

Systemwalker Service Catalog Manager V15.1.0 (Business Support System)

A decorative horizontal band with a blue background, featuring abstract geometric shapes like spheres and lines, some with a grid pattern, and glowing light effects.

Operator's Guide

B1WS-0988-02ENZ0(00)
July 2012

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.

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

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

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

Copyright (c) FUJITSU
LIMITED 2010-2012

All rights reserved, including those of translation into other languages. No part of this manual may be reproduced in any form whatsoever without the written permission of FUJITSU LIMITED.

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.

Table of Contents

	About this Manual.....	6
1	Introduction.....	9
1.1	BSS Architecture.....	9
1.2	Organizations and User Roles.....	10
2	Getting Started.....	12
2.1	Overview of Operation and Maintenance Tasks.....	12
2.2	First Steps.....	13
2.3	Startup and Shutdown.....	14
2.4	Tool Support for Operating BSS.....	14
3	Configuring BSS.....	16
3.1	Updating Configuration Settings.....	16
3.2	Adding a Currency to BSS.....	17
3.3	Configuring Timers.....	17
3.4	Full-Text Search Index Configuration.....	20
4	Monitoring, Backup and Recovery.....	21
4.1	Monitoring BSS.....	21
4.1.1	Log Types and Location of Log Files.....	21
4.1.2	Configuring the Log Level.....	21
4.1.3	Log Entry Format.....	23
4.1.4	Maintaining the Log Files.....	23
4.2	Exporting Operation Log Data.....	23
4.3	Monitoring the Calculation of Billing Data.....	24
4.4	Backup and Recovery.....	25
4.5	Checking the Load on the JMS Queue.....	25
5	Managing Organizations and Users.....	26
5.1	Creating an Organization.....	26
5.2	Managing User Accounts.....	27

5.3	Changing the Address Data of an Organization.....	27
5.4	Adding a Role to an Organization.....	28
6	Reporting.....	29
7	Managing Billing and Payment.....	30
7.1	Integrating PSP Support.....	30
7.1.1	Prerequisites.....	31
7.1.2	Registering a Payment Service Provider With BSS.....	32
7.1.3	Configuring PSP Communication and Payment Types.....	32
7.1.4	Setting the PSP Identifier for a Supplier.....	32
7.2	Assigning Payment Types to Suppliers.....	33
8	Integrating Certificates for Trusted Communication.....	34
8.1	Introduction.....	34
8.2	Requirements for Web Service Calls from BSS.....	35
8.3	Requirements for Web Services Calls to BSS.....	35
8.4	Certificate Integration Procedures.....	35
8.4.1	Creating a CA Root Certificate for BSS.....	37
8.4.2	Creating a Self-Signed Domain Certificate	37
8.4.3	Exporting the Self-Signed Domain Certificate	37
8.4.4	Importing the CA Root and Domain Certificates.....	37
8.4.5	Signing an Organization's Certificate.....	38
8.4.6	Setting the Distinguished Name for an Organization.....	38
9	Managing Marketplaces.....	40
Appendix A Application Server Resources.....		41
Appendix B Configuration Settings.....		43
B.1	BSS Configuration Settings.....	43
B.2	GlassFish Configuration Settings.....	50
B.3	GlassFish JMS Configuration Settings.....	51
B.4	Database Configuration Settings.....	51
Appendix C User Operation Log.....		53

Appendix D Menu Options and Required Roles.....56
Glossary59

About this Manual

This manual describes the basic tasks involved in the operation and maintenance of Systemwalker Service Catalog Manager - hereafter referred to as Business Support System (BSS).

The manual is structured as follows:

Chapter	Description
<i>Introduction on page 9</i>	Describes BSS and its architecture, its organizations and users.
<i>Getting Started on page 12</i>	Provides an overview of the tasks involved in the operation and maintenance of BSS, shows the initial steps to be performed after installing BSS, and introduces the tools that support you in performing operational tasks.
<i>Configuring BSS on page 16</i>	Describes how to maintain the BSS configuration settings, how to add a currency, and how to use and configure timers for automatic task processing.
<i>Monitoring, Backup and Recovery on page 21</i>	Describes the BSS log files, how to configure the log level, how to restart payment processes and billing runs, as well as which data is subject to regular backups. In addition, it describes how to monitor the JMS queue.
<i>Managing Organizations and Users on page 26</i>	Describes how to manage organizations and user accounts.
<i>Managing Billing and Payment on page 30</i>	Describes how to set up the payment types allowed for a supplier, and how to integrate BSS with a payment service provider (PSP).
<i>Integrating Certificates for Trusted Communication on page 34</i>	Describes the usage of certificates for secure communication between BSS and applications integrated with it.
<i>Managing Marketplaces on page 40</i>	Describes how to create, update, and delete marketplaces.
<i>Application Server Resources on page 41</i>	Describes the resources required for BSS on the application server.
<i>Configuration Settings on page 43</i>	Describes the BSS configuration settings.
<i>User Operation Log on page 53</i>	Describes the elements of the user operation log.
<i>Menu Options and Required Roles on page 56</i>	Gives an overview of the BSS user interface menu options and the roles required for using them.

Readers of this Manual

This manual is directed to operators who maintain and operate BSS in their environment.

It assumes that you are familiar with the following:

- Administration of the operating systems in use, including the adaption and execution of batch files or shell scripts.
- Java EE 5 technology, particularly as to the deployment on application servers.
- Relational databases and their administration, in particular the PostgreSQL database.
- BSS concepts as explained in the *Overview* manual.
- Web services concepts.
- Installation and administration of Web servers.
- Installation and administration of the GlassFish application server.
- Certificate-based authentication and communication.

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:

BSS	Business Support System
CA	Certification authority
JAX-WS	Java API for XML - Web Services
JMS	Java Message Service
JSP	Java Server Pages
LDAP	Lightweight Directory Access Protocol
PaaS	Platform as a Service
PSP	Payment service provider
SaaS	Software as a Service
SOAP	Simple Object Access Protocol
WSDL	Web Services Description Language
XSD	XML Schema Definition

Available Documentation

The following documentation on BSS is available:

- *Overview*: A PDF manual introducing BSS. It is written for everybody interested in BSS and does not require any special knowledge.
- *Online Help*: Online help pages describing how to work with the user interface of BSS. The online help is directed and available to everybody working at the user interface.
- *Installation Guide*: A PDF manual describing how to install and uninstall BSS. It is directed to operators who set up and maintain BSS in their environment.
- *Operator's Guide*: A PDF manual for operators describing how to administrate and maintain BSS.
- *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 BSS.
- *Supplier's Guide*: A PDF manual for suppliers describing how to define and manage service offerings for applications that have been integrated with BSS.
- *Marketplace Owner's Guide*: A PDF manual for marketplace owners describing how to administrate and customize marketplaces in BSS.
- *Developer's Guide*: A PDF manual for application developers describing the public Web service interface of BSS and how to use it to integrate applications and external systems with BSS.
- Javadoc documentation for the public Web service interface of BSS and additional resources and utilities for application developers.

1 Introduction

Business Support System (BSS) is a set of services which provide all business-related functions and features required for turning on-premise software applications into Software as a Service (SaaS) 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.

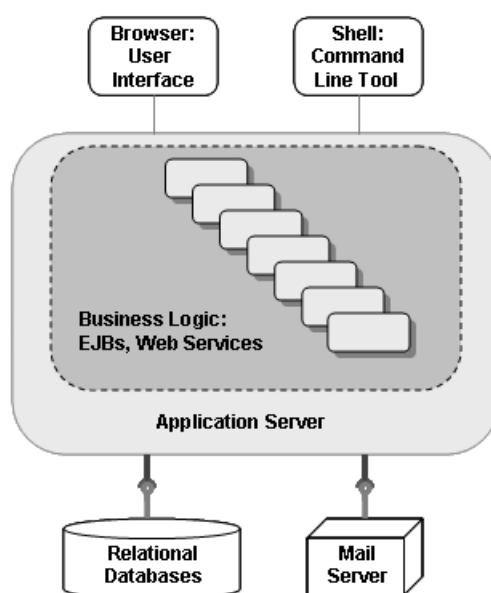
With its components, BSS covers all the business-related aspects of a Platform as a Service (PaaS) or Cloud platform. It supports software vendors as well as their customers in leveraging the advantages of Cloud Computing.

BSS is typically operated in data centers on servers providing for optimum performance, scalability, and non-stop operation. The applications integrated with BSS and their data may be hosted in the same data centers as BSS or in different locations.

1.1 BSS Architecture

BSS is implemented in Java, using Java Platform, Enterprise Edition 5 (Java EE 5) technology. It is deployed on an application server supporting this technology.

The following figure provides an overview of the architecture:



BSS has a three-tier architecture:

- The **presentation layer** in the application server's Web container includes the **user interface**, realized as JavaServer Faces. Users access the user interface in Web browsers. In addition, BSS provides a **command line tool**, which operators can install to carry out their tasks from a command shell instead of the user interface.
- The **business logic** is implemented in Enterprise JavaBeans (EJB). Both the Enterprise JavaBeans and the **public Web service interface** are available in the application server's EJB container. The public Web service interface is mainly used for integrating applications and external systems with BSS. However, it can also be employed for accessing BSS functionality from a Web service

client. HTTP is the protocol for any communication with the user interface and the public Web service interface.

- BSS **persists** its data through the Java Persistence API in **relational databases**.

For informing users about relevant issues (e.g. their registration or assignment to a subscription), BSS must have access to a mail server.

1.2 Organizations and User Roles

Each user working in BSS is a member of a specific organization. An organization typically represents a company, but it may also stand for a department of a company or a single person. Each organization in BSS has a unique account and ID as well as one or more of the following roles: **operator**, **technology provider**, **supplier**, **marketplace owner**, **customer**.

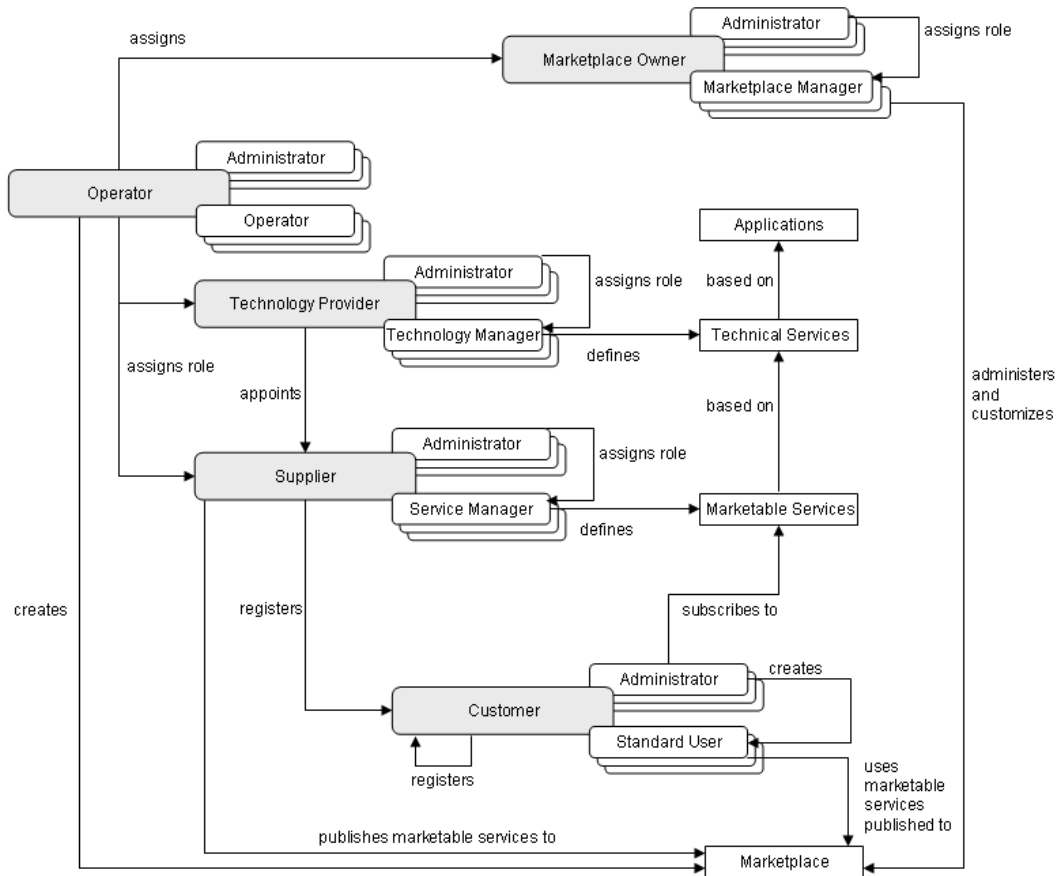
When a customer registers with BSS or is registered by a supplier, an organization with the customer role is created. Organizations with other roles can also act as customers, i.e. they are implicitly assigned the customer role. These organizations are created and assigned their roles as follows:

- When BSS is installed, an organization with the operator role is created.
- Operators can assign the supplier and/or technology provider role to any existing organization or create new organizations with these roles.
- When operators create a marketplace, they specify an existing organization as the owner of the marketplace. In this way, the organization is assigned the marketplace owner role.

The roles of an organization determine which features are available to its users at the BSS interfaces and which roles the users can be assigned. These user roles control the actions an individual user is allowed to carry out:

- **Standard user**: Users with this non-privileged role can work with services their organization has subscribed to. Every user registered in BSS automatically is a standard user. Additional user roles must be assigned explicitly by an administrator.
- **Administrator**: Each organization must have at least one user with this role. An administrator can manage the organization's account and subscriptions as well as its users and their roles. The first administrator of an organization is defined when the organization is created.
- **Technology manager**: This role allows a user to define technical services in BSS. It can be assigned to users of technology provider organizations.
- **Service manager**: This role allows a user to define marketable services and price models as well as publish marketable services. It can be assigned to users of supplier organizations.
- **Marketplace manager**: This role allows a user to define the suppliers who are allowed to publish services to a marketplace as well as update and customize a marketplace. This role can be assigned to users of marketplace owner organizations. It is automatically assigned to all administrators of the marketplace owner organization when a marketplace is created.
- **Operator**: This role allows a user to carry out configuration and maintenance tasks, manage organizations, and create marketplaces. The first operator is created together with its operator organization when BSS is installed.

The following illustration provides an overview of the organizations and user roles and the relationships between them:

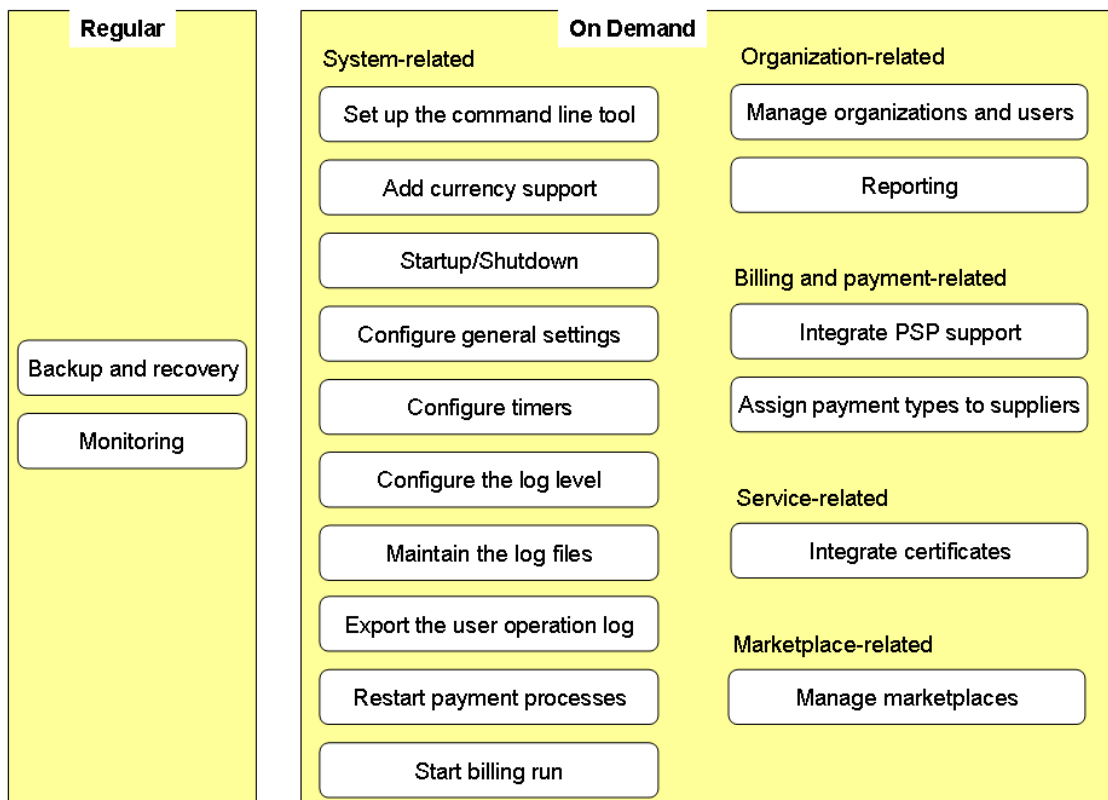


2 Getting Started

This chapter provides an overview of the tasks involved in the operation and maintenance of BSS. It shows the initial steps to be performed after installing BSS, and introduces the tools that support you in performing operational tasks.

2.1 Overview of Operation and Maintenance Tasks

Operating and maintaining BSS involves the following tasks that need to be performed on a regular basis or on demand:



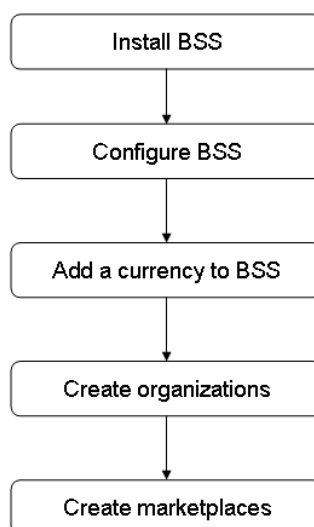
The following table lists where the tasks are described in this manual:

Task	Described in
Backup and recovery	<i>Backup and Recovery</i> on page 25
Monitoring	<i>Monitoring BSS</i> on page 21
Set up the command line tool	<i>Tool Support for Operating BSS</i> on page 14
Add currency support	<i>Adding a Currency to BSS</i> on page 17
Startup/Shutdown	<i>Startup and Shutdown</i> on page 14
Configure general settings	<i>Updating Configuration Settings</i> on page 16

Task	Described in
Configure timers	<i>Configuring Timers</i> on page 17
Configure the log level	<i>Configuring the Log Level</i> on page 21
Maintain the log files	<i>Maintaining the Log Files</i> on page 23
Export the user operation log	<i>Exporting Operation Log Data</i> on page 23
Restart payment processes	<i>Monitoring the Calculation of Billing Data</i> on page 24
Start billing run	<i>Monitoring the Calculation of Billing Data</i> on page 24
Manage organizations and users	<i>Managing Organizations and Users</i> on page 26
Reporting	<i>Reporting</i> on page 29
Integrate PSP support	<i>Integrating PSP Support</i> on page 30
Assign payment types to suppliers	<i>Assigning Payment Types to Suppliers</i> on page 33
Integrate certificates	<i>Integrating Certificates for Trusted Communication</i> on page 34
Manage marketplaces	<i>Managing Marketplaces</i> on page 40

2.2 First Steps

This section shows the first steps to be performed for getting started with operating a BSS installation:



When you have successfully installed and configured BSS following the instructions in the *Installation Guide*, you have access to the BSS user interface and a command line tool. For a description on how to access the available tools, refer to *Tool Support for Operating BSS* on page 14.

In order for a supplier to be able to define price models and for the rating and billing engine of BSS to be able to calculate usage costs, you need to add one or several currencies to the system. For details, refer to *Adding a Currency to BSS* on page 17.

In the next step, you can create technology provider and supplier organizations and set up marketplaces so that suppliers can offer their services to customers. For details, refer to *Managing Organizations and Users* on page 26 and *Managing Marketplaces* on page 40.

2.3 Startup and Shutdown

When you have completed the installation of BSS and provided the required resources on the application server as described in the *Installation Guide*, BSS is available on the application server as an Enterprise Application named `fujitsu-bss`. It can be started and accessed using the standard utilities of the application server. A shutdown of BSS is also performed using the standard functionality of the application server.

2.4 Tool Support for Operating BSS

BSS provides operator functions at its user interface that support you in performing on-demand maintenance and operation tasks. For some functions, you can also use the operator client, which is available as a command line tool.

The first operator is created when BSS is installed. This operator is initially assigned the following credentials:

- User ID: `administrator`
- User key: `1000`
- Initial password: `admin123`

The user ID and password must be specified for accessing the BSS user interface, the user key and password are required for executing commands with the command line tool.

It is recommended that you change the initial password at the BSS user interface (**Edit Profile** page in the **Account** menu).

BSS User Interface

You can access the BSS user interface in a Web browser using an URL in the following format:

```
http://<server>:<port>/fujitsu-bss-portal
```

`<server>` is the Web/application server where BSS has been deployed. `<port>` is the port to address the Web/application server (default: 80 for HTTP, 443 for HTTPS).

You are prompted for the user ID and password.

After login, the operator functionality is available in the **Operation** menu.

Command Line Tool

The command line tool provides a subset of the functions that are available in the **Operation** menu at the BSS user interface.

The command line tool is provided in the BSS installation package, `fujitsu-bss-install-pack.zip`, as `fujitsu-bss-operatorclient.zip`. The contents of this package can be made available in your environment as follows:

1. Extract the contents of the `fujitsu-bss-operatorclient.zip` file to a separate temporary directory on the system where you have installed BSS. The directory contains the required configuration files, jar files, and scripts.
2. Set the `GLASSFISH_HOME` environment variable to the directory where you have installed the application server.

3. Run the `prepareCP.cmd` script located in the directory to which you extracted the `fujitsu-bss-operatorclient.zip` file. This script copies the required `.jar` files from the application server installation directory to the environment where you want to use the command line tool.
4. Adapt the settings in the `env.properties` file to your environment. This file is located in the directory to which you extracted the `fujitsu-bss-operatorclient.zip` file.

Command Syntax:

The syntax of all available commands is as follows:

```
java -jar fujitsu-adm-um-operatorsvc-client.jar <userkey> <password>  
<command> <parameter>
```

where

`userkey` is the numeric key of the operator. The key of the initial operator is 1000.

`password` is the password of the operator.

`command` is the command to be executed.

`parameter` is a command parameter to be set.

3 Configuring BSS

This chapter describes:

- How to update configuration settings for BSS.
- How to add a currency definition to BSS.
- How to use and configure timers in BSS.
- How to configure where the full-text search index is stored.

3.1 Updating Configuration Settings

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

The BSS configuration settings are provided in the following files in the directory to which you extracted the `fujitsu-bss-setup.zip` file (`<setup_dir>`) when you installed BSS:

- `configsettings.properties`: Configuration settings for the BSS services.
The installation stores these settings in the database. You can change them later, if required. An update installation only adds new settings to the database but does not overwrite existing ones.
- `glassfish.properties`: Configuration settings for the application server.
- `glassfishJMSBrokerConfiguration.properties`: Configuration settings for the Java Message Service (JMS) in the application server.
- `db.properties`: Settings for database setup and access.

For details on the individual configuration settings, refer to *Configuration Settings* on page 43.

To update existing settings for the BSS services, choose one of the following options:

- **BSS user interface** - Choose **Update configuration settings** in the **Operation** menu.
The current configuration settings as stored in the database are retrieved. You can check all settings and update them as required by entering a new value for the corresponding configuration key.
For detailed step-by-step instructions, refer to the online help.
- **Edit the content of the `configsettings.properties` file as required.**
Then execute the `build-db.xml` file in `<setup_dir>` as follows:

```
<GLASSFISH_HOME>/lib/ant/bin/ant -f <setup_dir>/build-db.xml
updateConfigSettings
```

`<GLASSFISH_HOME>` is the installation directory of the application server.

- **Command line tool** - Use the `saveconfigurationsetting` command with the following syntax:

```
java -jar fujitsu-adm-um-operatorsvc-client.jar <userkey> <password>
saveconfigurationsetting settingKey=<key of the setting>
settingValue=<value>
```

Execute the above command for every configuration setting you want to update.

To update the configuration settings for the application server, proceed as follows:

1. Open the `glassfish.properties` file 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 `<setup_dir>`.
4. Update the settings and resources for BSS in the application server by executing the `build-glassfish-setup.xml` file in `<setup_dir>` as follows:

```
<GLASSFISH_HOME>/lib/ant/bin/ant -f <setup_dir>/build-glassfish-setup.xml
```

`<GLASSFISH_HOME>` is the installation directory of the application server.

Note: Be aware that existing entries in the JMS queue are deleted by executing the above command.

To update the configuration settings for JMS or the database, proceed as follows:

- Login to the administration console of the application server.
- Adapt the settings as required.

Retrieving the Current Configuration Settings

You can at any time view the current configuration settings for the BSS services.

To retrieve the current configuration settings, choose one of the following options:

- BSS user interface - Choose **Update configuration settings** in the **Operation** menu.
For detailed step-by-step instructions, refer to the online help.
- Command line tool - Use the `getconfigurationsettings` command with the following syntax:

```
java -jar fujitsu-adm-um-operatorsvc-client.jar <userkey> <password>  
getconfigurationsettings
```

For details on the individual configuration settings, refer to *Configuration Settings* on page 43.

3.2 Adding a Currency to BSS

After installation, you need to add the currency or currencies to be supported by BSS. These currencies will be available to suppliers when defining the price models for marketable services. The selected currencies are used by the integrated rating and billing engine of BSS when calculating subscription usage charges.

To add a currency, choose one of the following options:

- BSS user interface - Choose **Manage currencies** in the **Operation** menu.
For detailed step-by-step instructions, refer to the online help.
- Command line tool - Use the `addcurrency` command with the following syntax:

```
java -jar fujitsu-adm-um-operatorsvc-client.jar <userkey> <password>  
addcurrency currencyISOCode=<ISO Code>
```

3.3 Configuring Timers

Timers are used to handle background tasks, for example, to check for expired subscriptions that are to be deleted. Each timer has a time interval that is defined and can be changed in the BSS configuration settings. If there is no setting for a timer, the default value is zero and the timer is disabled.

Note: Be aware that several functions in BSS will not work if you do not enable the timers. For example, if a customer specifies that a subscription is to expire in 10 days, and you did not configure the timer `RESTRICTED_SUBSCRIPTION_USAGE_PERIOD`, the subscription will not expire after 10 days.

By default, January 1st is assumed to be the date when every timer was executed for the first time. This date is taken as the initial expiration date for the timers. Consequently, the time for the next expiration is calculated based on this assumption. To avoid the expiration of all timers at the same date, which would result in heavy load on the system, there is an additional setting: A separate **offset** for each timer. The offset is added to the initial expiration time.

Example

For the timer used to remove customer accounts that have not been confirmed, the following configuration settings are defined:

```
PERMITTED_PERIOD_UNCONFIRMED_ORGANIZATIONS=604800000
TIMER_INTERVAL_ORGANIZATION=86400000
TIMER_INTERVAL_ORGANIZATION_OFFSET=300000
```

The first setting indicates how long an organization account is allowed to remain unconfirmed: 7 days. The second setting indicates the time interval at which the timer is executed: Every 24 hours. The third setting, the offset, is set to 5 minutes. The timer will expire for the first time on the day after starting BSS at 5 minutes past midnight. Further expirations will occur at 0:05 a.m. every day. The offset is not accumulated, but stays the same every day.

Available Timers

There are the following timers:

- Timer to check for organization accounts that have not been confirmed by a login of the initial administrator within a certain period of time. When this timer expires, the respective organization accounts are removed.

This timer requires the following configuration settings:

- `TIMER_INTERVAL_ORGANIZATION`
- `TIMER_INTERVAL_ORGANIZATION_OFFSET`
- `PERMITTED_PERIOD_UNCONFIRMED_ORGANIZATIONS`
- Timer used to ensure that subscriptions can only be used for the time specified in the underlying service's parameters. If this period is exceeded, the timer-related operations must be executed to make sure that the subscription cannot be used anymore unless the supplier upgrades or downgrades the underlying service.

This timer requires the following configuration settings:

- `TIMER_INTERVAL_SUBSCRIPTION_EXPIRATION`
- `TIMER_INTERVAL_SUBSCRIPTION_EXPIRATION_OFFSET`
- Timer used to check pending subscriptions. When the timeout time is reached, an email is sent to the administrator of the organization owning the subscriptions, informing him about the timeout.

This timer requires the following configuration settings:

- `TIMER_INTERVAL_TENANT_PROVISIONING_TIMEOUT`
- `TIMER_INTERVAL_TENANT_PROVISIONING_TIMEOUT_OFFSET`

- Timer for billing runs. The billing run interval is one month and cannot be changed.
This timer requires the following configuration setting:
 - `TIMER_INTERVAL_BILLING_OFFSET`
- Timer used to check whether the end date for discounts granted to customers has been reached. The timer interval is one day and cannot be changed.
This timer requires the following configuration settings:
 - `TIMER_INTERVAL_DISCOUNT_END_NOTIFICATION_OFFSET`
- Timer used to remove non-existing users from the database that were created because an organization acted on behalf of another organization.
The timer for database cleanup requires the following configuration settings:
 - `TIMER_INTERVAL_INACTIVE_ON_BEHALF_USERS`
 - `TIMER_INTERVAL_INACTIVE_ON_BEHALF_USERS_OFFSET`

The values for timers are indicated in milliseconds. The offset is always based on January 1, 00:00. For a detailed description of the timers, refer to *BSS Configuration Settings* on page 43.

To configure a timer, choose one of the following options:

- BSS user interface - Choose **Update configuration settings** in the **Operation** menu.
For detailed step-by-step instructions, refer to the online help.
- Command line tool - Use the `saveconfigurationsetting` command with the following syntax:

```
java -jar fujitsu-adm-um-operatorsvc-client.jar <userkey> <password>
saveconfigurationsetting settingKey=<key of the setting>
settingValue=<value>
```

Retrieving Expiration Times

You can check when the currently registered timers expire.

To retrieve the expiration times, choose one of the following options:

- BSS user interface - Choose **Manage timers** in the **Operation** menu.
For detailed step-by-step instructions, refer to the online help.
- Command line tool - Use the `retrievetimerexpirations` command with the following syntax:

```
java -jar fujitsu-adm-um-operatorsvc-client.jar <userkey> <password>
retrievetimerexpirations
```

Re-Initializing Timers

When you set a timer or update the settings for a timer, you need to re-initialize the timers in order to start them.

To re-initialize the timers, choose one of the following options:

- BSS user interface - Choose **Manage timers** in the **Operation** menu.
For detailed step-by-step instructions, refer to the online help.

- Command line tool - Use the `reinittimers` command with the following syntax:

```
java -jar fujitsu-adm-um-operatorsvc-client.jar <userkey> <password>
reinittimers
```

3.4 Full-Text Search Index Configuration

On marketplaces, users can search for services using the BSS full-text search. By default, the full-text search index is stored in the current working directory of the GlassFish domain where BSS has been deployed, for example:

```
<GLASSFISH_HOME>/domains/<your_domain>/config
```

It is recommended to set the path explicitly, so that you know where to look for the index files. This is useful if you need to delete the index files, for example, because the full-text search index ran out of synch.

To set the path where to store the full-text search index, proceed as follows:

1. Make sure that the BSS domain is running.
2. Log in to the GlassFish administration console.
3. Go to **Configuration** -> **System properties**.
4. Add the following property:

`|hibernate.search.default.indexBase|` and specify the path where to store the index as the value.

5. Restart the domain.

4 Monitoring, Backup and Recovery

Regular system operation and maintenance includes the monitoring of the system and its processes as well as the backup of the databases, configuration settings, and log files.

This chapter describes:

- The monitoring of BSS, including the types and locations of log files.
- How to configure the log level.
- The format of the log entries.
- How to maintain the log files.
- How to export logged user operations related to subscriptions.
- How to monitor and manage the calculation of billing data.
- The backup of the databases, configuration settings, and log files.
- The monitoring of the JMS queue.

4.1 Monitoring BSS

In addition to entries in the standard application server log file, BSS provides log files of its own that help you detect problems and identify, for example, illegal access to the system.

The logging of BSS is based on the `log4j` tool.

4.1.1 Log Types and Location of Log Files

BSS writes its logging information into the following log files:

- `system.log`: System status, errors, warnings, and debug information.
- `access.log`: User logins.
- `audit.log`: System security status.

The log files are stored in the location defined by the `LOG_FILE_PATH` configuration setting. By default, they are located in the following directory:

```
<GLASSFISH_HOME>/domains/<your_domain>/logs
```

In addition to the BSS-specific log files, the application server log file (`server.log`) is another source for tracking system usage and possible errors.

4.1.2 Configuring the Log Level

BSS supports the following types of log information with their corresponding log level:

- **ERROR**: Problems that do not allow to continue working with BSS in the current transaction or that indicate an issue which must be solved. With the `ERROR` log level set, the log file contains all exceptions that occurred at runtime with a complete stack trace.
- **WARN**: Problems that allow for the completion of an operation, irrespective of whether the operation is completed fully or only partially. For example, an email could not be sent. With the `WARN` log level set, the log file comprises all messages of type `WARN` and `ERROR`.
- **INFO**: Basic information such as the state of the server, whether it was started or stopped, whether a user logged in. With the `INFO` log level set, the log file comprises all messages of type `INFO`, `WARN`, and `ERROR`.

- **DEBUG:** Detailed information with references to the BSS implementation, mainly start and exit of methods as well as proposals of how to solve the problem. With the `DEBUG` log level set, the log file comprises all messages.

The `log4j` logging framework optionally uses the `log4j.properties` configuration file. The `log4j.properties` contains the settings for the log level and a list of configuration settings that determine the behavior of the framework at runtime. The log level defines which type of logged information is written to the log files. You can change the log level by adapting the file. By default, the `INFO` log level is used.

The `LOG_CONFIG_FILE` configuration setting stores the absolute path to the `log4j.properties` configuration file. By default, BSS looks for this file in the `conf` subdirectory of your application server domain:

```
<GLASSFISH_HOME>/domains/<your_domain>/config
```

When the `log4j.properties` file is specified for the first time, BSS needs to be restarted. Afterwards, if the `log4j.properties` file is found, its content is checked for changes every 60 seconds. If there are changes to the log level, they are applied to all loggers at runtime; BSS need not be restarted.

If `log4j.properties` does not exist or the value for the configuration setting `LOG_CONFIG_FILE` is not set or has been changed, the log level is the one stored in the `LOG_LEVEL` configuration setting. If you change this setting, you need to restart BSS for the new level to become effective.

The `log4j.properties` file may contain entries for defining different log levels for the different BSS services. For example, if the following is defined in the `log4j.properties` file, all events caused by the Identity Service are logged on `DEBUG` level, all others on `INFO` level in the `system.log` file.

```
log4j.rootLogger=INFO
log4j.logger.sys.com.fujitsu.adm.um.identityservice=DEBUG
```

You can define different log levels for the following services:

Service	Package name
Account management service	com.fujitsu.adm.um.accountservice
Billing service	com.fujitsu.adm.um.billingservice
Categorization service	com.fujitsu.adm.um.categorizationservice
Discount service	com.fujitsu.adm.um.discountservice
Event management service	com.fujitsu.adm.um.eventservice
Identification service	com.fujitsu.adm.um.identityservice
Marketplace management service	com.fujitsu.adm.um.marketplace
Reporting service	com.fujitsu.adm.um.reportingservice
Review service	com.fujitsu.adm.um.reviewservice
SAML service	com.fujitsu.adm.um.credential
Search service	com.fujitsu.adm.um.searchservice
Service provisioning service	com.fujitsu.adm.um.serviceprovisioningservice
Session service	com.fujitsu.adm.um.sessionservice

Service	Package name
Subscription management service	com.fujitsu.adm.um.subscriptionsservice
Tag service	com.fujitsu.adm.um.serviceprovisioningservice
Trigger service	com.fujitsu.adm.um.triggerservice
Trigger definition service	com.fujitsu.adm.um.triggerdefinitionservice
VAT service	com.fujitsu.adm.um.vatservice

4.1.3 Log Entry Format

Every entry in a BSS log file has the following format:

```
<YYYY-MM-DD'T'HH:MM:SS z> LogLabel:<label> LogType:<level> MessageID: <MessageID>
Message: <Message>
```

`LogLabel` identifies the product that created the log entry. For BSS, this is `FSP_INTS-BSS`. `LogType` is one of the log levels, `MessageID` is a numeric identifier of the entry.

Detailed information on the logged issue is contained in the `Message` text. The text contains the ID of the thread handling a transaction, which can be a useful indicator of relations between the activities on the server itself. Knowing the ID of the thread handling a transaction, you can completely track all steps involved in the transaction and thus easily investigate an error.

4.1.4 Maintaining the Log Files

New messages are continually appended to the log files at runtime. When a file reaches a size of 10 MB, a new one is created; the number 1 is appended to the old one. When the second log file reaches 10 MB in size, the number 2 is appended to it, and a third log file is created. When the maximum number of 5 log files is reached, the first one is overwritten.

It is recommended to check the log files and their sizes from time to time and to create a backup when you see that the maximum number of files will be reached.

4.2 Exporting Operation Log Data

You can export operation log data stored in the database for a specific period in time. The data represents an audit log showing all user operations related to subscriptions on your platform. Exporting this data may be useful, for example, to check when and by whom a price model was changed or when subscriptions were created.

The period in time is specified as start date and end date in the format `YYYY-MM-DD`.

To export the operation log, choose one of the following options:

- BSS user interface - Choose **Export operation log** in the **Operation** menu.
For detailed step-by-step instructions, refer to the online help.
- Command line tool - Use the `getuseroperationlog` command with the following syntax:

```
java -jar fujitsu-adm-um-operatorsvc-client.jar <userkey> <password>
getuseroperationlog filename=test.csv entitytype=SUBSCRIPTION
from=<startdate> to=<enddate>
```

You can choose to view the exported data or save it to a file. For details on the content, refer to *User Operation Log* on page 53.

4.3 Monitoring the Calculation of Billing Data

At regular intervals (every month), the database content is checked for organizations which have produced billing-relevant data and the data is collected. The billing data is calculated. This check and calculation is called "billing run". When the billing run is finished, the billing data that must be handled by a payment service provider (PSP) are processed. This transaction is called "payment processing".

Start Billing Run

If you want to calculate the billing data for the last billing period of all organizations managed on your platform, you can explicitly start a billing run. This is useful, for example, if you want to check the payment processing of billing data by a PSP test system, and you do not want to wait for the automatic monthly calculation of the data.

To start a billing run, choose one of the following options:

- BSS user interface - Choose **Execute billing tasks** in the **Operation** menu, and click **Execute** in the respective section of the Web page.
- Command line tool - Use the `startbillingrun` command with the following syntax:

```
java -jar fujitsu-adm-um-operatorsvc-client.jar <userkey> <password>  
startbillingrun
```

Start Payment Processing

You can manually start the payment processing of billing data that must be handled by a PSP.

To start the payment processing, choose one of the following options:

- BSS user interface - Choose **Execute billing tasks** in the **Operation** menu, and click **Execute** in the respective section of the Web page.
- Command line tool - Use the `startpaymentprocessing` command with the following syntax:

```
java -jar fujitsu-adm-um-operatorsvc-client.jar <userkey> <password>  
startpaymentprocessing
```

Retry Failed Payment Processes

When communication problems caused the automatic payment processing for an organization to fail, you can re-invoke these payment processes manually.

To retry failed payment processes, choose one of the following options:

- BSS user interface - Choose **Execute billing tasks** in the **Operation** menu, and click **Execute** in the respective section of the Web page.
- Command line tool - Use the `retryfailedpaymentprocesses` command with the following syntax:

```
java -jar fujitsu-adm-um-operatorsvc-client.jar <userkey> <password>  
retryfailedpaymentprocesses
```


Export Billing Data

You can export existing billing data for a customer organization for a specific time frame. The billing data can be saved to an XML file or opened in an editor of your choice. You can edit the billing data and work with it as required, for example, forward the data to an accounting system. If no billing data exists, a billing run is started; however, the calculated data is not stored.

The period in time is specified by entering the first day of a month and the first day of the following month. The dates need to be specified in the format `YYYY-MM-DD`.

To export billing data, choose one of the following options:

- BSS user interface - Choose **Export billing data** in the **Operation** menu.
For detailed step-by-step instructions, refer to the online help.
- Command line tool - Use the `getbillingdata` command with the following syntax:

```
java -jar fujitsu-adm-um-operatorsvc-client.jar <userkey> <password>
getbillingdata orgid=<organizationID> from=<startdate> to=<enddate>
```

The exported billing data is saved to an XML file (`<date>BillingData.xml`) and opened in the editor of your choice.

You find a detailed description of the XML file elements in the *Developer's Guide*.

4.4 Backup and Recovery

BSS does not offer integrated backup and recovery mechanisms. Use the standard file system, application server, and database mechanisms instead.

It is recommended to create a regular backup of the following data according to the general guidelines of the data center:

- BSS databases. The frequency of database backups depends on the amount of changes and on the availability of time slots with low load. The PostgreSQL database supports backups without previous shutdown. For details, refer to the PostgreSQL documentation.
- BSS log files located in the configured directory.
- Configuration files.

4.5 Checking the Load on the JMS Queue

BSS handles calls to the notification service of an external process control system via Web services. The calls are stored in a JMS queue before they are actually executed.

It is recommended to check the load on the JMS queue on a regular basis using the application server administration console. If the load is too high, you may want to set up more cluster nodes or take another appropriate action.

Note: In case a restart of the database used for JMS data (`bssjms`) is required, make sure to also restart the GlassFish domains in order to speed up the JMS recovery.

5 Managing Organizations and Users

This chapter describes how to:

- Create a technology provider or supplier organization.
- Manage user accounts: lock and unlock user accounts, and reset the password for a user.
- Maintain the address data and roles of organizations.

5.1 Creating an Organization

As an operator, you can at any time create a new supplier or technology provider organization. For every new organization, you must define a user who is to become the first administrator. This user can later register new users and assign roles to them.

To create an organization, choose one of the following options:

- BSS user interface - Choose **Create organization** in the **Operation** menu.

For detailed step-by-step instructions, refer to the online help.

- Command line tool:

You need to define several properties for the new organization in a properties file before executing the respective command. Proceed as follows:

1. Edit the `org.properties` file located in the directory to which you extracted the `fujitsu-bss-operatorclient.zip` file. The following entries are mandatory:

- `user.address`: Address of the administrator.
- `user.additionalname`: Any additional name the administrator might have.
- `user.email`: Email address of the organization's administrator. Notifications about the creation of the account and access information will be sent to this address.
- `user.firstname`: The administrator's first name.
- `user.identifier`: User ID of the organization's administrator. User IDs are restricted to 100 characters and must not contain any of the following characters:
! " # \$ % & ' * + , / : ; < = > ? \ ^ `
- `user.lastname`: The administrator's last name.
- `user.locale`: Language in which the administrator will work by default.
The user interface, available documentation, and emails will be presented to the administrator in the language specified here.
- `user.phone`: Contact telephone or mobile number of the administrator.
- `user.salutation`: The administrator's title, such as Mr. or Mrs.
- `organization.description`: Description of the organization, for example, information from the company's profile.
- `organization.email`: Email address for general communication with the organization. The organization can later specify a separate email address to which invoices are sent.
- `organization.locale`: Default language used in the organization.
- `organization.phone`: Telephone or mobile number of a contact person in the organization.
- `organization.url`: URL of the organization's website. This website can be accessed by customers when they display details on a supplier organization that offers a service they want to subscribe to.

- `organization.name`: Name of the organization, for example, the company name.
 - `organization.address`: Address of the organization. The organization's administrator can later specify a different address to be printed on invoices.
 - `organization.domicile`: Country where the organization is located. Specify an ISO country code, for example `DE` or `US`.
 - `organization.roles`: Roles the new organization is to have. Valid values are `TECHNOLOGY_PROVIDER` and `SUPPLIER`. If the organization wants to act both as a supplier and a technology provider, separate the values by a comma without any blanks.
2. Execute the `createorganization` command with the following syntax:

```
java -jar fujitsu-adm-um-operatorsvc-client.jar <userkey> <password>
createorganization -f org.properties
```

5.2 Managing User Accounts

As an operator, you can lock the account of a specific user, for example, when you assume unauthorized access. You can also unlock an account, for example, in case it was locked because the user tried to log in using a wrong password for the configured number of times (default: 3). In addition, you can initiate the generation of a new password for a user. In all cases, the user is notified by email.

To manage user accounts, choose one of the following options:

- BSS user interface - Choose **Manage users** in the **Operation** menu and click the appropriate button.

For detailed step-by-step instructions, refer to the online help.

- Command line tool - Use the following commands:
 - `lockuser` for locking a user account:

```
java -jar fujitsu-adm-um-operatorsvc-client.jar <userkey> <password>
lockuser userid=<userID>
```

- `unlockuser` for unlocking a user account:

```
java -jar fujitsu-adm-um-operatorsvc-client.jar <userkey> <password>
unlockuser userid=<userID>
```

- `resetpasswordforuser` for resetting the password of a user:

```
java -jar fujitsu-adm-um-operatorsvc-client.jar <userkey> <password>
resetpasswordforuser userid=<userID>
```

5.3 Changing the Address Data of an Organization

As an operator, you can at any time update the address data of an organization.

To update the address data of an organization, choose one of the following options:

- BSS user interface - Choose **Manage organization** in the **Operation** menu.

For detailed step-by-step instructions, refer to the online help.

- Command line tool:

You need to define the organization properties in the `org.properties` file and save it to an `update_org.properties` file before executing the respective command. Proceed as follows:

1. Copy the `org.properties` file located in the directory to which you extracted the `fujitsu-bss-operatorclient.zip` file and save it with the name `update_org.properties`.
2. Adjust the entries in the `update_org.properties` file and add the following entry:
`organization.id`: ID of the organization whose data you want to update.
3. Use the `updateorganization` command with the following syntax:

```
java -jar fujitsu-adm-um-operatorsvc-client.jar <userkey> <password>  
updateorganization -f update_org.properties
```

5.4 Adding a Role to an Organization

As an operator, you can add a role to an organization (technology provider or supplier).

To do so, choose one of the following options:

- BSS user interface - Choose **Manage organization** in the **Operation** menu.
For detailed step-by-step instructions, refer to the online help.
- Command line tool - Use the `addorganizationtorole` command with the following syntax:

```
java -jar fujitsu-adm-um-operatorsvc-client.jar <userkey> <password>  
addorganizationtorole orgid=<organizationID> role={SUPPLIER |  
TECHNOLOGY_PROVIDER}
```

6 Reporting

BSS offers comprehensive reports for different purposes and at different levels of detail. You can choose from various predefined reports.

The following reports are available for operators:

- **Supplier revenue report:** The report is focused on the revenue of each supplier managed on the platform. This is useful in case there is an agreement on a shared revenue model for using the platform. You can use the data for issuing invoices to the individual suppliers. You can specify a time period for the report. Dates are to be entered in the format `YYYY-MM`.

For each supplier, the report shows the supplier's name, organization ID, and revenue per currency. The report adds up all the listed revenues for each currency.

- **External services report:** The report shows all suppliers who have published services with the external access type to a marketplace managed on your platform.

For each supplier, the report shows the marketable services with their activation and deactivation time. If the services were activated and deactivated several times, all timestamps are listed.

To create a supplier revenue report, choose one of the following options:

- BSS user interface - Choose **Create report** in the **Account** menu, and select the report of your choice.

For detailed step-by-step instructions, refer to the online help.

The generated report is instantly displayed at the BSS user interface. You can choose to print the report or save it in several formats using the icons in the pane where the report is displayed.

- Command line tool - Use the `getsupplierrevenuelist` command with the following syntax:

```
java -jar fujitsu-adm-um-operatorsvc-client.jar <userkey> <password>
getsupplierrevenuelist month=<YYYY-MM>
```

The command returns a CSV list in the following format:

`FROM, TO:` Dates specifying the period for which billing data have been generated.

`ID, NAME:` ID and name of the supplier.

`AMOUNT:` For each supplier, total revenue amount per currency and marketplace.

`CURRENCY:` ISO currency code.

`MARKETPLACE:` Name of the marketplace as defined in the locale setting of the calling user. If not set, the marketplace ID is returned.

To create an external services report, choose the following option:

- BSS user interface - Choose **Create report** in the **Account** menu, and select the report of your choice.

For detailed step-by-step instructions, refer to the online help.

The generated report is instantly displayed at the BSS user interface. You can choose to print the report or save it in several formats using the icons in the pane where the report is displayed.

Note: The reporting functionality is available only if you have defined the following configuration parameters: `REPORT_ENGINEURL`, `REPORT_SOAP_ENDPOINT`, `REPORT_WSDLURL`. For details, refer to *BSS Configuration Settings* on page 43.

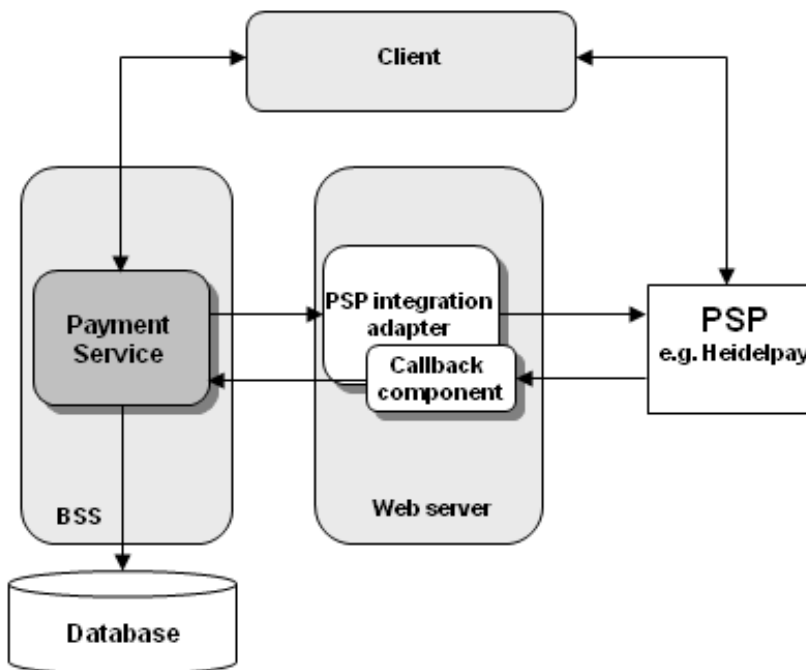
7 Managing Billing and Payment

BSS automatically calculates billing-relevant data based on the price models related to subscriptions. At regular intervals (every month), BSS executes a billing run and produces raw billing data that can be used in an accounting system. You should check at regular intervals whether billing runs fail, and, if yes, re-invoke the payment processes or explicitly start the billing run for the last billing period. For details, refer to *Monitoring the Calculation of Billing Data* on page 24.

For payment collection, a supplier organization can use the facilities of a commercial payment service provider (PSP). This chapter describes the prerequisites for integrating PSP support, how to integrate a PSP, and how to set up the payment types a supplier can offer to customers.

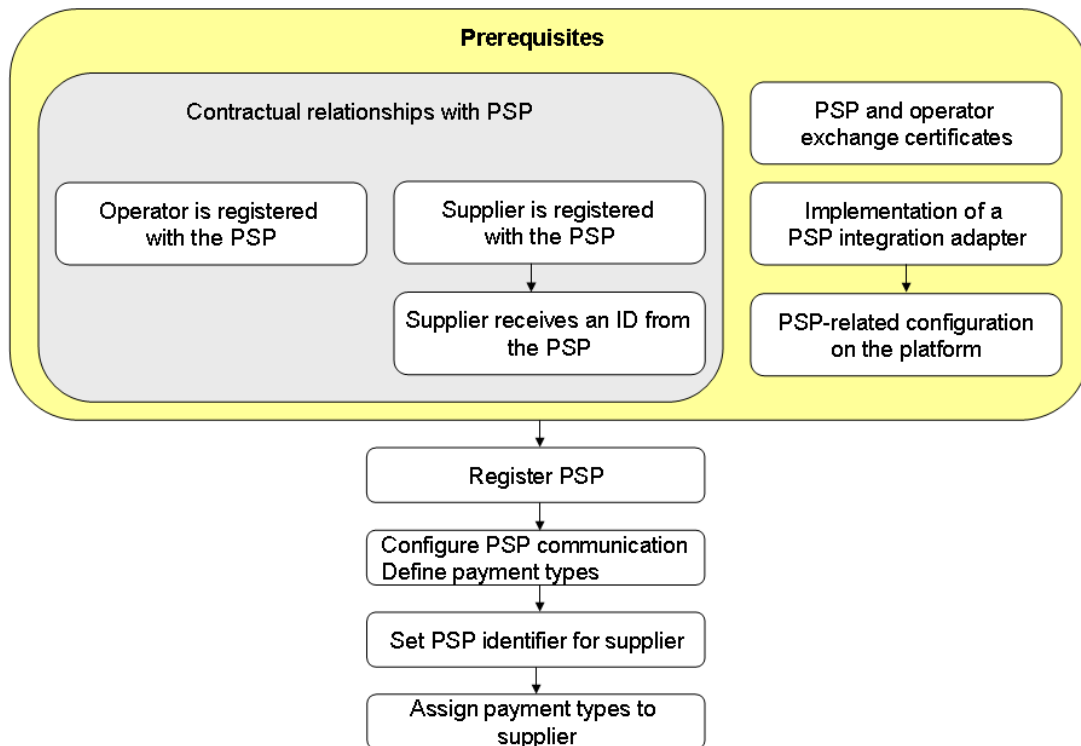
7.1 Integrating PSP Support

BSS provides a Web service interface (payment service) to enable operator organizations to integrate payment service providers (PSPs) with BSS. As a sample PSP integration, Heidepay is supported. The following figure illustrates the communication paths between the components:



You find a detailed description of the implementation of a PSP integration adapter in the documentation shipped with the `fujitsu-bss-integration-psp-pack.zip` file. This package also contains the `heidepay-sample-integration.zip` file which you can use to integrate Heidepay as a PSP.

The process of integrating PSP support for a supplier organization and the tasks to be performed by an operator are as follows:



The subsequent sections describe the prerequisites and the operator tasks in detail.

7.1.1 Prerequisites

Before a supplier can use the services of a commercial payment service provider (PSP), several steps need to be performed:

1. Your organization, i.e. the operator organization, must be registered in the PSP system.
2. The suppliers who are to use the services of the PSP need to have a contractual relationship with the PSP. The supplier organizations must also be registered in the PSP system.
3. The PSP issues an identifier for the supplier so that payments can be immediately linked to the supplier's account.
4. Your own organization and the PSP organization must have exchanged certificates so that trusted and secure communication between BSS and the PSP system is ensured.

For details, refer to *Integrating Certificates for Trusted Communication* on page 34.

5. A PSP integration adapter must have been implemented as a Web service and deployed on an application server. You need the URL pointing to the WSDL file of this service.

In addition, the PSP must have informed you about the types of payment he offers and which configuration values to set for using his services.

You find a detailed description of the implementation of a PSP integration adapter in the documentation shipped with the `fujitsu-bss-integration-psp-pack.zip` file. This package also contains the `heidelpay-sample-integration.zip` file.

6. You need to make the following configuration setting for BSS:

`PSP_USAGE_ENABLED`: Set the value to `true` in order to enable PSP integration (default: `false`).

For details, refer to *Updating Configuration Settings* on page 16.

7.1.2 Registering a Payment Service Provider With BSS

Before a supplier can use the services of a commercial PSP, you need to register the PSP with BSS.

To register a PSP, you have the following option:

- BSS user interface - Choose **Create payment service provider** in the **Operation** menu.
For detailed step-by-step instructions, refer to the online help.

You need to provide the following information:

- Identifier of the PSP. This ID uniquely identifies the PSP in BSS and is visible to the operator only.
- WSDL URL: Link to the WSDL file containing the PSP integration adapter. This adapter is a Web service definition providing the methods for integrating the PSP with BSS. For details on implementing a PSP integration adapter, refer to the documentation shipped with the `fujitsu-bss-integration-bsp-pack.zip` file.
- Distinguished name retrieved from the certificate of the Web server on which the PSP integration adapter has been deployed. The integration of a PSP with BSS requires the usage of certificates for trusted communication between the platform and the PSP. For details, refer to *Integrating Certificates for Trusted Communication* on page 34.

7.1.3 Configuring PSP Communication and Payment Types

Before a supplier can use the services of a commercial PSP, you need to specify the parameter values required for communicating with the PSP and define the payment types offered by the PSP.

To configure a PSP and define the payment types offered by the PSP, you have the following option:

- BSS user interface - Choose **Manage payment service provider** in the **Operation** menu.
For detailed step-by-step instructions, refer to the online help.

7.1.4 Setting the PSP Identifier for a Supplier

When defining the payment types offered by a PSP for a supplier organization, you need to set the ID provided by the PSP for the supplier. This ID may be, for example, the supplier's account number at the PSP organization.

Suppliers need to have a contractual relationship with the PSP. The PSP configures the channel for a supplier and issues an identifier so that payments can be immediately linked to the supplier's account.

To set the PSP identifier for a supplier, choose one of the following options:

- BSS user interface - Choose **Manage payment service provider** in the **Operation** menu.
For detailed step-by-step instructions, refer to the online help.
- Command line tool - Use the `setpspaccountforsupplier` command with the following syntax:

```
java -jar fujitsu-adm-um-operatorsvc-client.jar <userkey> <password>
setpspaccountforsupplier orgid=<organizationID> pspid=<Name of PSP>
pspidentifier=<ID provided by PSP>
```


7.2 Assigning Payment Types to Suppliers

You are responsible for setting up the payment types that suppliers can offer to their customers. By default, suppliers can allow customers to pay on invoice so that they can subscribe to a service that is not free of charge.

If a supplier requests a payment type other than on invoice, he must use the services of a commercial PSP.

To set the payment types for a supplier, choose one of the following options:

- BSS user interface - Choose **Manage organization** in the **Operation** menu.
For detailed step-by-step instructions, refer to the online help.
- Command line tool - Use the `addavailablepaymenttypes` command with the following syntax:

```
java -jar fujitsu-adm-um-operatorsvc-client.jar <userkey> <password>  
addavailablepaymenttypes orgid=<organizationID>  
paymenttypes=<types>
```

If several payment types are to be set, separate the entries by a comma without any blank. You can enter any payment type as defined when configuring payment types for a PSP. For details, refer to *Configuring PSP Communication and Payment Types* on page 32.

8 Integrating Certificates for Trusted Communication

Certificates are required for BSS to authenticate a calling client and to allow for trusted communication between BSS and an application underlying a technical service or a payment service provider (PSP).

The following organizations are involved in handling certificates:

- Operator
- Technology providers integrating their applications with BSS
- PSPs whose services are to be integrated with BSS for invoicing and payment collection.
- Any other organization using Web service calls to or from BSS

8.1 Introduction

Web service calls coming from BSS (e.g. for provisioning and process integration) or sent to it can be secured with SSL. SSL is used for authentication and for encryption at the transport level.

Every HTTPS connection involves a client and a server. Depending on the calling direction, BSS can act as a server (Web service calls to BSS) or as a client (Web service calls from BSS).

The client must provide its authenticating data to the server. Two options are available:

- **Authentication with basic authentication:**

The caller sends the key and password of a BSS user. The user key can be looked up at the BSS user interface on the **Edit Profile** page. This authentication mechanism is not related to certificates.

When using basic authentication, the access to Web services should be restricted to SSL/TLS communication and HTTPS. Since the BSS APIs are accessible through JNDI lookups, configure your network's firewall to block JNDI lookups from the outside.

- **Authentication with certificates:**

The caller provides a certificate to the BSS server. In this case, the following requirements must be fulfilled:

- The distinguished name (DN) of the client's certificate must correspond to the DN configured and stored in BSS for the corresponding organization.
- BSS must trust the client's certificate: The BSS truststore must contain a certificate with a valid signing chain to the certificate presented by the client.

BSS uses an X.509 certificate to prove the identity of an entity. This certificate is always used to prove the server's identity and optionally to prove the client's identity.

A certificate has a subject which usually identifies the owner of the certificate, and an issuer who signed the certificate. A certificate also includes a validity period. Cryptographic algorithms ensure that the information contained in the certificate cannot be changed without breaking the signature of the certificate.

The subject as well as the issuer is given as a distinguished name (DN) consisting of a list of key-value pairs. One of the standardized keys is called common name (CN). The CN is of particular importance to HTTPS servers: The CN must contain the server's domain; otherwise the client will refuse the connection.

The process of issuing a certificate for another entity is called **signing**. Certificates always form a **chain** up to a certain **root certificate**. In a root certificate, the subject and the issuer are one and the same entity. Such certificates are called "**self-signed**".

Signing certificates or proving that someone is the owner of a certificate requires the possession of the corresponding **private key**. While certificates can be distributed to other parties, special care must be taken to keep the private key secret.

Each client and server may have a keystore and a truststore. A **keystore** is used to keep certificates along with the corresponding private key. This means that a keystore is used to prove your own identity or to sign certificates. A **truststore** contains public certificates of other entities.

8.2 Requirements for Web Service Calls from BSS

For provisioning and process integration, BSS calls other Web services which can be addressed by HTTPS. In this scenario, BSS is the Web service client while the other entity is the HTTPS server. The following requirements must be fulfilled to establish a connection to the server:

- The server must present a valid certificate: The CN must correspond to the server's domain name and it must be valid at the time of calling.
- The client (BSS) must trust the server's certificate. To this end, the client's truststore must contain a certificate with a valid signing chain to the certificate presented by the server.

8.3 Requirements for Web Services Calls to BSS

BSS provides its Web services (platform services) that can be called by other systems. In this scenario, BSS is the HTTPS server while the other system is a Web service client. The following requirements must be met to establish a connection to BSS:

- The BSS server must present a valid certificate: The CN must correspond to the server's domain name and it must be valid at the time of calling.
- The Web service client must trust the server's certificate: The client's truststore must contain a certificate with a valid signing chain to the certificate presented by the server.

8.4 Certificate Integration Procedures

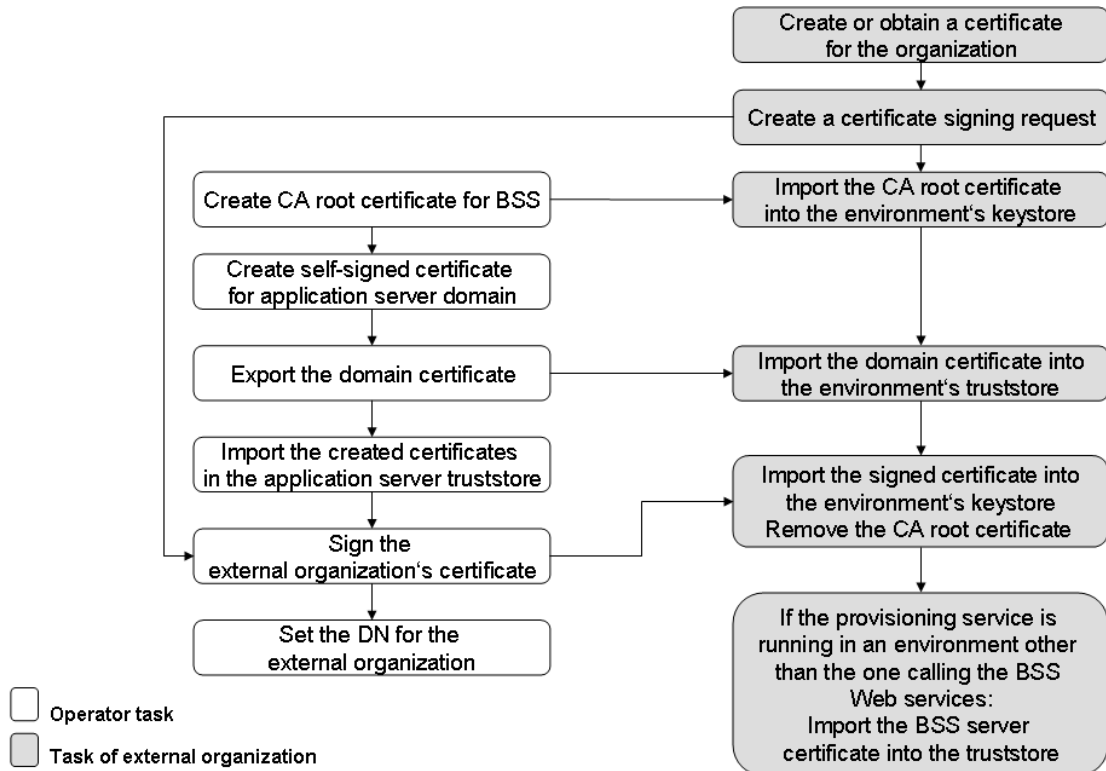
For implementing a secure BSS environment where the nodes trust each other, the usage of custom root certificates is recommended. The basic procedure is as follows:

The BSS operator creates the root certificate. This certificate is imported to the BSS truststore as well as handed out to all technology providers or other organizations calling BSS Web services. Each organization gets its certificates signed by the operator.

With this type of implementation, new applications can be added as technical services without touching the BSS truststore. You must, however, ensure the identity of each organization and manually sign the certificates.

The subsequent sections describe in detail how to set up and use custom root certificates.

The following figure illustrates the process and tasks involved in using custom root certificates for secure communication:



The operator tasks are described in detail in the subsequent sections. The tasks of an external organization that wants to make use of certificate-based authentication are described in the *Developer's Guide*.

Prerequisites

- You have a Java 5 environment installed. This is required for creating, exporting and importing certificates.
- A tool for signing certificates is installed in your environment, for example `OpenSSL` (<http://www.openssl.org/>). The samples mentioned in the next sections assume that `OpenSSL` is installed.
- There are no keystore and truststore files in the application server domain where BSS has been deployed. Check for the files


```
<GLASSFISH_HOME>/domains/<your_domain>/config/cacerts.jks and
<GLASSFISH_HOME>/domains/<your_domain>/config/keystore.jks.
```

 If they are present, delete them.

8.4.1 Creating a CA Root Certificate for BSS

The creation of the CA (Certification Authority) root certificate for BSS involves the following steps:

1. Create a private key with the following `OpenSSL` command:

```
<OPENSSL-Install-Dir>/openssl genrsa -des3 -out ca.key 4096
```

A `ca.key` file is created in the directory where the command was executed. It contains the private key and is required for all subsequent certificate signing operations. `4096` is the number of bits the key will have.

Note: Observe applicable national regulations as to the allowed number of bits the key may have.

2. To create the CA root certificate, execute the following command:

```
<OPENSSL-Install-Dir>/openssl req -new -x509 days 3650 -key ca.key -out ca.crt
```

Using the private key created before, a certificate file (`ca.crt`) for the CA organization is created. The option `-days 3650` states that the created certificate will be valid for 10 years. The default (if you leave the value empty) is 1 month. The certificate file must be provided to all organizations that request their certificate to be signed.

8.4.2 Creating a Self-Signed Domain Certificate

To create a self-signed certificate for the application server domain where BSS has been deployed, execute the following command of the JRE you are using:

```
<YourJRE>/bin/keytool -genkey -alias slas -keysize 1024 -keystore keystore.jks
```

8.4.3 Exporting the Self-Signed Domain Certificate

The self-signed domain certificate needs to be exported and sent to the external organization that wants to make use of certificate-based authentication.

Execute the following command to export the certificate:

```
<YourJRE>/bin/keytool -export -file bssDomain.crt -alias slas -keystore <GLASSFISH_HOME>/domains/<your_domain>/config/keystore.jks
```

The certificate is stored in the `bssDomain.crt` file.

8.4.4 Importing the CA Root and Domain Certificates

You need to import the following certificate files created in the previous steps into the truststore of your application server:

1. `bssDomain.crt`
2. `ca.crt`

Execute the following commands:

```
<YourJRE>/bin/keytool -import -trustcacerts -file bssDomain.crt -alias
slas
-keystore <GLASSFISH_HOME>/domains/<your_domain>/config/cacerts.jks
```

```
<YourJRE>/bin/keytool -import -trustcacerts -file CA.crt -alias CA
-keystore <GLASSFISH_HOME>/domains/<your_domain>/config/cacerts.jks
```

After having imported the certificates, restart the application server.

8.4.5 Signing an Organization's Certificate

A technology provider (or any other external organization that wants to make use of certificate-based authentication) has to provide the certificate for his organization, and a certificate signing request. The BSS operator needs to sign the certificate.

To sign a certificate, use `OpenSSL`. Make sure that you have access to the private key of the BSS CA root certificate and the certificate of the external organization (`tpcert.csr`).

Execute the following command:

```
<OPENSSL-Install-Dir>/openssl x509 -req -days 3650 -in tpcert.csr
-CA ca.crt -CAkey ca.key -set_serial 01 -out tpcert.crt
```

The serial number (01 in the sample above) must be increased with every invocation of the command.

Proceed with sending the output file `tpcert.crt`, the CA root certificate, and the `bssDomain.crt` (domain certificate) to the technology provider.

8.4.6 Setting the Distinguished Name for an Organization

The logic of the BSS call handling checks the distinguished name in a client's certificate. If it matches the one specified by the BSS operator, the caller is assumed to be authorized to connect to BSS.

You can retrieve the distinguished name of an organization from its certificate with the following call in your Java runtime environment:

```
<YourJRE>/bin/keytool -printcert -file <filename.crt>
```

The distinguished name is composed of the following elements:

- CN (common name)
- OU (organizational unit)
- O (organization)
- L (location)
- ST (state)
- C (country)

Sample output:

```
Owner: EMAILADDRESS=test@test.com, CN=test, OU=test, O=Test, L=Test,
ST=Test, C=GE
Issuer: EMAILADDRESS=test@test.com, CN=test, OU=test, O=Test, L=Test,
ST=Test, C=GE
Serial number: 932577b192765bf6
Valid from: Mon Jan 23 16:26:58 CET 2012 until: Thu Jan 20 16:26:58 CET
2022
```

Certificate fingerprints:

```
MD5: 46:2E:CE:05:1A:57:B6:16:15:8A:4B:8A:CE:1D:F8:21  
SHA1: 8A:A6:B9:44:FE:E5:18:0B:92:FD:04:D4:4E:69:0F:09:FB:7B:02:34
```

Note: When integrating the services of commercial payment service providers (PSPs), the distinguished name must always be set. You set the distinguished name when registering the PSP with BSS. For details, refer to *Registering a Payment Service Provider With BSS* on page 32.

To set the distinguished name, choose one of the following options:

- BSS user interface - Choose **Manage organization** in the **Operation** menu.
For detailed step-by-step instructions, refer to the online help.
- Command line tool - Use the `setdistinguishedname` command with the following syntax:

```
java -jar fujitsu-adm-un-operatorsvc-client.jar <userkey> <password>  
setdistinguishedname orgid=<organizationID>  
distinguishedname=<distinguished name>
```

The DN must exactly match the one as retrieved using the `keytool` command (see above).

Sample DN:

```
CN=test, OU=test, O=Test, L=Test, ST=Test, C=GE
```

9 Managing Marketplaces

You are responsible for creating marketplaces for the organizations that want to authorize suppliers to publish their services using the facilities of BSS. Creating a marketplace includes defining the marketplace properties and assigning an organization as the owner of the marketplace. All administrators of the assigned owner organization automatically receive the marketplace manager role. The marketplace manager role enables them to administrate and customize the marketplace. As a prerequisite for creating a marketplace, the organization to be assigned as the marketplace owner must already exist.

For details on administrating and customizing your own marketplaces, refer to the *Marketplace Owner's Guide*.

The operator tasks involved in managing marketplaces are the following:

- Create marketplaces
- Change the owner of a marketplace
- Delete marketplaces
- Use the BSS reporting facilities, for example, to retrieve information on the services published to your marketplace. For details, see chapter *Reporting* on page 29.

Creating a Marketplace

If you want to operate one or several marketplaces on the same platform, you need to create them, and assign an organization as the owner to each of them. The administrator(s) of the owner organization are automatically assigned the marketplace manager role.

To create a marketplace, you have the following option:

- BSS user interface - Choose **Create marketplace** in the **Marketplace** menu.
For detailed step-by-step instructions, refer to the online help.

Changing the Owner of a Marketplace

The owner of a marketplace is responsible for administrating and customizing the marketplace to which suppliers can publish their services. Assigning a new owner to a marketplace may remove the marketplace owner role from the previous owner organization and the marketplace manager role from its users. This is the case if the marketplace for which you change the owner is the last one owned by the organization.

To assign another owner to a marketplace, you have the following option:

- BSS user interface - Choose **Update marketplace** in the **Marketplace** menu.
For detailed step-by-step instructions, refer to the online help.

Deleting a Marketplace

When deleting a marketplace with activated services, these services are automatically deactivated. Customers can no longer subscribe to them. Existing subscriptions, however, are not affected.

To delete a marketplace, you have the following option:

- BSS user interface - Choose **Delete marketplace** in the **Marketplace** menu.
For detailed step-by-step instructions, refer to the online help.

Appendix A: Application Server Resources

The following sections provide an overview of the resources which must be provided to BSS on the application server.

For details and hints on how to create these resources, refer to the `build-glassfish-setup.xml` file, which is part of the setup utilities provided by BSS.

Data Source

A JDBC data source with a corresponding connection pool is required for the relational databases where BSS stores its data.

Name	BSSDS
Class name	<code>org.postgresql.xa.PGXADDataSource</code>
Resource type	<code>javax.sql.XADDataSource</code>
Database schema	As distributed with BSS.

JMS Queue

A JMS queue and a corresponding connection factory are required for asynchronous processing.

Queue name	<code>jms/bss/triggerQueue</code>
Connection factory name	<code>jms/bss/triggerQueueFactory</code>
Maximum number of active consumers	1
Transaction support	<code>XATransaction</code>
Idle timeout	600
User name	<code>jmsuser</code>
Password	<code>jmsuser</code>

Realm

BSS comes with a custom realm implementation in the `fujitsu-bss-security.jar` archive, which is included in the installation package. The realm must be configured as follows:

Name	<code>bss-realm</code>
JAAS context	<code>bssRealm</code>
Implementation class	<code>com.fujitsu.adm.um.security.ADMRealm</code>

Certificate Realm

The certificate realm of the application server must be assigned the role `CLIENTCERT`.

Default File Encoding

Ensure that the default file encoding of the application server's Java virtual machine is UTF-8. This can be achieved by setting the `file.encoding` system property to `UTF8`.

Appendix B: Configuration Settings

The configuration settings for BSS are provided in the following files in the directory to which you extracted the `fujitsu-bss-setup.zip` file (<setup_dir>):

- `configsettings.properties`
- `glassfish.properties`
- `glassfishJMSBrokerConfiguration.properties`
- `db.properties`

This appendix describes the settings in detail.

B.1 BSS Configuration Settings

The `configsettings.properties` file contains the configuration settings for the BSS services.

BASE_URL

Mandatory. The base URL is used to access the BSS landing page. It is particularly required to create the URL for accessing the confirmation page for customers to confirm their registration.

Example: `BASE_URL=http://<host>:<port>/fujitsu-bss-portal`

BASE_URL_HTTPS

Mandatory for PSP integration. The base URL is required to create the URL for accessing services via HTTPS.

Example: `BASE_URL=https://<host>:<port>/fujitsu-bss-portal`

DECIMAL_PLACES

Optional. Specifies the number of decimal places in which usage charges are calculated.

Allowed values: 2, 3, 4, 5, 6

Default: 2

HIDDEN_UI_ELEMENTS

Optional. Specifies menus, pages, or fields to be hidden from the BSS user interface.

If you want to hide a specific page, you can find out which value needs to be specified here as follows:

1. Open the respective page at the BSS user interface.
2. Display the online help for this page.
3. Have a look at the name of the online help HTML page.
4. Omit the file extension `.htm` and replace the underscore by a dot.

Example:

You want to hide the **Manage VAT rates** page. The online help HTML page name is `organization_manageVats.htm`. Thus, the respective user interface page is `organization.manageVats`. You need to set the configuration key as follows:

```
HIDDEN_UI_ELEMENTS=organization.manageVats
```

To hide several pages, separate the entries by a comma.

Below, you find some more examples of values that can be used to hide a specific page or group of fields. The list is not complete.

- `organization.edit`: **Edit profile** page
- `shop.editSkin`: **Customize layout** page
- `techService.edit`: **Update service definition** page

To hide a complete menu, enter one of the following values:

- `navigation.myAccount`: **Account** menu
- `navigation.operator`: **Operation** menu
- `navigation.techService`: **Technical service** menu
- `navigation.service`: **Marketable service** menu
- `navigation.priceModel`: **Price model** menu
- `navigation.marketPlace`: **Marketplace** menu

HTTP_PROXY

Optional. The proxy to be used for HTTP connections, if any.

Example: `proxy.domain` or the proxy server IP address.

HTTP_PROXY_PORT

Optional. The proxy port to be used for HTTP connections, if any.

Allowed values: Any value between 1 and 65535

Default: 1080

IDP_ASSERTION_EXPIRATION

Mandatory. In case a technical service is defined with the USER access type and uses BSS as a SAML Identity Provider, this key must be set. For details on access types, refer to the *Technology Provider's Guide*.

This setting represents the number of milliseconds an assertion generated by the SAML Identity Provider is valid. It is the difference between the `NotAfterOrOn` and `IssueInstant` attributes of the assertion.

Allowed values: Any value between 1 and 9223372036854775807

Default: 1800000 milliseconds (30 minutes)

IDP_ASSERTION_VALIDITY_TOLERANCE

Mandatory. In case a technical service is defined with the USER access type and uses BSS as a SAML Identity Provider, this key must be set. For details on access types, refer to the *Technology Provider's Guide*.

This setting represents the number of milliseconds an assertion generated by the SAML Identity Provider was valid in the past. It is used to avoid synchronization problems between servers. The number is the difference between the `IssueInstant` and `NotBefore` attributes of the assertion.

Allowed values: Any value between 1 and 9223372036854775807

Default: 600000 milliseconds (10 minutes)

IDP_PRIVATE_KEY_FILE_PATH

Mandatory. In case a technical service is defined with the USER access type and uses BSS as a SAML Identity Provider, this key must be set. For details on access types, refer to the *Technology Provider's Guide*.

This setting represents the URL to the private key in Distinguished Encoding Rules (DER) format (.der file). This URL is used by the Identity Provider to sign SAML messages.

Example: /opt/glassfish-2.1.1/domains/integration-bes/config/akey.der

IDP_PUBLIC_CERTIFICATE_FILE_PATH

Mandatory. In case a technical service is defined with the USER access type and uses BSS as a SAML Identity Provider, this key must be set. For details on access types, refer to the *Technology Provider's Guide*.

This setting represents the URL to the public certificate in Distinguished Encoding Rules (DER) format (.der file). This URL is added to the signature of SAML messages by the Identity Provider, so that the application can validate the signature.

Example: /opt/glassfish-2.1.1/domains/integration-bes/config/cacert.der

LDAP_SEARCH_LIMIT

Optional. The maximum number of entries that will be returned by an LDAP query in case an organization uses an external LDAP system for user authentication.

Allowed values: Any value between 1 and 9223372036854775807

Default: 100

LOG_CONFIG_FILE

Optional. The path to the log4j configuration file of BSS.

Default: ./log4j.properties in the <GLASSFISH_HOME>/domains/<your_domain>/config directory

LOG_FILE_PATH

Mandatory. The path to the BSS log files.

Default: ../logs, which is the <GLASSFISH_HOME>/domains/<your_domain>/logs directory

LOG_LEVEL

Optional. The log level for BSS. This setting applies to all logging classes if it is not overridden by the content of the log4j.properties file.

Allowed values: ERROR, WARN, INFO, DEBUG

Default: INFO

MAIL_JA_CHARSET

Optional. Special character encoding for emails sent in Japanese.

Default: UTF-8

MAIL_PORT

Deprecated - use the setting in the `glassfish.properties` file instead.

MAIL_RESPONSE_ADDRESS

Deprecated - use the setting in the `glassfish.properties` file instead.

MAIL_SERVER

Deprecated - use the `MAIL_HOST` setting in the `glassfish.properties` file instead.

MAIL_USER

Deprecated - use the setting in the `glassfish.properties` file instead.

MAIL_USER_PWD

Deprecated - use the setting in the `glassfish.properties` file instead.

MAX_NUMBER_LOGIN_ATTEMPTS

Optional. The maximum number of allowed login attempts to BSS. If a user does not log in successfully with this number of attempts, his account is locked.

Allowed values: Any value between 1 and 9223372036854775807

Default: 3

PERMITTED_PERIOD_INACTIVE_ON_BEHALF_USERS

Optional. The time after which a user who logged in on behalf of a customer and was inactive will be removed from the system. The value is in milliseconds.

Allowed values: Any value between 1 and 9223372036854775807

Default: 604800000, i.e. 7 days

PERMITTED_PERIOD_UNCONFIRMED_ORGANIZATIONS

Optional. The maximum time until an organization's initial administrative account must be confirmed. When this time has passed, the account is removed. The value is in milliseconds.

Allowed values: Any value between 1 and 9223372036854775807

Default: 604800000, i.e. 7 days

PSP_USAGE_ENABLED

Mandatory. Specifies whether PSP integration is used for the current environment.

Allowed values: `true`, `false`

Default: `false`

REPORT_ENGINEURL

Mandatory if you want to use the BSS reports. The URL template of the report engine. If you do not specify a correct URL template, BSS will not be able to generate any reports, since the Report Web service cannot be called correctly.

The required value is:

```
http://<host>:<port>/eclipse-birt-runtime/frameset?__report=$
  {reportname}.rptdesign&SessionId=$
  {sessionid}&__locale=$
  {locale}&WSDLURL=$
  {wsdlurl}&SOAPEndPoint=${soapendpoint}
```

Note: If you specified the `BASE_URL` host by an IP address, also specify the `REPORT_ENGINEURL` host by an IP address. Alternatively, use the server name for both settings.

REPORT_SOAP_ENDPOINT

Mandatory if you want to use the BSS reports. The SOAP end point of the Report Web service. All report data is retrieved via a call to the Report Web service. If you do not specify a correct value, BSS will not be able to generate any reports, since the Report Web service cannot be called correctly.

Required value: `http://<host>:<port>/Report/ReportingServiceBean`

REPORT_WSDLURL

Mandatory if you want to use the BSS reports. The URL of the WSDL file of the Report Web service. All report data is retrieved via a call to the Report Web service. If you do not specify a correct value, BSS will not be able to generate any reports, since the Report Web service cannot be called correctly.

Required value: `http://<host>:<port>/Report/ReportingServiceBean?wsdl`

SUPPLIER_SETS_INVOICE_AS_DEFAULT

Optional. Specifies whether invoice is to be used as the default payment type for all customers.

Allowed values: `true, false`

Default: `false`

TAGGING_MAX_TAGS

Mandatory. The maximum number of tags composing the tag cloud.

The tag cloud is the area of a marketplace containing defined search terms (tags). The more often a tag is used in services, the bigger the characters of the tag are displayed. Customers can use the tags to search for services, provided that the tag cloud is enabled for the marketplace by the marketplace owner.

Allowed values: Any value between 0 and 2147483647

Default: 20

TAGGING_MIN_SCORE

Mandatory. The minimum number of times a tag must be used in services to be shown in the tag cloud.

The tag cloud is the area of a marketplace containing defined search terms (tags). The more often a tag is used in services, the bigger the characters of the tag are displayed. Customers can use the tags to search for services, provided that the tag cloud is enabled for the marketplace by the marketplace owner.

Allowed values: Any value between 1 and 2147483647

Default: 1, i.e. a tag must have been used at least once so that it is shown in the tag cloud.

TIME_ZONE_ID

Optional. The time zone to be used for display.

Allowed values: All time zones supported by Java. This can be an abbreviation such as `PST`, a full name such as `America/Los_Angeles`, or a custom ID such as `GMT-8:00`. For a list of IDs, refer to `java.util.TimeZone`.

Default: `GMT`

TIMER_INTERVAL_BILLING_OFFSET

Optional. The offset for the timer for billing runs. The value is in milliseconds, based on January 1, 00:00.

Allowed values: Any value between 0 and 9223372036854775807

Default: 345600000, i.e. 4 days. This means that a billing run starts on the 5th of every month.

TIMER_INTERVAL_DISCOUNT_END_NOTIFICATION_OFFSET

Optional. The offset for the timer for terminating the discounts for all organizations. The timer interval is one day and cannot be changed. The offset is in milliseconds, based on January 1, 00:00.

Allowed values: Any value between 0 and 9223372036854775807

Default: 0

TIMER_INTERVAL_INACTIVE_ON_BEHALF_USERS

Optional. The time interval at which a check for non-existing users acting on behalf of another organization is executed. The value is in milliseconds. A value of 0 indicates that this timer is disabled.

A technical service definition may contain a flag (`allowingOnBehalfActing`) to indicate that an organization can act in the name of another organization. The organization must be a customer of the other organization, which must have both the technology provider and supplier role. Additionally, the customer organization must have allowed the other organization to log in on its behalf. This is achieved via a subscription whose underlying technical service has the `allowingOnBehalfActing` flag set to `true`.

When an organization acts in the name of another organization, an artificial user ID is generated.

Cleaning up the BSS database from time to time to remove such users who no longer exist might be required since it cannot be ensured that a technical service always removes such users itself.

Allowed values: Any value between 0 and 9223372036854775807

Default: 0

TIMER_INTERVAL_INACTIVE_ON_BEHALF_USERS_OFFSET

Optional. The offset for the timer for removing inactive "on behalf" users. The offset is in milliseconds, based on January 1, 00:00.

Allowed values: Any value between 0 and 9223372036854775807

Default: 0

TIMER_INTERVAL_ORGANIZATION

Optional. The time interval at which tasks related to organizations are executed. The value is in milliseconds. A value of 0 indicates that this timer is disabled.

Allowed values: Any value between 0 and 9223372036854775807

Default: 0

TIMER_INTERVAL_ORGANIZATION_OFFSET

Optional. The offset for the timer for organization-related tasks. The value is in milliseconds, based on January 1, 00:00.

Allowed values: Any value between 0 and 9223372036854775807

Default: 0

TIMER_INTERVAL_SUBSCRIPTION_EXPIRATION

Optional. The time interval at which a check for expired subscriptions is executed. The value is in milliseconds. This timer cannot be disabled, i.e. it cannot be set to 0.

Allowed values: Any value between 1 and 9223372036854775807

Default: 86400000, i.e. 1 day

TIMER_INTERVAL_SUBSCRIPTION_EXPIRATION_OFFSET

Optional. The offset for the timer for subscription expiration checks. The value is in milliseconds, based on January 1, 00:00.

Allowed values: Any value between 0 and 9223372036854775807

Default: 0

TIMER_INTERVAL_TENANT_PROVISIONING_TIMEOUT

Optional. The time interval at which a check for timed-out subscriptions is executed. The value is in milliseconds. A value of 0 indicates that this timer is disabled.

Allowed values: Any value between 0 and 9223372036854775807

Default: 0

TIMER_INTERVAL_TENANT_PROVISIONING_TIMEOUT_OFFSET

Optional. The offset for the timer for pending subscription checks. The value is in milliseconds, based on January 1, 00:00.

Allowed values: Any value between 0 and 9223372036854775807

Default: 0

WS_TIMEOUT

Mandatory. The timeout for outgoing Web service calls in milliseconds. After this time has passed, a timeout exception is thrown by the JAX-WS framework.

An outgoing Web service call is a call initiated by BSS. A typical example is the invocation of the `createUsers` method of the `ProvisioningService` interface, which is implemented by an application. If the timeout is reached before the Web service call returns, the operation is aborted and an exception is thrown.

Allowed values: Any value between 1 and 9223372036854775807

Default: 30000, i.e. 30 seconds

Note: Make sure that timeouts set in the GlassFish application server, e.g. request timeouts, do not conflict with or overrule the timeouts defined in the BSS configuration settings.

B.2 GlassFish Configuration Settings

The `glassfish.properties` file contains the configuration settings for the GlassFish application server. The settings are required for configuring your GlassFish domain for BSS.

Below you find a detailed description of the settings.

GF_ADMIN_DOMAIN_PORT

The administration port of the GlassFish domain used for BSS.

Example: 4848

GLASSFISH_HOME

The absolute path and name of the GlassFish installation directory.

JDBC_DRIVER_JAR_PATH

The absolute path of the PostgreSQL JDBC driver jar as available after installation.

Example: `<setup_dir>/lib`

JDBC_DRIVER_JAR_NAME

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

Example: `postgresql-8.4-701.jdbc3.jar`

GF_BROKER_DESTDIR

The absolute path and name of the directory of the message broker valid for your GlassFish domain.

Example: `<GLASSFISH_HOME>/domains/<your_domain>/imq/instances/imqbroker`

GF_DOMAIN_NAME

The name of the GlassFish domain where you deployed BSS.

GF_DOMAIN_ADMIN_USER

The user name of the GlassFish administrator.

GF_DOMAIN_ADMIN_PWD

The password of the GlassFish administrator.

GF_WS_PORT

The port used for an HTTP listener for Web service connections of the application server.

Example: 8081

MAIL_HOST

The host name or IP address of your mail server. This setting is required for the application server mail resource.

MAIL_PORT

The port of your mail server.

Default: 25

MAIL_RESPONSE_ADDRESS

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

Example: `saas@yourcompany.com`

MAIL_USE_AUTHENTICATION

Optional. Defines whether mails can only be sent 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.

B.3 GlassFish JMS Configuration Settings

The `glassfishJMSBrokerConfiguration.properties` file contains the configuration settings for the Java Message Service (JMS) in the application server.

For details, refer to the *Sun GlassFish Message Queue 4.4 Administration Guide*.

B.4 Database Configuration Settings

The `db.properties` file contains the configuration settings for 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_URL

The connection URL of the database.

Default: `jdbc:postgresql://localhost:5432/bss`

DB_HOST

The database host.

Default: `localhost`

DB_PORT

The database port.

Default: `5432`

DB_NAME

The name of the database.

Default: `bss`

DB_USER

The name of the user to connect to the database.

Default: `bssuser`

DB_PWD

The password of the user to connect to the database.

Default: `bssuser`

DB_TYPE

The type of the database.

Default: `postgresql`

Appendix C: User Operation Log

The user operation log shows all user operations related to subscriptions on your platform. Single entries of the user operation log consist of a **header** and a **message**, separated by a comma. The header consists of the following elements, separated by a blank:

- `MM/DD/YYYY_HH:MM:SS.FFF`: The server date and time.
- `FSP_INTS-BSS`: The predefined log label for log entries resulting from user operations on your platform. This label cannot be changed.
- `INFO`: The default log level. This cannot be changed.
- `3<nnnn>`: The ID of the log message. This is a number between 30001 and 39999. The ID is unique for each message type. For example, the ID of the `SUBSCR` message type is 30001, of `SUBSCR_USER`, it is 30002.

Following the header, the log message is appended. It consists of name-value pairs with additional information, separated by commas.

Every message starts with the following name-value pairs:

- `log, <message type>`: The `<message type>` can either be `SUBSCR` (subscription-related operations), `SUBSCR_USER` (subscription user-related operations), or `SUBSCR_PRICE` (price model-related operations). See below for details.
- `op, ADD | MODIFY | DELETE`: The type of operation that has been recorded in the log. `MODIFY` means, for example, that a user has changed a subscription.
- `user, <user id>`: The ID of the user who executed the operation.

Following the above name-value pairs, a common set of customer-specific information is appended to the message. This common set is the same for all message types. Finally, additional name-value pairs specific to the message type are appended to the message. These are described in detail below.

Common Set

The common set of information appended to log messages of all types consists of customer-specific information:

Name	Value
subscription	Name of a subscription as entered when subscribing to a service.
customer	Name of the organization that subscribed to a service.
customer id	ID of the organization that subscribed to a service.

Example:

```
<Header>,log,SUBSCR_PRICE,op,MODIFY,user,smith
,subscription,Docs2Go Gold - dept. PM,customer,BMW,customer id,90349ce5
,<additional name-value pairs depending on the message type>
```

Message Type: SUBSCR

This message type provides additional name-value pairs with subscription details.

Name	Value
service	Name in the database of the marketable service subscribed to.
activation	Activation date of the subscription in the format MM/DD/YYYY_HH:MM:SS.FFF.
status	Status of the subscription. Can be ACTIVE, PENDING, DEACTIVATED, EXPIRED, INVALID, or SUSPENDED.
deactivation	Deactivation date of the subscription in the format MM/DD/YYYY_HH:MM:SS.FFF.
marketplace	ID of the marketplace to which the marketable service has been published.
payment type	Name of the payment type defined for the customer organization that subscribed to the service.
billing contact	Name of the billing contact defined by the customer organization that subscribed to the service.
reference	Reference number, e.g. purchase order number, specified when subscribing to the service.
access	Information on how to access the application underlying the subscription.
url	Base URL defined for the service.
login path	Login path defined for the service.
timeoutmailed	Flag indicating whether the sending of an email has timed out. Can be TRUE OR FALSE.

Example:

```
<Header><Common Set>,  
service, Docs2Go Gold, activation, 09/21/2011_12:03:24.231, status, DEACTIVATED,  
deactivation, 09/21/2011_15:46:20.859, marketplace, FUJITSU,  
payment type, Invoice, billing contact, my home address, reference, 4711,  
access, PLATFORM, url, http://myServer.domain.com, login  
path, /login, timeoutmailed, FALSE
```

Message Type: SUBSCR_USER

This message type provides additional name-value pairs with information on a user assigned to a subscription:

Name	Value
user id	ID of the user assigned to a subscription.
first name	First name of the user assigned to a subscription.
last name	Last name of the user assigned to a subscription.
email	Email address of the user assigned to a subscription.
app user id	If defined for the underlying application: ID of the user used in the application for the platform user.

Name	Value
app user role	If defined for the underlying application: Service role of the user.

Example:

```
<Header><Common Set>,
user id,smith,first name,John,last
name,Smith,email,John.Smith@myCompany.com,
app user id,jsmith,app user role,Guest
```

Message Type: SUBSCR_PRICE

This message type provides additional name-value pairs with information on a price model defined for the marketable service that has been subscribed to:

Name	Value
free	Specifies whether the usage of the underlying service is free of charge. Can be YES or NO.
one time fee	One time fee as defined in the price model.
period	Period for which a recurring charge has been defined in the price model. Can be MONTH, WEEK, DAY, or HOUR.
price	Recurring charge per subscription as defined in the price model.
price per user	Recurring charge per user as defined in the price model.
currency	ISO currency code of the currency in which the prices of the price model are defined.

Examples:

```
<Header><Common Set>,
free,NO,one time fee,100.00,price,5.00,price per user,0.00,currency,EUR
```

```
<Header><Common Set>,
free,YES,one time fee,,price,,price per user,,currency,
```

Appendix D: Menu Options and Required Roles

The roles of an organization determine which functions are available to its users at the BSS user interface and which roles the users can be assigned. The user roles control the actions an individual user is allowed to execute.

This appendix provides a list of the available user interface functions and shows which user of which organization is allowed to execute a function.

Account Menu

This menu is available to users of all organizations:

Function	Organization Role	User Role
Edit profile	Any	Any
Change password	Any	Any
Register new users	Any	Administrator
Manage users	Any	Administrator
Create report	Any	Administrator
Process triggers	Any	Administrator
Manage processes	Any	Any
Manage suppliers	Technology provider	Technology manager
Manage custom attributes	Supplier	Service manager

Customer Menu

This menu is available to supplier organizations only.

Function	Organization Role	User Role
Register customer	Supplier	Service manager
Manage customer	Supplier	Service manager
Manage payment types	Supplier	Service manager
Manage VAT rates	Supplier	Service manager
Export billing data	Supplier	Service manager
View subscription	Supplier	Service manager
Edit subscription attributes	Supplier	Service manager
Terminate a subscription	Supplier	Service manager

Marketable Service Menu

This menu is available to supplier organizations only.

Function	Organization Role	User Role
Define a service	Supplier	Service manager
Update a service	Supplier	Service manager
Copy a service	Supplier	Service manager
Delete a service	Supplier	Service manager
Define up/downgrade options	Supplier	Service manager
Activate or deactivate services	Supplier	Service manager

Marketplace Menu

This menu is available to marketplace owner and operator organizations.

Function	Organization Role	User Role
Manage categories	Marketplace owner	Marketplace manager
Manage suppliers	Marketplace owner	Marketplace manager
Create marketplace	Operator	Operator
Update marketplace	Marketplace owner or operator	Marketplace manager or operator. The operator can assign a different marketplace owner, the marketplace manager can change the name and settings of the marketplace.
Delete marketplace	Operator	Operator
Customize stage	Marketplace owner	Marketplace manager
Customize texts	Marketplace owner	Marketplace manager
Customize layout	Marketplace owner	Marketplace manager

Operation Menu

This menu is available to operator organizations only.

Function	Organization Role	User Role
Manage users	Operator	Operator
Create organization	Operator	Operator
Create payment service provider	Operator	Operator
Manage organization	Operator	Operator
Manage timers	Operator	Operator

Function	Organization Role	User Role
Manage payment service provider	Operator	Operator
Manage currencies	Operator	Operator
Update configuration settings	Operator	Operator
Export billing data	Operator	Operator
Execute billing tasks	Operator	Operator
Export operation log	Operator	Operator

Price Model Menu

This menu is available to supplier organizations only.

Function	Organization Role	User Role
Define for service	Supplier	Service manager
Define for customer	Supplier	Service manager
Delete for customer	Supplier	Service manager
Define for subscription	Supplier	Service manager

Technical Service Menu

This menu is available to technology provider organizations only.

Function	Organization Role	User Role
Register technical service	Technology provider	Technology manager
Import technical service	Technology provider	Technology manager
Update technical service	Technology provider	Technology manager
Export technical service	Technology provider	Technology manager
Delete technical service	Technology provider	Technology manager

Glossary

Administrator

A privileged user role within an organization. Each organization has at least one administrator.

Application

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

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 BSS 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 BSS, based on a technical service. A marketable service defines prices, conditions, and restrictions for using the underlying application.

Marketplace

A virtual platform for suppliers in BSS to provide their services to customers.

Marketplace Owner

An organization which holds a marketplace in BSS, where one or more suppliers 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 BSS.

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, marketplace owner, operator.

Payment Service Provider (PSP)

A company that offers suppliers 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 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 BSS; the supplier 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.

Role

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

Service

Generally, a discretely defined set of contiguous or autonomous business or technical functionality, for example, an infrastructure or Web service. BSS 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 BSS. By subscribing to a service, the customer is given access to the underlying application under the conditions defined in the marketable service.

Supplier

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

Technical Service

The representation of an application in BSS. 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 BSS.