

**V17.6.0**



Open Service  
Catalog Manager

# **OpenStack Integration**

December 2017 - Initial draft

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

Apache Ant, Ant, Apache Tomcat, Tomcat, 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 2017

## 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

	<b>About this Manual.....</b>	<b>4</b>
<b>1</b>	<b>Introduction.....</b>	<b>6</b>
1.1	Components Involved in the OpenStack Integration.....	7
1.2	Usage Scenarios.....	8
<b>2</b>	<b>Creating and Publishing Services.....</b>	<b>9</b>
2.1	Prerequisites and Preparation.....	9
2.2	Creating Technical Services.....	9
2.3	Creating and Publishing Marketable Services.....	10
<b>3</b>	<b>Using OpenStack Services in OSCM.....</b>	<b>11</b>
3.1	Subscribing to Services.....	11
3.2	Executing Service Operations.....	11
3.3	Terminating Subscriptions.....	12
<b>4</b>	<b>Administrating the OpenStack Integration.....</b>	<b>13</b>
4.1	Controlling the Provisioning Process.....	13
4.2	Handling Problems in the Provisioning Process.....	13
4.3	Handling Communication Problems Between APP and OSCM.....	14
4.4	Backup and Recovery.....	15
4.5	Updating Configuration Settings.....	16
4.6	Providing Templates for the Provisioning of Resources.....	18
4.7	Adapting the Log Configuration.....	18
<b>5</b>	<b>Uninstallation.....</b>	<b>20</b>
	<b>Appendix A Controller Configuration Settings.....</b>	<b>21</b>
	<b>Appendix B Service Parameters and Operations.....</b>	<b>24</b>
	<b>Glossary .....</b>	<b>28</b>

---

## About this Manual

This manual describes the integration of OpenStack, an open-source cloud operating system, with Open Service Catalog Manager (OSCM) .

This manual is structured as follows:

Chapter	Description
<i>Introduction</i> on page 6	Provides an overview of the OSCM OpenStack integration, the components involved, and the supported usage scenarios.
<i>Creating and Publishing Services</i> on page 9	Describes how to create and publish services for OpenStack in OSCM.
<i>Using OpenStack Services in OSCM</i> on page 11	Describes how to provision and deprovision virtual systems in OpenStack through services in OSCM.
<i>Administering the OpenStack Integration</i> on page 13	Describes administration tasks related to the OSCM OpenStack integration software.

The descriptions of services and usage scenarios also apply to FUJITSU Cloud Service K5, a cloud platform based on OpenStack and integrated with OSCM.

## Readers of this Manual

This manual is intended for technology managers and providers who want to offer virtual systems controlled by OpenStack through services on a marketplace provided by OSCM. It assumes that you have access to an existing OSCM installation and an OpenStack Web server. In addition, you should have basic knowledge of OpenStack and you should be familiar with the concepts and administration of OSCM.

## Notational Conventions

This manual uses the following notational conventions:

<b>Add</b>	Names of graphical user interface elements.
<code>init</code>	System names, for example command names and text that is entered from the keyboard.
<code>&lt;variable&gt;</code>	Variables for which values must be entered.
<code>[option]</code>	Optional items, for example optional command parameters.
<code>one   two</code>	Alternative entries.
<code>{one   two}</code>	Mandatory entries with alternatives.

## Abbreviations

This manual uses the following abbreviations:

**APP** Asynchronous Provisioning Platform

<b>DBMS</b>	Database Management System
<b>IaaS</b>	Infrastructure as a Service
<b>IdP</b>	SAML Identity Provider
<b>OSCM</b>	Open Service Catalog Manager
<b>SAML</b>	Security Assertion Markup Language
<b>STS</b>	Security Token Service
<b>WSDL</b>	Web Services Description Language
<b>WSIT</b>	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.
- *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.
- *OpenStack Integration*: A PDF manual for operators describing how to offer and use virtual systems controlled by OpenStack through services in OSCM.
- *Amazon Web Services Integration*: 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.
- *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.

# 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.

When integrating applications with OSCM, the instance provisioning can be done in two provisioning modes: synchronous or asynchronous mode.

Asynchronous provisioning is required if provisioning operations take a long time because long-running processes or manual steps are involved. This is the case, for example, when provisioning virtual machines on a virtual machine server. OSCM supports the integration of such applications with its asynchronous provisioning platform (APP). This is a framework which provides a provisioning service as well as functions, data persistence, and notification features which are always required for integrating applications in asynchronous mode.

APP also includes the operation service interface for executing technical service operations on the integrated applications from the OSCM user interface.

When deploying OSCM, a service controller for communication with the application is preconfigured and registered. This service controller can be used for integrating OpenStack services with OSCM.

OpenStack is an open-source cloud operating system that controls large pools of processing, storage, and networking resources throughout a data center. It is managed by the OpenStack Foundation.

OpenStack allows users to deploy virtual systems which handle different tasks for managing a cloud environment. It makes horizontal scaling easy, which means that tasks which benefit from running concurrently can easily serve users by spinning up more resources. For example, a mobile application which needs to communicate with a remote server might be able to divide the work of communicating with each user across many different resources, all communicating with one another but scaling quickly and easily as the application gains more users.

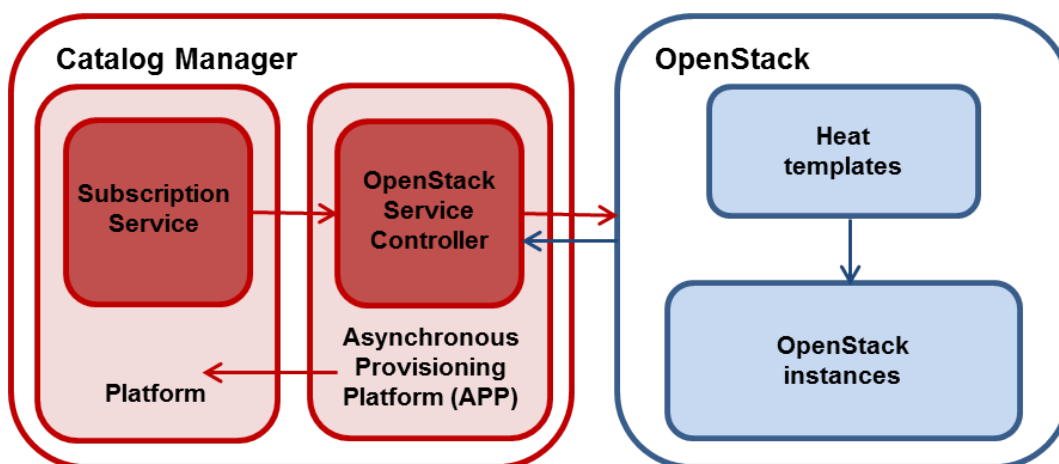
The integration of OpenStack with OSCM provides for an Infrastructure as a Service (IaaS) solution that leverages the features of both products: Through services, which are published on a marketplace in OSCM, users can request and use virtual systems in OpenStack. The usage costs can be calculated and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

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

FUJITSU Cloud Service K5 is a cloud platform based on OpenStack and integrated with OSCM. The descriptions of services and usage scenarios also apply to this platform.

## 1.1 Components Involved in the OpenStack Integration

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



In OSCM, customer subscriptions are managed by means of the **Subscription service**. When a customer creates or terminates a subscription for a virtual system in OpenStack, the Subscription service asynchronously triggers the corresponding actions in OpenStack through the **Asynchronous Provisioning Platform (APP)** and the **OpenStack service controller**: Virtual systems are created or deleted in OpenStack.

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 OpenStack service controller are the main components that make up the OpenStack integration software.

Heat, the orchestration component provided by OpenStack, offers template-based mechanisms for describing cloud applications. Through both an OpenStack REST API and a CloudFormation-compatible Query API, Heat provides compatibility with existing template formats. The integration of OpenStack with OSCM supports the AWS CloudFormation Template and the Heat Orchestration Template.

A template enables application developers to describe and automate the deployment of infrastructure, services, and applications. Collections of resources (e.g. networks, servers, or storage) can be deployed from a single template. The template serves as an orchestration document that details everything needed to carry out the orchestration. Once instantiated, the resources are also referred to as stacks.

Each APP installation supports one OpenStack service controller. This limitation can be overcome by installing APP several times to different application server domains. The need for more than one service controller may arise because multiple OpenStack accounts or technology provider organizations have to be used.

## 1.2 Usage Scenarios

The OSCM OpenStack integration supports the following usage scenarios:

- **Provisioning of a virtual system:** When a customer subscribes to a corresponding service on an OSCM marketplace, the service controller triggers OpenStack to create an OpenStack instance based on a specific template.
- **Suspending and resuming a virtual system:** A customer can explicitly suspend and resume an OpenStack instance by executing a service operation at the corresponding subscription.
- **Starting and stopping servers:** A customer can explicitly start and stop all the servers in an OpenStack instance by executing a service operation at the corresponding subscription.
- **Deletion of a virtual system:** When a customer terminates a subscription for an OpenStack instance, the service controller triggers OpenStack to delete the instance. The subscription is terminated in OSCM independent of whether the deletion is successful in OpenStack.

In OpenStack, the virtual systems created for OSCM subscriptions are managed in the same way as other virtual systems. They can be viewed and monitored with the available OpenStack tools.

Modifying a subscription and thereby triggering modifications of the virtual system in OpenStack is not supported. For more details on the supported usage scenarios, refer to *Using OpenStack Services in OSCM* on page 11.



## 2 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 virtual systems in OpenStack.

### 2.1 Prerequisites and Preparation

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

- To create technical services for the OpenStack 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 OpenStack service controller as specified in the configuration settings for the installation.
- Templates for the virtual systems to be provisioned must exist. They form the basis for the technical services in OSCM.

Templates can be provided in one of the following ways:

1. The technology provider organization responsible for the OpenStack service controller provides them on an external host in a location whose URL can be reached from OSCM via HTTP or HTTPS.
2. The OSCM operator provides them on the OSCM host in a location whose URL can be reached from OSCM via HTTP or HTTPS.
3. The technology provider organization responsible for the OpenStack service controller imports the templates into the `bssapp` database.

Refer to *Providing Templates for the Provisioning of Resources* on page 17 for details.

- The user specified in the configuration settings for the installation must have the necessary credentials to create and configure virtual systems for your organization based on these templates in OpenStack.

In FUJITSU Cloud Service K5, the `cpf_systemowner` role is required.

- To create marketable services for the OpenStack 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.

### 2.2 Creating Technical Services

The first step in providing OSCM services for OpenStack is to create one or more technical services.

Proceed as follows:

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

The OpenStack integration package, `oscm-openstack-install-pack.zip`, includes technical services as samples:

- `samples/TechnicalService_OpenStack.xml` for native OpenStack
- `samples/TechnicalService_K5.xml` for FUJITSU Cloud Service K5

The technical services specify sample templates as parameter options. The sample templates are also included in the `oscm-openstack-install-pack.zip` file:

- `samples/template.json` for a template of type AWS CloudFormation for native OpenStack

- `samples/template.yaml` for a template of type Heat Orchestration Template for native OpenStack
- `samples/template_k5.yaml` for a template of type Heat Orchestration Template for FUJITSU Cloud Service K5

You can use the samples to see how templates are mapped to technical service definitions. To use the sample templates, make them available in one of the ways described in *Providing Templates for the Provisioning of Resources* on page 17.

In the technical service definition, be sure to specify:

- The asynchronous provisioning type
- The direct access type
- Service parameters which correspond to the parameters specified in the template defined in OpenStack. For details on the supported service parameters, refer to *Service Parameters and Operations* on page 24.

<b>Note:</b> Make sure that you do not specify the <code>baseUrl</code> attribute in the technical service definition XML file. It specifies an application's remote interface and is not needed for providing OSCM services for OpenStack.
---

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.

## 2.3 Creating and Publishing Marketable Services

As soon as the technical services for the OpenStack 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 virtual systems.

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 OpenStack.
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.

## 3 Using OpenStack Services in OSCM

The following sections describe how users can subscribe to and work with the services you have created for OpenStack in OSCM. You will find details of the supported usage scenarios outlined in *Usage Scenarios* on page 7.

### 3.1 Subscribing to Services

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

To enable the provisioning of an OpenStack instance, the customer has to enter the name of the key pair of the virtual system when subscribing to the corresponding service in OSCM. The key pair name and the associated private key are used to securely access the OpenStack instance. For details on creating key pairs, refer to the user documentation of OpenStack.

When subscribing to a service, the customer also has to enter a name for the stack to be instantiated. Before the provisioning operation is started, the name is checked against the OpenStack conventions or a pattern specified in the technical service definition. OpenStack generates a random number that is appended to this stack name to make it unique.

Depending on the service parameters, the technical service either maps to one specific template or the customer can choose from different templates that specify the resource orchestration in OpenStack.

Depending on the parameters defined for the technical service, the customer can additionally choose from different options to configure the resources for the instance to be provisioned, for example, the instance type, the number of CPUs, or the storage size.

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 virtual system provided by OpenStack using the access information indicated in the subscription details on the marketplace in OSCM.

In OpenStack, the virtual systems created for OSCM subscriptions are managed in the same way as other instances. They can be viewed and monitored with the standard OpenStack tools. Changes, however, should not be made as this may cause problems and inconsistencies between OpenStack and OSCM.

### 3.2 Executing Service Operations

Customers can explicitly start and stop servers and suspend and resume instances in OpenStack from OSCM. To do this, they execute the appropriate service operation from the subscription for the virtual system:

- **Start:** Starts all servers in the OpenStack instance that were stopped.
- **Stop:** Stops all servers in the OpenStack instance that were started.
- **Suspend:** Suspends the OpenStack instance.
- **Resume:** Resumes the OpenStack instance if it was suspended.

As a prerequisite, the service operations must be defined in the technical service underlying the subscribed service.

Customers can view the status of all operations they started in their account information on the OSCM marketplace.

### **3.3 Terminating Subscriptions**

A customer can at any time terminate a subscription for a virtual system in OpenStack.

OpenStack is triggered to delete the virtual system. The subscription is terminated in OSCM independent of whether the deletion is successful in OpenStack.

## 4 Administrating the OpenStack Integration

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

- *Controlling the Provisioning Process* on page 13
- *Handling Problems in the Provisioning Process* on page 13
- *Handling Communication Problems Between APP and OSCM* on page 14
- *Backup and Recovery* on page 15
- *Updating Configuration Settings* on page 16
- *Providing Templates for the Provisioning of Resources* on page 17
- *Adapting the Log Configuration* on page 18

### 4.1 Controlling the Provisioning Process

The OpenStack integration provides you with the following feature for controlling the provisioning and deprovisioning of virtual systems:

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

If you specify this parameter, the OpenStack 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.

### 4.2 Handling Problems in the Provisioning Process

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

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

After solving the problem, the OpenStack 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 OpenStack service controller.
2. Invoke the instance status interface of APP for the OpenStack service controller by opening the following URL in a Web browser:

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

For example:

```
https://127.0.0.1:8881/oscm-app/controller/?controllerid=ess.openstack
```

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

3. Find the subscription for which you solved the problem in the most recent provisioning or delete operation.
4. In the **Action** column, select the action for the OpenStack 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 OpenStack does not respond.
  - `UNLOCK` - to remove the lock for an OpenStack instance in APP.
  - `DELETE` - to terminate the subscription in OSCM and remove the instance in APP, but keep the virtual system in OpenStack 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 virtual system in OpenStack. The service manager role is required for this action.
  - `ABORT_PENDING` - to abort a pending provisioning operation in OSCM. OSCM is notified to roll back the changes made for the subscription and return it to its previous state. In OpenStack, no actions are carried out.
  - `COMPLETE_PENDING` - to complete a pending provisioning 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*.

## 4.3 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`.

## 4.4 Backup and Recovery

The OpenStack 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:

- Database (`bssapp`). 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.

Make sure to also make a backup of the file containing the key required for encryption and decryption of service parameters with data type `PWD` and custom attributes marked for encryption. By default, this file is named `key` and located in:

`<GLASSFISH_HOME>/glassfish/domains/app-domain/config`

The location of this file can be changed using the `APP_KEY_PATH` configuration setting.

- 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 OpenStack integration software, always create a backup of the data mentioned above.

### Recovery

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

1. Restore the `bssapp` database from the backup using the relevant PostgreSQL commands.
2. Make sure that the file containing the key required for encryption and decryption of service parameters with data type `PWD` and custom attributes marked for encryption exists in the location specified in the `APP_KEY_PATH` configuration setting. By default, this file is to be named `key` and located in:

`<GLASSFISH_HOME>/glassfish/domains/app-domain/config`

If the file is missing, copy it from your backup to the correct location.

3. Stop the `app-domain` domain of the application server.

4. Restore the certificate truststore of the `app-domain` domain (`cacerts.jks` file) from the backup.
5. Start the `app-domain` domain.

## 4.5 Updating Configuration Settings

The OpenStack 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 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 OpenStack service controller.  
The initial installation stores these settings in the `bssapp` database. You can change them later using a graphical user interface.
- **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*.

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, you have the following options:

- Run the installation script:
  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
```

- 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 you specified for `BSS_USER_KEY` in the APP *configuration settings* or as another administrator of the same organization.
  3. Edit the settings as required.
  4. Click **Save Configuration** to save the settings.

To update the configuration settings for the OpenStack service controller:

1. In a Web browser, access the URL of the OpenStack service controller, for example:  
`http://127.0.0.1:8880/oscm-app-openstack.`
2. Log in with the ID and password of the user specified in the configuration settings for the OpenStack service controller (`BSS_USER_ID` and `BSS_USER_PWD`) or as another technology manager registered for the same organization.
3. Enter the required settings. Be aware that some of them can also be set in technical service definitions. In this case, the technical service parameter overrides the setting you specify here. Refer to *Controller Configuration Settings* on page 21 and *Service Parameters and Operations* on page 24 for details on the settings and parameters.

Apart from updating the configuration settings, you can import templates for the provisioning of VMs into the `bssapp` database. Refer to *Providing Templates for the Provisioning of Resources* on page 17 for details.

4. Save the settings.

If you want to **change the technology provider** organization responsible for the OpenStack 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 OpenStack service controller,  
`ess.openstack.`
4. Save the settings.
5. Make sure that the configuration settings for the OpenStack service controller are updated.

Any technology manager registered for the technology provider organization you specified can log in to the graphical user interface for updating the controller configuration settings (see above). At least the ID and password of the user to be used for accessing OSCM must be changed in the controller configuration settings.

## 4.6 Providing Templates for the Provisioning of Resources

There are the following ways to provide templates for the provisioning of virtual resources. These templates can be referenced in technical service definitions.

**To provide templates:**

- Copy the template files to a location of your choice that can be reached from OSCM by HTTP or HTTPS. For example, you could copy them to the `docroot/templates` directory of the `app-domain` domain in your application server.

Specify this location in the `TEMPLATE_BASE_URL` controller configuration setting.

- Import the template files into the `bssapp` database:
  1. In a Web browser, access the URL of the OpenStack service controller, for example:  
`http://127.0.0.1:8880/oscm-app-openstack`
  2. Log in with the ID and password of the user specified in the configuration settings for the OpenStack service controller (`BSS_USER_ID` and `BSS_USER_PWD`) or as another technology manager registered for the same organization.
  3. Click in the **File name** field or click **Browse**, and select the file containing the desired template.
  4. Click **Import**.

If a template with the same name is imported twice, the original (first) one is replaced by the second one.

The OpenStack integration software looks for the template with the name specified in the technical service definition (`TEMPLATE_NAME` service parameter) as follows:

1. The template is searched for in the `bssapp` database.
2. If it is not found, the template is searched for in the location specified in the `TEMPLATE_BASE_URL` controller configuration setting.

## 4.7 Adapting the Log Configuration

The OpenStack 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-openstack.log`: Log of the OpenStack 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.openstack.properties`: Log configuration of the OpenStack 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 OpenStack integration software checks for changes in the log configuration. There is no need to restart the application.

## 5 Uninstallation

If you want to uninstall the OpenStack 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 15.
- In OSCM, delete the marketable services and technical services related to OpenStack.

**To uninstall the OpenStack integration software:**

1. Stop the `app-domain` domain in the application server.
2. Delete the `app-domain` domain.
3. Delete the `bssapp` database 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: Controller Configuration Settings

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

A technology provider can define service parameters in the technical service definition. If such a parameter has the same ID as a controller configuration setting stored in the APP database, it overrides the configuration setting in the database when the marketable service based on such a technical service is subscribed to. By default the values in the controller configuration settings are used. Refer to the *Technology Provider's Guide* for details on defining technical services.

In addition, a supplier can define custom attributes for subscriptions and for customers. If such an attribute has the same ID as a controller configuration setting stored in the APP database as well as a corresponding technical service parameter, it overrides the technical service parameter as well as the configuration setting in the database when the marketable service based on such a technical service is subscribed to.

The controller configuration settings are evaluated as follows:

1. Configuration setting as stored in the APP database.
2. Technical service parameter. If defined, it overrides 1.
3. Custom attribute for customer. If defined, it overrides 1. and 2.
4. Custom attribute for subscription. If defined, it overrides 1. and 2. and 3.

### CONTROLLER\_ID

`CONTROLLER_ID=ess.openstack`

The identifier of the service controller.

### BSS\_ORGANIZATION\_ID

`BSS_ORGANIZATION_ID=<organizationID>`

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

### 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\_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\_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.

## API\_USER\_NAME

`API_USER_NAME=<userName>`

The user name to be used to access the tenant for your organization in OpenStack. Once authenticated, this user is authorized to access the Heat API.

The user must have the necessary credentials to create and configure virtual systems for the tenant. In FUJITSU Cloud Service K5, the `cpf_systemowner` role is required.

## API\_USER\_PWD

`API_USER_PWD=_crypt:<password>`

The password of the user for accessing the tenant in OpenStack.

Replace `<password>` with the plain text password which is valid for the user given in `API_USER_NAME`. The password is encrypted when it is stored in the database.

## KEYSTONE\_API\_URL

`KEYSTONE_API_URL=https://<keystone endpoint>/<version>/auth`

The URL of the Keystone API for authenticating the user specified in `API_USER_NAME`. Keystone is the identity service used by OpenStack.

Replace `<keystone endpoint>` with the URL leading to the Keystone API. Replace `<version>` with the API version to use, currently `v3`. HTTP and HTTPS are supported.

The URL of the Keystone API can also be specified as a parameter in technical service definitions. The technical service parameter overrides the setting you specify here.

## TEMPLATE\_BASE\_URL

`TEMPLATE_BASE_URL=https://<app>/templates/`

The URL leading to the templates that are mapped to technical service definitions. The file names of the templates to be used are specified when customers subscribe to a corresponding service on an OSCM marketplace.

Replace `<app>` with the URL used to access the `app-domain` domain in your application server or another URL that can be reached from OSCM by HTTP or HTTPS.

The template URL can also be specified as a parameter in technical service definitions. The technical service parameter overrides the setting you specify here.

<p><b>Note:</b> First, the <code>bssapp</code> database is searched for the template specified with the <code>TEMPLATE_NAME</code> service parameter in the technical service definition. Only if the template is not found in the database, the <code>TEMPLATE_BASE_URL</code> setting is evaluated.</p>
---

## TENANT\_ID

`TENANT_ID=<tenant>`

The identifier of the tenant for your organization in OpenStack. The tenant is specified in each request to the Heat API.

The tenant ID can also be specified as a parameter in technical service definitions. The technical service parameter overrides the setting you specify here.

## **DOMAIN\_NAME**

`DOMAIN_NAME=<domain>`

Optional. The name of the Keystone domain to use. A domain is a container for projects, users, and groups in the OpenStack environment. If not specified, the default Keystone domain is used (name: default).

The domain name can also be specified as a parameter in technical service definitions. The technical service parameter overrides the setting you specify here.

## **READY\_TIMEOUT**

`READY_TIMEOUT=300000`

Optional. The number of milliseconds to wait for the completion of an operation to start or stop the servers in a stack. If the operation is not completed within the given time, it is aborted with a timeout error. The administrators of the customer organization owning the subscription from which the operation was started are informed by email. After the problem has been sorted out, the start or stop operation can be repeated by the customer.

If the setting is not specified or the given value is not a number, it is ignored, and start and stop operations never time out.

## Appendix B: Service Parameters and Operations

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

### Service Parameters

The OpenStack service controller supports the parameters below.

**Note:** All parameters defined in the technical service definition must be one-time parameters since the modification of parameters is not supported. Be sure to set their `modificationType` to `ONE_TIME`.

---

#### APP\_CONTROLLER\_ID

Mandatory. The ID of the service controller as defined in its implementation.

Default (must not be changed): `ess.openstack`

---

#### ACCESS\_INFO\_PATTERN

Mandatory. The access information to be output in the subscription details on the marketplace as soon as the provisioning is complete. This information must give all the details the customer needs to access a provisioned instance, e.g. an IP address and a key pair name.

The information must correspond to the output parameters specified in the template file. If the values do not match, the subscription is rejected.

Customers should not be able to enter a value for this parameter, i.e. it should not be configurable.

Example: `Key pair name: {KP_Out}; IP: {IP_Out}`

---

#### MAIL\_FOR\_COMPLETION

Optional. The address to which emails are to be sent that describe manual steps required to complete an operation. If you specify this parameter, the service controller interrupts the processing of each operation before its completion and waits for a notification about the execution of a manual action. Omit this parameter if you do not want to interrupt the processing.

Example: `info@company.com`

---

#### STACK\_NAME

Mandatory. The name of the virtual system to be instantiated. This name must be specified by customers when they subscribe to a corresponding service.

The name is restricted to 30 characters. It must start with a letter and only contain the following characters: Letters `A-Z` and `a-z`, numbers `0-9`, hyphen `-`, underscore `_`, period `.`

OpenStack generates a random number that is appended to the name to make it unique.

Example: `MySystem`



---

### STACK\_NAME\_PATTERN

Optional. A regular expression specifying a pattern for the stack names entered by the users when they subscribe to a corresponding service. If the names do not match the pattern, the subscription is rejected.

Stack names must comply with both, the pattern specified here and the general naming conventions (see the description of `STACK_NAME`). If no pattern is given, the general conventions apply.

Customers should not be able to enter a value for this parameter, i.e. it should not be configurable.

Example: `(host_[0-9]*){1,30}`

---

### TEMPLATE\_BASE\_URL

Optional. The URL leading to the template file or files specified in the `TEMPLATE_NAME` parameter. Specify a URL that can be reached from OSCM by HTTP or HTTPS.

If not specified, the URL is obtained from the controller configuration settings.

**Note:** First, the `bssapp` database is searched for the template specified with the `TEMPLATE_NAME` service parameter in the technical service definition. Only if the template is not found in the database, the `TEMPLATE_BASE_URL` setting is evaluated.

Customers should not be able to enter a value for this parameter, i.e. it should not be configurable.

Example: `https://myserver:8880/oscm-app/docroot/templates/`

---

### TEMPLATE\_NAME

Mandatory. The name or relative path and name of the template file which forms the basis for the OpenStack instance to be instantiated. The `TEMPLATE_NAME` is added to the `TEMPLATE_BASE_URL` specified in the technical service definition or in the controller configuration settings.

The template file details everything needed to carry out the resource orchestration in OpenStack. The template must be specified by customers when they subscribe to a corresponding service.

Users should not be able to enter a value for this parameter. This means the parameter should not be configurable for customers or, in case you specify more than one template, have fixed parameter options for selection.

Example: `MyTemplate.json`

---

### TENANT\_ID

Optional. The identifier of the tenant for your organization in OpenStack. The tenant is specified in each request to the Heat API.

If not specified, the tenant ID is obtained from the controller configuration settings.

Customers should not be able to enter a value for this parameter, i.e. it should not be configurable.

Example: `6f4c1e4cbfef4d5a8a1345882fbca110`

---

**KEYSTONE\_API\_URL**

Optional. The URL of the Keystone API for authenticating the user specified to access the tenant for your organization in OpenStack. Keystone is the identity service used by OpenStack.

Specify the URL according to the following format:

```
http[s]://<keystone_endpoint>/<version>/auth
```

<keystone\_endpoint> is the URL leading to the Keystone API, <version> the API version to use, currently v3. HTTP and HTTPS are supported.

If not specified, the URL is obtained from the controller configuration settings.

Customers should not be able to enter a value for this parameter, i.e. it should not be configurable.

Example: `https://my.keystone.com:5000/v3/auth`

---

**DOMAIN\_NAME**

Optional. The name of the Keystone domain to use. A domain is a container for projects, users, and groups in the OpenStack environment.

If not specified, the domain is obtained from the controller configuration settings. If it is not specified there either, the default Keystone domain is used (name: `default`).

Customers should not be able to enter a value for this parameter, i.e. it should not be configurable.

Example: `mydomain`

---

**TP\_ImageId**

Mandatory. The virtual machine image for the instance to be instantiated. Any valid OpenStack image can be specified.

Users should not be able to enter a value for this parameter. This means the parameter should not be configurable for customers or, in case you specify more than one template, have fixed parameter options for selection.

Example: `cedarish` for a cedar image.

---

**TP\_InstanceType (OpenStack) / TP\_flavor (FUJITSU Cloud Service K5)**

Mandatory. The flavor for the instance to be instantiated. The flavor defines the compute, memory, and storage capacity. Any valid OpenStack flavor can be specified.

Users should not be able to enter a value for this parameter. This means the parameter should not be configurable for customers or, in case you specify more than one flavor, have fixed parameter options for selection.

Example: `m1.small`

---

**TP\_KeyName**

Mandatory. The key pair name of the instance to be instantiated.

The key pair name must be specified by the customer when subscribing to an OpenStack service. To log in to the instance, the customer must enter the key pair name and the associated private key.

For details on creating key pairs, refer to the user documentation of OpenStack.

Example: `my-key-pair`

---

**TP\_\***

Optional. Any number of parameters that are mapped from the parameters defined in the template file. The parameters in the template file detail everything needed to carry out the resource orchestration in OpenStack. For each parameter in the template file, there must be a corresponding parameter in the technical service definition.

The parameter names must correspond to the names in the template file. The string **TP\_** must be prepended to the name. If the names do not match this pattern, the subscription is rejected.

## Service Operations for OpenStack Instances

The OpenStack service controller supports the service operations below for the OpenStack instances.

The `actionURL` for each operation is:

`http://oscm-app:8880/oscm-app/webservices/oscm-app/oscm-app/org.oscm.app.v2_0.service.Asynch`  
`<oscm-app>` and `<8880>` are the server and port of the container where the OpenStack service controller is deployed.

### SUSPEND\_VIRTUAL\_SYSTEM

Suspends the OpenStack instance.

### RESUME\_VIRTUAL\_SYSTEM

Resumes the OpenStack instance if it was suspended.

### START\_VIRTUAL\_SYSTEM

Starts all servers in the OpenStack instance that were stopped. For this operation, a timeout value can be configured in the controller configuration settings.

### STOP\_VIRTUAL\_SYSTEM

Stops all servers in the OpenStack instance that were started. For this operation, a timeout value can be configured in the controller configuration settings.

**Note:** If you provision virtual systems that do not support some of these operations or do not contain servers that can be started and stopped, make sure that you remove the corresponding service operations from the technical service definition.

FUJITSU Cloud Service K5 does not support the `SUSPEND_VIRTUAL_SYSTEM` and `RESUME_VIRTUAL_SYSTEM` operations.

# 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.

**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 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.