`

VM Network Parameters

The following parameters define the network for the VM instances to be provisioned:

NUMBER_OF_NICS

Mandatory. The number of network interface cards (NICs) which are to be used for new VMs. The number must match the number defined in the VM template. You can configure up to 4 networks for a VM.

Default: 1

NIC#_NETWORK_ADAPTER

Optional. The network adapter with the given number (#) which is to be used when a VM is provisioned. If specified, it replaces the network adapter defined in the VM template.

Example: `NIC1_NETWORK_ADAPTER="VLAN1"`

NIC#_NETWORK_SETTING

Optional. Determines how the IP address for the NIC with the given number (#) is set. The following options can be specified:

- **DHCP:** The DHCP protocol defines the IP address.
- **MANUAL:** The IP address is specified manually using the `NIC#_*` service parameters described below.
- **DATABASE:** The IP address is retrieved from the `vmware` database. The first free address is taken. The available IP addresses must have been defined in and imported from the `ippool.csv` configuration file.

Example: `NIC1_NETWORK_SETTING="DHCP"`

NIC#_IP_ADDRESS

Optional. The IP address for the NIC with the given number (#). Only useful when the `NIC#_NETWORK_SETTING` parameter is set to `MANUAL`.

Example: `NIC1_IP_ADDRESS="172.16.254.1"`

NIC#_SUBNET_MASK

Optional. The subnet mask for the NIC with the given number (#). Only useful when the `NIC#_NETWORK_SETTING` parameter is set to `MANUAL`.

Example: `NIC1_SUBNET_MASK="255.255.252.0"`

NIC#_GATEWAY

Optional. The gateway for the NIC with the given number (#). Only useful when the `NIC#_NETWORK_SETTING` parameter is set to `MANUAL`.

Example: `NIC1_GATEWAY="192.0.2.254"`

NIC#_DNS_SERVER

Optional. The DNS server for the NIC with the given number (#). Only useful when the `NIC#_NETWORK_SETTING` parameter is set to `MANUAL`.

Example: `NIC1_DNS_SERVER="208.67.222.222"`

NIC#_DNS_SUFFIX

Optional. The DNS suffix for the NIC with the given number (#). The suffix is used in DNS name registration and resolution. This policy lets you specify a suffix for a group of VMs, and prevents users from changing it.

Example: `NIC1_DNS_SUFFIX="example.com"`

VM Configuration Parameters

The following parameters define the configuration of the VMs to be provisioned:

SCRIPT_URL

Optional. The URL points to a shell script (Linux) or batch file (Microsoft Windows) that is downloaded from the given location, and uploaded and executed on a VM after the VM has been created and configured. You can use such a script or batch file, for example, for formatting and mounting additional data disks, or changing the root password. For security reasons, the `SCRIPT_URL` parameter must not be configurable by the user subscribing to the service.

Apart from the `SCRIPT_URL` parameter, the following service parameters must be specified:

- `SCRIPT_USERID`
- `SCRIPT_PWD`

Scripts or batch files can be provided, for example, in the `docroot` directory of the `app-domain` domain.

A script or batch file can contain commands or placeholders for the following service parameters:

- `LINUX_ROOT_PWD`
- `NIC#_DNS_SERVER` (# being a digit between 1 and 4)
- `NIC#_DNS_SUFFIX` (# being a digit between 1 and 4)
- `NIC#_GATEWAY` (# being a digit between 1 and 4)
- `NIC#_IP_ADDRESS` (# being a digit between 1 and 4)
- `NIC#_SUBNET_MASK` (# being a digit between 1 and 4)
- `INSTANCENAME`
- `SCRIPT_URL`
- `SCRIPT_USERID`
- `SCRIPT_PWD`
- `DOMAIN_NAME`
- `DATA_DISK_TARGET_#` (# being a digit between 1 and 4)
- `DATA_DISK_SIZE_#` (# being a digit between 1 and 4)
- `DATA_DISK_TARGET_VALIDATION_#` (# being a digit between 1 and 4)

If the script or batch file contains a placeholder for a service parameter, you must insert the user credentials of a service manager of a supplier organization as VMware service controller settings:

- `USERID_<organizationID>`
- `USERKEY_<organizationID>`
- `USERPWD_<organizationID>`

For details, refer to *Controller Configuration Settings* on page 45.

A sample script for changing the Linux root password is available in the `<install_pack_dir>/samples` directory: `changeRootPassword.sh`.

Example: `http://localhost:8080/changeNetworkSettings.sh`

SCRIPT_USERID

Optional. The user ID for executing the shell script (Linux) or batch file (Microsoft Windows) in the VM.

SCRIPT_PWD

Optional. The password for executing the shell script (Linux) or batch file (Microsoft Windows) in the VM.

DOMAIN_NAME

Optional. The domain part of the fully qualified domain name (FQDN) for Linux environments, or the Microsoft Windows domain that the VM will join if it is a Microsoft Windows system.

LINUX_ROOT_PWD

Optional. The root account password for Linux environments. Initially, a VM is provisioned with the user credentials derived from the VM template.

The root password can be changed only by executing a script using the `SCRIPT_URL` parameter. A sample of such a script is provided in the `<install_pack_dir>/samples` directory:

```
changeRootPassword.sh.
```

In the sample script, the `LINUX_ROOT_PWD` parameter contains the new Linux root password which is set when the script is executed. The sample script can be provided, for example, in the `docroot` directory of the `app-domain` domain.

WINDOWS_DOMAIN_JOIN

Optional. If set to `true`, the VM will join a Microsoft Windows domain, if set to `false`, the VM will be part of a Microsoft Windows Workgroup.

Default: `false`

WINDOWS_DOMAIN_ADMIN

Optional. The name of a Microsoft Windows domain administrator.

WINDOWS_DOMAIN_ADMIN_PWD

Optional. The password of the Microsoft Windows domain administrator.

WINDOWS_WORKGROUP

Optional. The Microsoft Windows Workgroup the VM will join, if specified by the `WINDOWS_DOMAIN_JOIN` parameter.

WINDOWS_LOCAL_ADMIN_PWD

Optional. The local administrator password of the VM.

WINDOWS_LICENSEKEY

Optional. The license key of the Microsoft Windows system.

SYSPREP_RUNONCE_COMMAND

Optional. The Microsoft Windows batch file that is executed by the Microsoft Windows system preparation script (`sysprep`) after the first boot.

Example: `cmd /c CALL \\server\share\file.bat`

Service Operations for VMs

The VMware service controller supports the service operations below for VMs.

The `actionURL` for each operation is:

`https://<host>:<port>/OperationService/AsynchronousOperationProxy?wsdl`

<host> and <port> are the server and port of the app-domain domain where the vSphere integration software is deployed.

START_VM

Starts a VM in vSphere if it was stopped.

STOP_VM

Stops a VM in vSphere if it was started.

RESTART_VM

Shuts down a VM in vSphere and starts it again.

SNAPSHOT_VM

Creates a snapshot of the VM. Be aware that no snapshot versions are stored. When a new snapshot is created, the previous one is overwritten.

RESTORE_VM

Restores a previously created snapshot of the VM.

Glossary

Administrator

A privileged user role within an organization with the permission to manage the organization's account and subscriptions as well as its users and their roles. Each organization has at least one administrator.

Application

A software, including procedures and documentation, which performs productive tasks for users.

Billing System

A system responsible for calculating the charges for using a service. OSCM comes with a native billing system, but can also be integrated with external ones.

Broker

An organization which supports suppliers in establishing relationships to customers by offering the suppliers' services on a marketplace, as well as a privileged user role within such an organization.

Cloud

A metaphor for the Internet and an abstraction of the underlying infrastructure it conceals.

Cloud Computing

The provisioning of dynamically scalable and often virtualized resources as a service over the Internet on a utility basis.

Customer

An organization which subscribes to one or more marketable services in OSCM in order to use the underlying applications in the Cloud.

Infrastructure as a Service (IaaS)

The delivery of computer infrastructure (typically a platform virtualization environment) as a service.

Marketable Service

A service offering to customers in OSCM, based on a technical service. A marketable service defines prices, conditions, and restrictions for using the underlying application.

Marketplace

A virtual platform for suppliers, brokers, and resellers in OSCM to provide their services to customers.

Marketplace Owner

An organization which holds a marketplace in OSCM, where one or more suppliers, brokers, or resellers can offer their marketable services.

Marketplace Manager

A privileged user role within a marketplace owner organization.

Operator

An organization or person responsible for maintaining and operating OSCM.

Organization

An organization typically represents a company, but it may also stand for a department of a company or a single person. An organization has a unique account and ID, and is assigned one or more of the following roles: technology provider, supplier, customer, broker, reseller, marketplace owner, operator.

Organizational Unit

A set of one or more users within an organization representing, for example, a department in a company, an individual project, a cost center, or a single person. A user may be assigned to one or more organizational units.

OU Administrator

A privileged user role within an organization allowing a user to manage the organizational units for which he has been appointed as an administrator, and to create, modify, and terminate subscriptions for these units.

Payment Service Provider (PSP)

A company that offers suppliers or resellers online services for accepting electronic payments by a variety of payment methods including credit card or bank-based payments such as direct debit or bank transfer. Suppliers and resellers can use the services of a PSP for the creation of invoices and payment collection.

Payment Type

A specification of how a customer may pay for the usage of his subscriptions. The operator defines the payment types available in OSCM; the supplier or reseller determines which payment types are offered to his customers, for example payment on receipt of invoice, direct debit, or credit card.

Platform as a Service (PaaS)

The delivery of a computing platform and solution stack as a service.

Price Model

A specification for a marketable service defining whether and how much customers subscribing to the service will be charged for the subscription as such, each user assigned to the subscription, specific events, or parameters and their options.

Reseller

An organization which offers services defined by suppliers to customers applying its own terms and conditions, as well as a privileged user role within such an organization.

Role

A collection of authorities that control which actions can be carried out by an organization or user to whom the role is assigned.

Seller

Collective term for supplier, broker, and reseller organizations.

Service

Generally, a discretely defined set of contiguous or autonomous business or technical functionality, for example an infrastructure or Web service. OSCM distinguishes between technical services and marketable services, and uses the term "service" as a synonym for "marketable service".

Service Manager

A privileged user role within a supplier organization.

Standard User

A non-privileged user role within an organization.

Software as a Service (SaaS)

A model of software deployment where a provider licenses an application to customers for use as a service on demand.

Subscription

An agreement registered by a customer for a marketable service in OSCM. By subscribing to a service, the customer is given access to the underlying application under the conditions defined in the marketable service.

Subscription Manager

A privileged user role within an organization with the permission to create and manage his own subscriptions.

Supplier

An organization which defines marketable services in OSCM for offering applications provisioned by technology providers to customers.

Technical Service

The representation of an application in OSCM. A technical service describes parameters and interfaces of the underlying application and is the basis for one or more marketable services.

Technology Manager

A privileged user role within a technology provider organization.

Technology Provider

An organization which provisions applications as technical services in OSCM.