FUJITSU Software
Systemwalker
Service Catalog Manager V15.3.0

Operator's Guide

Contents

About this Manual

This manual describes the basic tasks involved in the operation and maintenance of FUJITSU Software Systemwalker Service Catalog Manager - hereafter referred to as Catalog Manager (CT-MG).

The manual is structured as follows:

Chapter	Description
Introduction on page 9	Describes CT-MG and its architecture, its organizations and users.
Getting Started on page 13	Provides an overview of the tasks involved in the operation and maintenance of CT-MG, shows the initial steps to be performed after installing CT-MG, and introduces the tools that support you in performing operational tasks.
Configuring CT-MG on page 20	Describes how to maintain the CT-MG configuration settings, how to tune system performance, how to add a currency, and how to use and configure timers for automatic task processing.
Monitoring, Backup and Recovery on page 27	Describes the CT-MG log files and log entry types, how to configure the log level, as well as which data is subject to regular backups. In addition, it describes how to export the audit log data and how to monitor the JMS queues.
Managing Organizations and Users on page 32	Describes how to create and manage organizations and user accounts.
Reporting on page 36	Describes the reports available for operators in CT-MG.
Managing Billing and Payment on page 38	Describes how to define revenue shares, how to set up the payment types allowed for suppliers and resellers, and how to integrate CT-MG with a payment service provider (PSP). In addition, it describes how to handle billing data: how to restart payment processes and billing runs, preview and export billing data.
Integrating Certificates for Trusted Communication on page 46	Describes the usage of certificates for secure communication between CT-MG and applications integrated with it.
Managing Marketplaces on page 53	Describes how to create, update, and delete marketplaces.
Application Server Resources on page 55	Describes the resources required for CT-MG on the application server.
Configuration Settings on page 58	Describes the CT-MG configuration settings.

Chapter	Description
Audit Log on page 76	Describes the elements of the audit log.
Menu Options and Required Roles on page 97	Gives an overview of the CT-MG administration portal menu options and the roles required for using them.

Readers of this Manual

This manual is directed to operators who maintain and operate CT-MG 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 technology, particularly as to the deployment on application servers.
- Relational databases and their administration, in particular the PostgreSQL database.
- CT-MG 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.
`init`	System names, for example, command names, and text that is entered from the keyboard are shown in Courier font.
`<variable>`	Variables for which values must be entered are enclosed in angle brackets.
`[option]`	Optional items, for example, optional command parameters, are enclosed in square brackets.
`one	two`	Alternative entries are separated by a vertical bar.
`{one	two}`	Mandatory entries with alternatives are enclosed in curly brackets.

Abbreviations

This manual uses the following abbreviations:

API	Application Programming Interface
CT-MG	Catalog Manager
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
SAML	Security Assertion Markup Language
SOAP	Simple Object Access Protocol
STS	Security Token Service
WSDL	Web Services Description Language
XSD	XML Schema Definition

Available Documentation

The following documentation on CT-MG is available:

- *Overview:* A PDF manual introducing CT-MG. It is written for everybody interested in CT-MG and does not require any special knowledge.
- *Online Help:* Online help pages describing how to work with the administration portal of CT-MG. The online help is directed and available to everybody working with the administration portal.
- *Installation Guide:* A PDF manual describing how to install and uninstall CT-MG. It is directed to operators who set up and maintain CT-MG in their environment.
- *Operator's Guide:* A PDF manual for operators describing how to administrate and maintain CT-MG.
- *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 CT-MG.
- *Supplier's Guide:* A PDF manual for suppliers describing how to define and manage service offerings for applications that have been integrated with CT-MG.
- *Reseller's Guide:* A PDF manual for resellers describing how to prepare, offer, and sell services defined by suppliers.
- *Broker's Guide:* A PDF manual for brokers describing how to support suppliers in establishing relationships to customers by offering their services on a marketplace.
- *Marketplace Owner's Guide:* A PDF manual for marketplace owners describing how to administrate and customize marketplaces in CT-MG.
- *Developer's Guide:* A PDF manual for application developers describing the public Web service interface of CT-MG and how to use it to integrate applications and external systems with CT-MG.
- Javadoc documentation for the public Web service interface of CT-MG and additional resources and utilities for application developers.

1 Introduction

Catalog Manager (CT-MG) 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, CT-MG 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.

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

1.1 CT-MG Architecture

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

The following figure provides an overview of the architecture:
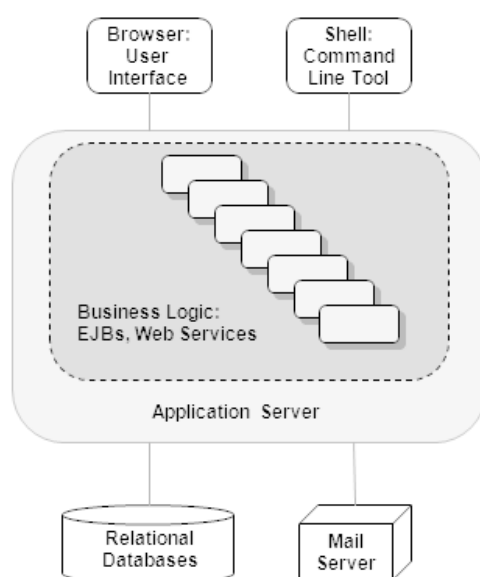
CT-MG has a three-tier architecture:

- The **presentation layer** in the application server's Web container includes the **user interface** (administration portal and marketplaces), realized as JavaServer Faces. Users access the user interface in Web browsers. In addition, CT-MG provides a **command line tool**, which operators can install to carry out their tasks from a command shell instead of the administration portal.

- 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 CT-MG. However, it can also be employed for accessing CT-MG functionality from a Web service client. HTTPS must be used for communication with the public Web service interface.
- CT-MG **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), CT-MG must have access to a mail server.

1.2   Organizations and User Roles

Each user working in CT-MG 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 CT-MG has a unique account and ID as well as one or more of the following roles: **operator**, **technology provider**, **supplier**, **reseller**, **broker**, **marketplace owner**, **customer**.

Customers can register themselves with CT-MG or be registered by a supplier, reseller, broker, or operator. In any case, 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 CT-MG is installed, an organization with the operator role is created.
- Operators can assign the supplier, reseller, broker, and technology provider role to any existing organization or create new organizations with these roles. An organization can have both the supplier and the technology provider role. The reseller and broker roles, however, cannot be combined with each other or with the supplier or technology provider role.
- When operators create a marketplace, they specify an existing organization as its owner. 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 CT-MG 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 CT-MG 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.
- **Subscription manager**: This role allows a user to subscribe to services and manage his own subscriptions. Unlike administrators, subscription managers are not permitted to work on subscriptions belonging to others or on subscription data related to billing and payment.
- **Technology manager**: This role allows a user to define technical services. 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.
- **Reseller**: This role allows a user to publish a supplier's marketable services applying different terms and conditions. It can be assigned to users of reseller organizations.

- **Broker**: This role allows a user to publish a supplier's marketable services without changing the terms and conditions defined by the supplier. It can be assigned to users of broker organizations.
- **Marketplace manager**: This role allows a user to define the organizations 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 CT-MG is installed.

The following illustration provides an overview of how organizations with the different roles are created and related with each other:
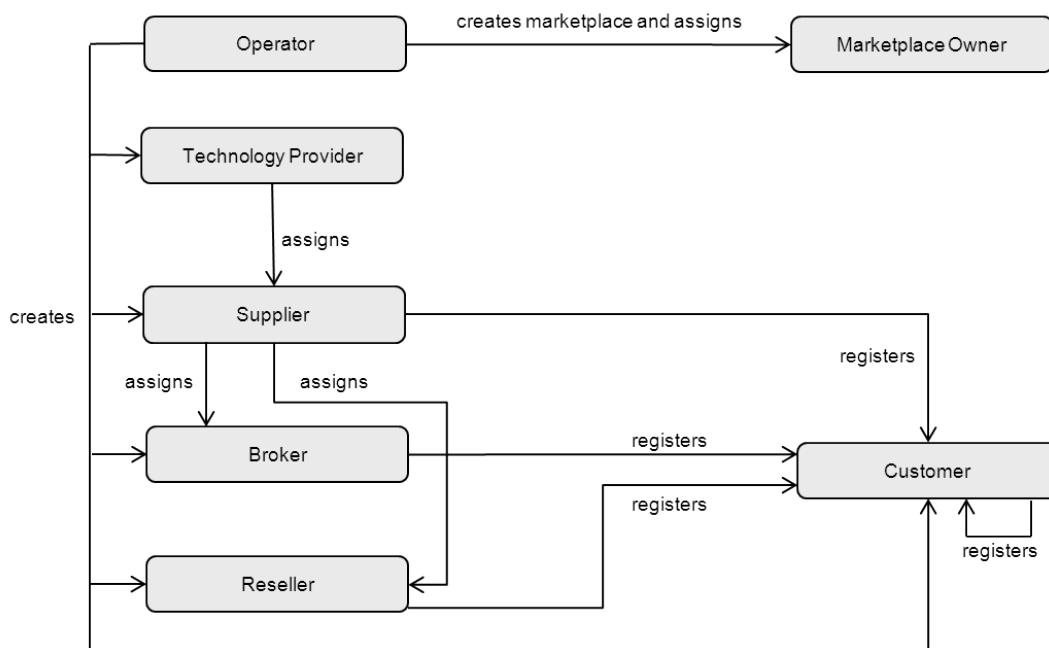
The following illustration provides an overview of the user roles and the main tasks of users with these roles:
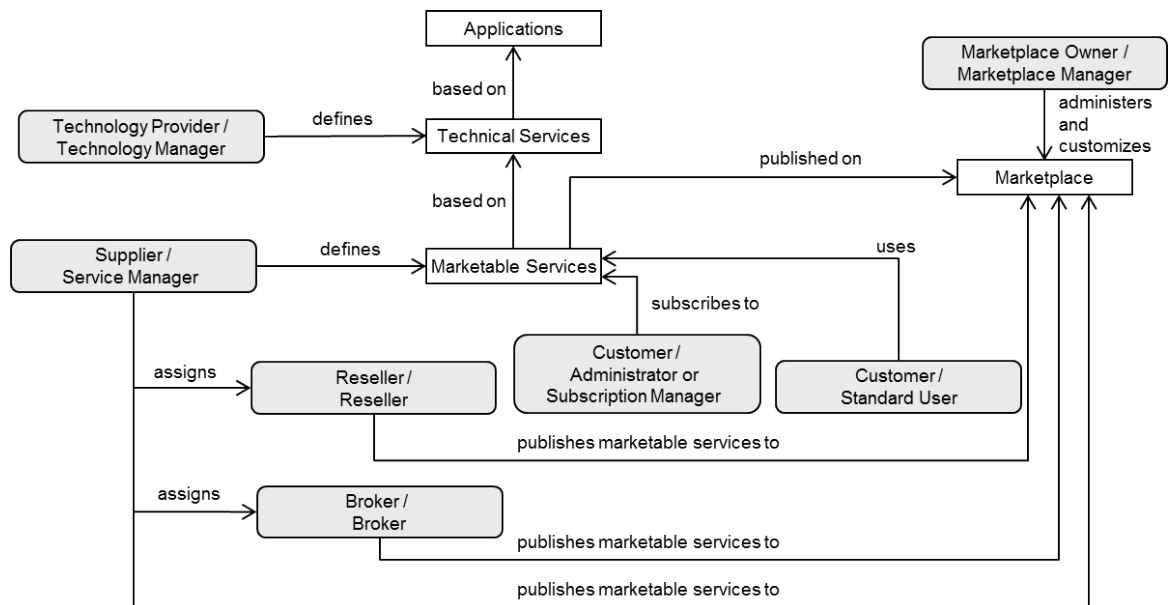
2 Getting Started

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

2.1 Overview of Operation and Maintenance Tasks

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

Regular	System-related	On Demand
	Set up the command line tool	**Organization-related**
	Add currency support	Manage organizations and users
	Startup/Shutdown	Reporting
	Configure general settings	**Billing and payment-related**
	Configure timers	Define revenue shares
Backup and recovery	Manage LDAP settings	Integrate PSP support
Monitoring	Tune performance	Assign payment types to sellers
Check JMS queues	Configure the log level	Preview billing data
	Maintain the log files	Export revenue share data
	Export audit log data	**Service-related**
	Restart payment processes	Integrate certificates
	Start billing run	**Marketplace-related**
		Manage marketplaces

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

Task	Described in
Backup and recovery	*Backup and Recovery* on page 30
Monitoring	*Monitoring CT-MG* on page 27
Check the JMS queues	*Checking the Load on the JMS Queues* on page 31
Set up the command line tool	*Tool Support for Operating CT-MG* on page 16
Add currency support	*Adding a Currency to CT-MG* on page 21

Task	Described in
Startup/Shutdown	*Startup and Shutdown* on page 15
Configure general settings	*Updating Configuration Settings* on page 20
Configure timers	*Configuring Timers* on page 22
Manage LDAP settings	*Managing LDAP Settings* on page 24
Tune performance	*Tuning Performance* on page 25
Configure the log level	*Configuring the Log Level* on page 27
Maintain the log files	*Maintaining the Log Files* on page 29
Export audit log data	*Exporting Audit Log Data* on page 29
Restart payment processes	*Handling Billing Data* on page 43
Start billing run	*Handling Billing Data* on page 43
Manage organizations and users	*Managing Organizations and Users* on page 32
Reporting	*Reporting* on page 36
Define revenue shares	*Defining Revenue Shares* on page 38
Integrate PSP support	*Integrating PSP Support* on page 40
Assign payment types to sellers	*Assigning Payment Types to Suppliers and Resellers* on page 43
Preview billing data	*Handling Billing Data* on page 43
Export revenue share data	*Handling Billing Data* on page 43
Integrate certificates	*Integrating Certificates for Trusted Communication* on page 46
Manage marketplaces	*Managing Marketplaces* on page 53

2.2 First Steps

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

```
┌─────────────────────┐
│     Install BSS      │
└─────────────────────┘
           │
           ▼
┌─────────────────────┐
│    Configure BSS     │
└─────────────────────┘
           │
           ▼
┌─────────────────────┐
│  Add a currency to BSS │
└─────────────────────┘
           │
           ▼
┌─────────────────────┐
│  Create organizations │
└─────────────────────┘
           │
           ▼
┌─────────────────────┐
│  Create marketplaces │
└─────────────────────┘
```
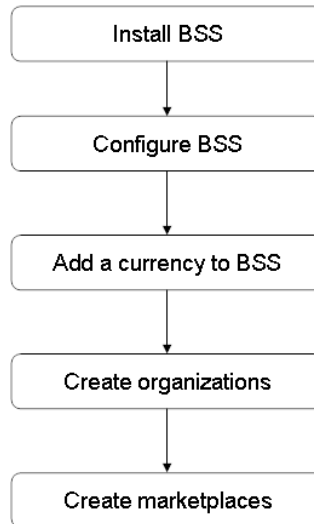
When you have successfully installed and configured CT-MG following the instructions in the *Installation Guide*, you have access to the CT-MG administration portal and a command line tool. For a description on how to access the available tools, refer to *Tool Support for Operating CT-MG* on page 16.

In order for a supplier to be able to define price models and for the rating and billing engine of CT-MG 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 CT-MG* on page 21.

In the next step, you can create technology provider and seller organizations (suppliers, resellers, brokers), and set up marketplaces so that services can be offered to customers. For details, refer to *Managing Organizations and Users* on page 32 and *Managing Marketplaces* on page 53.

2.3 Startup and Shutdown

When you have completed the installation of CT-MG and provided the required resources on the application server as described in the *Installation Guide*, the following domains with the following applications are available on the application server:

- `bes-domain` domain with the `fujitsu-bss` Enterprise Application (the CT-MG core software) and the `fujitsu-bss-portal` application for accessing the CT-MG administration portal and marketplaces.

- `master-indexer-domain` domain with the `fujitsu-bss-search` Enterprise Application of the search indexer application.

The domains and applications can be started and accessed using the standard utilities of the application server.

A shutdown of the domains and applications is also performed using the standard functionality of the application server.

When stopping the domains, always stick to the following sequence:

1. Stop the `bes-domain` domain.
2. Stop the `master-indexer-domain` domain.

When restarting the domains, always stick to the following sequence:

1. Restart the `master-indexer-domain` domain.
2. Restart the `bes-domain` domain.

2.4 Tool Support for Operating CT-MG

CT-MG provides operator functions in its administration portal 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.

CT-MG Administration Portal

You can access the CT-MG administration portal in a Web browser using an URL in the following format:

`http://<server>:<port>/<context-root>`

`<server>` is the application server where CT-MG has been deployed. `<port>` is the port to address the application server (default: `8080` for HTTP, `8081` for HTTPS). `<context-root>` is the context root of CT-MG (default: `fujitsu-bss-portal`).

You are prompted for the user ID and password. The initial credentials are as follows:

- **INTERNAL** authentication mode:

    User ID: `administrator`

    Password: `admin123`

    It is recommended that you change the initial password in the CT-MG administration portal (**Change Password** page in the **Account** menu).

- **SAML_SP** authentication mode:

    User ID: The ID you specified in the `ADMIN_USER_ID` configuration key in the `sso.properties` configuration file before installing CT-MG.

    Password: Password as known in the IdP system for the above user. You can change your password in the external authentication system only.

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

Refer to *Authentication* on page 17 for details on authentication modes.

Command Line Tool

The command line tool provides a subset of the functions that are available in the **Operation** menu in the CT-MG administration portal.

The command line tool is provided in the CT-MG 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 directory on the system where you have installed CT-MG. 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.

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.

For executing a command, change to the directory to which you extracted the `fujitsu-bss-operatorclient.zip` file.

Command Syntax:

The syntax of all available commands is as follows:

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

where

`<JAVA_HOME>` is the installation directory of your JDK (for example `C:\XXX\jdk1.7.0_09`).

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

`password` is the password of the operator. The initial passwort is `admin123`.

`command` is the command to be executed.

`parameter` is a command parameter to be set.

When using the command line tool, you are not authenticated against the IdP. You need to use the user key `1000` and the initial password `admin123`. You can change the initial password in the CT-MG administration portal (**Change Password** page in the **Account** menu).

2.5 Authentication

The authentication mode in which CT-MG is installed, determines how and where the credentials of users are verified. It also determines the initial credentials of the first operator created by the installation.

The following authentication modes are supported:

- **INTERNAL**:

   CT-MG is installed as a platform for public access from anywhere in the Internet. Users are authenticated with CT-MG and can be managed in CT-MG or an existing LDAP system of an organization. Web service calls are formatted as SOAP messages and are authenticated in CT-MG either by providing user ID and password in their header, or by certificates.

- **SAML_SP**:

   CT-MG is installed as a SAML 2.0 service provider.

   SAML (Security Assertion Markup Language) is an XML-based protocol that uses SAML assertions to pass information about a user between a SAML identity provider (IdP) and a SAML service provider (CT-MG). With SAML 2.0, Web browser single sign-on within a company is provided.

   All users are authenticated with the IdP; Web service calls are authenticated by means of a Security Token Service (STS). This is a Web service that issues security tokens as defined in the WS-Security/WS-Trust specification. CT-MG as a service provider connects to the

IdP or STS and sends a request for a SAML assertion. The IdP or STS returns an assertion authenticating the calling user or Web service.

The user data is managed in the IdP system. Additionally, all users that are to work with CT-MG must be registered explicitly in CT-MG.

Customer self-registration is not supported in this case, and can thus be disabled.

INTERNAL Authentication Mode

What is accessed?	Authentication by	Authentication via	Initial credentials
Administration portal	CT-MG	User ID Password	User ID: `administrator` Password: `admin123`

Web service	CT-MG	User key
		Password
			User key:
			`1000`
			Password:
			`admin123`

Command line tool	CT-MG	User key
		Password
			User key:
			`1000`
			Password:
			`admin123`

It is recommended that you change the initial password in the CT-MG administration portal (**Change Password** page in the **Account** menu).

SAML_SP Authentication Mode

What is accessed?	Authentication by	Authentication via	Initial credentials
Administration portal	IdP SAML 2.0 assertion in CT-MG	User ID Password	User ID: The ID you specified in the `ADMIN_USER_ID` configuration key in the `sso.properties` configuration file. Password: Password as known in the IdP system for the operator user ID.

What is accessed?	Authentication by	Authentication via	Initial credentials
Web service	STS SAML 2.0 assertion in CT-MG	User ID Password	User ID: The ID you specified in the `ADMIN_USER_ID` configuration key in the `sso.properties` configuration file. Password: Password as known in the STS system for the operator user ID.
Command line tool	CT-MG	User key Password	User key: `1000` Password: `admin123`

If you are the first operator accessing CT-MG, you need to change your password in the CT-MG administration portal (**Change Password** page in the **Account** menu). Any additional operator as well as all other users can change their password in the external authentication system only.

3 Configuring CT-MG

This chapter describes:

- How to update configuration settings for CT-MG.
- How to add a currency definition to CT-MG.
- How to use and configure timers in CT-MG.
- How to define and manage system-wide LDAP connection settings.
- How to tune the CT-MG performance.

3.1 Updating Configuration Settings

The CT-MG 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 CT-MG configuration settings are provided in the following files in the directory to which you extracted the `fujitsu-bss-setup.zip` file (`<install_dir>/fujitsu_bss_setup`) when you installed CT-MG. In the following sections, this directory is referred to as `<setup_dir>`.

- `db.properties`: Settings for database setup and access.
- `glassfish.properties`: Configuration settings for the application server.
- `glassfishJMSBroker.properties`: Configuration settings for the Java Message Service (JMS) in the application server.
- `configsettings.properties`: Configuration settings for the CT-MG 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.

- `sso.properties`: Configuration setting for creating the first platform operator in CT-MG (`ADMIN_USER_ID`). Only required for the first installation when installing CT-MG in SAML_SP authentication mode. This setting cannot be updated.

For details on the individual configuration settings, refer to *Configuration Settings* on page 58.

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

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

To update the configuration settings for the application server, 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 CT-MG in the application server by executing the `build-glassfish-setup.xml` file in `<setup_dir>` as follows:

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

`<ANT_HOME>` is the installation directory of Apache Ant.

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

To update existing settings for the CT-MG services, choose one of the following options:

- CT-MG administration portal - 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:

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

  `<ANT_HOME>` is the installation directory of Apache Ant.

Note: If you execute the above command while CT-MG is running, you need to restart CT-MG for viewing the updated configuration settings in the administration portal, because the previous settings still exist in the cache.

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

```
<JAVA_HOME>/bin/java -jar lib/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.

Retrieving the Current Configuration Settings

You can at any time view the current configuration settings for the CT-MG services.

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

- CT-MG administration portal - 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_HOME>/bin/java -jar lib/fujitsu-adm-um-operatorsvc-client.jar
 <userkey> <password>
 getconfigurationsettings
```

For details on the individual configuration settings, refer to *Configuration Settings* on page 58.

3.2   Adding a Currency to CT-MG

After installation, you need to add the currency or currencies to be supported by CT-MG. 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 CT-MG when calculating subscription usage charges.

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

- CT-MG administration portal - 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_HOME>/bin/java -jar lib/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 specifying when it is executed periodically. The values are indicated in milliseconds.

Time intervals are defined and can be changed in the CT-MG configuration settings.

Note:	Be aware that several functions in CT-MG 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.

The initial expiration time of a timer for which a time interval is defined is calculated based on January 1st, 00:00:00.000, of the current year. For example, if you specify an interval of one week for a timer on January 5th, 14:30:00, the timer will expire for the first time on January 8th at 00:00:00.000, next on the 15th, the 22nd, etc. Or, if you specify an interval of one month for a timer on August 4th, 17:00:00, the timer will expire for the first time on September 1st at 00:00:00.000, next on October 1st, November 1st, etc. To avoid the expiration of several timers at the same time, which would result in heavy load on the system, there is an additional setting: An **offset** for each timer. The offset is added to the 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 check for unconfirmed accounts is executed: every 24 hours. The third setting, the offset, is set to 5 minutes. The timer will expire 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 administrators and the relevant subscription managers of the organizations who created the subscriptions, informing them 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 calculating subscription usage costs (customer billing data) or revenue share data. The interval for this timer is one day 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`

For a detailed description of the timers, refer to *CT-MG Configuration Settings* on page 62.

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

- CT-MG administration portal - 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_HOME>/bin/java -jar lib/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:

- CT-MG administration portal - 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_HOME>/bin/java -jar lib/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:

• CT-MG administration portal - 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_HOME>/bin/java -jar lib/fujitsu-adm-um-operatorsvc-client.jar
 <userkey> <password>
 reinittimers
```

3.4   Managing LDAP Settings

Note: This section is relevant only if CT-MG is installed in INTERNAL authentication mode. Refer to *Authentication* on page 17 for details on authentication modes.

User IDs and passwords of an organization can be created and maintained in the platform or in an existing LDAP system of an organization.

When maintained in the platform, the user data is stored in the platform's database. An organization's administrator can register new users, and, if required, request passwords to be reset by the operator.

When using an LDAP system, an organization does not need to register its users manually with the platform. The organization's administrator can import the users from the LDAP system, thus automatically registering them with the platform. The users are managed in the LDAP system. The platform continuously synchronizes its information on the users. Connection settings have to be defined in a configuration file so that the platform can connect to the LDAP system.

Whether an organization uses an LDAP system for user management is determined when the organization is created in the platform. An organization can be created in several ways:

• A customer registers himself. In this way, an organization with the customer role is created. Users are always managed in the platform, and an LDAP system cannot be used.

• A seller (supplier, reseller, or broker) registers a customer. In this way, an organization with the customer role is created. The seller can specify whether user management in an external LDAP system is to be used.

• You as the platform operator create an organization of any role. For any organization, you can specify whether user management in an external LDAP system is to be used. For details, refer to *Creating an Organization* on page 32.

A mixture of maintaining users in the platform and in the LDAP system is not supported. In addition, the type of user management can no longer be changed once an organization has been created.

The operator organization cannot use an external LDAP system. So you can always log in to the platform and change connection settings, for example, if the LDAP system of an organization is not available and thus authentication against it is not possible.

As an operator, you can define default LDAP configuration settings for the entire platform. These settings apply as long as no organization-specific LDAP settings are specified. You can also check an organization's connection to its LDAP system.

Defining Default LDAP Settings

Defining default LDAP settings for the entire platform is useful if settings are to be reused for several organizations. If organization-specific LDAP settings exist, they overrule the default LDAP settings.

To define default LDAP settings, choose the following option:

- CT-MG administration portal - Choose **Manage LDAP settings** in the **Operation** menu.

  For detailed step-by-step instructions, refer to the online help. For a list of LDAP keys, refer to *LDAP Keys* on page 74.

Checking the Connection to an LDAP System

If for some reason an organization's LDAP system cannot be reached, you can check and restore the connection so that user authentication is possible again.

To check an LDAP connection, choose the following option:

- CT-MG administration portal - Choose **Manage LDAP settings** in the **Operation** menu.

  For detailed step-by-step instructions, refer to the online help. For a list of LDAP keys, refer to *LDAP Keys* on page 74.

3.5   Tuning Performance

If you encounter a decrease in performance when operating CT-MG, this might be caused by an insufficient JDBC pool size setting for the CT-MG domain in your application server together with an insufficient setting for the maximum number of connections in your PostgreSQL server.

Check the application server log file. If you find exceptions like the following, proceed as described below.

```
--- ERROR org.hibernate.util.JDBCExceptionReporter
- Error in allocating a connection.
Cause: In-use connections equal max-pool-size and
 expired max-wait-time.
Cannot allocate more connections.
-- javax.persistence.PersistenceException: org.hibernate.exception.
GenericJDBCException: Cannot open connection
```

The `max-pool-size` setting in GlassFish determines the maximum number of connections that can be created by CT-MG to the database server. The `max-wait-time` is the time CT-MG waits for a response before sending a connection timeout (default: 60,000 milliseconds).

Proceed as follows:

1. In the application server's administration console, go to **Resources > JDBC > Connection Pools > BSSPool**.

2. In the **General** section, set the **Maximum Pool Size** to a value higher than the default value (`32`), for example, to `100`.

3. Stop and restart the CT-MG domain.

4. Stop your PostgreSQL server.

5. Edit the file

   `<postgres_dir>/data/postgresql.conf`

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

   Set the `max_connections` property value to a value higher than or equal to the size of all the JDBC connection pools on your application server. This setting determines the maximum number of concurrent connections to the PostgreSQL server. If you are running no applications other than CT-MG, the value is calculated as follows:

   Maximum pool size of the BSSPool + 92 (default for JMS connections in the CT-MG domain and the master indexer domain).

   Example: Set the pool size for the BSSPool in the application server to `100`. Then you need to set the maximum number of connections in your database server to at least `192`.

6. Restart your PostgreSQL server for the changes to take effect.

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 CT-MG, 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 audit log data.
- The backup of the databases, configuration settings, and log files.
- The monitoring of the JMS queues.

4.1 Monitoring CT-MG

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

The logging of CT-MG is based on the `log4j` tool.

4.1.1 Log Types and Location of Log Files

CT-MG 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>/glassfish/domains/bes-domain/logs`

In addition to the CT-MG-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

CT-MG supports the following types of log information with their corresponding log level:

- **ERROR**: Problems that do not allow to continue working with CT-MG 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 CT-MG 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` file contains the settings for the log level and a list of configuration settings that determine the behavior of the framework at runtime. For details on `log4j` settings, refer to the relevant Apache documentation

The log level defines which type of information is written to the log files. You can change the log level by adapting the file (see below). 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, CT-MG looks for this file in the `config` subdirectory of your application server domain:

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

When the `log4j.properties` file is specified for the first time, CT-MG 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; CT-MG need not be restarted.

If the `log4j.properties` file does not exist or the value for the `LOG_CONFIG_FILE` configuration setting 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 CT-MG for the new level to become effective.

The `log4j.properties` file may contain entries for defining different log levels for the different CT-MG 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.subscriptionservice
Tag service	com.fujitsu.adm.um.tagservice
Trigger service	com.fujitsu.adm.um.triggerservice
Trigger definition service	com.fujitsu.adm.um.triggerdefinitionservice
VAT service	com.fujitsu.adm.um.vatservice

To log every operation in CT-MG, you can use the following line in the `log4j.properties` file:

```
log4j.logger.sys.com.fujitsu.adm.um=DEBUG
```

4.1.3 Log Entry Format

Every entry in a CT-MG 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 CT-MG, 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 Audit Log Data

You can export information on all kinds of user operations related to subscriptions and marketable services including their price model. The data represents an audit log. Exporting this data may be useful, for example, to check when and by whom a price model was changed or when subscriptions were created.

You can export information on subscription-related operations performed by administrators or standard users, as well as information on service and price model-related operations performed by service managers of supplier organizations. Administrator operations include actions performed by subscription managers. Service manager operations include actions performed by resellers and brokers.

Note: The system records user operations only if the `AUDIT_LOG_ENABLED` configuration setting is set to `true`.

You can specify the start and end date of the time period for which you want to export the log data. The dates must be specified in the format `YYYY-MM-DD`.

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

- CT-MG administration portal - Choose **Export audit 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_HOME>/bin/java -jar lib/fujitsu-adm-um-operatorsvc-client.jar
 <userkey> <password>
getuseroperationlog filename=test.csv
 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 *Audit Log* on page 76.

4.3 Backup and Recovery

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

Backup

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

- CT-MG database (`bss`). The frequency of database backups depends on the amount of changes and on the availability of time slots with low load. PostgreSQL supports database backups without previous shutdown. For details, refer to the PostgreSQL documentation.
- Certificates contained in the `bes-domain` keystore and truststore (`keystore.jks` and `cacerts.jks` files).
- CT-MG log files located in the configured directory.
- Configuration files.
- Customizations made to marketplaces, especially style sheets and localized texts presented at the user interface.

Note: When preparing for an update installation of your current CT-MG release, always create a backup of all data mentioned above.

Recovery

If you need to recover your CT-MG installation, the recommended procedure is as follows:

1. Make sure that all JMS messages have been processed. They are stored in the `bssjms` database. For example, check the JMS broker as follows:

```
<GLASSFISH_HOME>/mq/bin/imqcmd.exe -b localhost:8076
   -u admin query bkr
```

where `8076` is the port where the JMS broker is running.

2. Restore the `bss` database from the backup using the relevant PostgreSQL commands.
3. Stop the CT-MG domain and the master indexer domain.

4. Delete the shared directory containing the search index data. The location of the directory is specified by the `master.slave.shared.hibernate.search.sourceBase` setting in the `glassfish.properties` file.

5. On the master indexer domain, delete the master search index base directory:

   `<GLASSFISH_HOME>/glassfish/domains/master-indexer-domain/masterIndexBase`

6. On the CT-MG domain, delete the slave index base directory:

   `<GLASSFISH_HOME>/glassfish/domains/bes-domain/slaveIndexBase`

7. Start the master indexer domain.

8. Start the CT-MG domain.

   The search index is rebuilt automatically.

4.4 Checking the Load on the JMS Queues

CT-MG uses JMS queues for asynchronous processing of requests of different types:

- The **trigger queue** handles calls to the notification service of an external process control system.
- The **task queue** handles requests from the Java Mail session for sending notification emails to CT-MG users.
- The **indexer queue** handles requests for synchronizing search index data on the slave nodes with master index data.
- The **master indexer queue** handles requests for replicating the master search index data in the shared directories of the slave nodes.

The requests are stored in the relevant JMS queues before they are actually executed.

It is recommended to check the load on the JMS queues 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 two domains, `bes-domain` and `master-indexer-domain`, in order to speed up the JMS recovery.

5 Managing Organizations and Users

This chapter describes how to:

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

5.1 Creating an Organization

You can create an organization and specify its roles (technology provider, supplier, reseller, broker, or - implicitly - customer). For every new organization, you must define a user who is to become its first administrator. This user can later register new users and assign roles to them.

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

- CT-MG administration portal - 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. Specify the following entries:

     - `user.address`: Address of the organization's administrator.
     - `user.additionalname`: Any additional name the administrator might have.
     - `user.email`: Email address of the 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 administrator. User IDs are restricted to 100 characters and must not contain any of the following characters:

       `! " # $ % & ' * + , / : ; < = > ? \ ^ \``

> **Note:** If CT-MG is installed in INTERNAL authentication mode and you create an organization with LDAP-based user management, the user ID must match the ID in the external LDAP system. The organization's LDAP configuration determines which additional settings are required. All settings defined in the configuration must be mapped to corresponding values, for example, first name, last name, email address, and language.
>
> If CT-MG is installed in SAML_SP authentication mode, the user ID must match the ID in the IdP system. Refer to *Authentication* on page 17 for details on authentication modes.

  - `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 displaying details on the 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`, `SUPPLIER`, `RESELLER`, `BROKER`. If the organization wants to act, for example, both as a supplier and a technology provider, separate the values by a comma without any blanks. The `RESELLER` and `BROKER` roles cannot be combined with each other or with the `TECHNOLOGY_PROVIDER` or `SUPPLIER` role. If you do not specify a role, an organization with the `CUSTOMER` role will be created. In this case, you must specify the marketplace to which the customer organization will be directed (`organization.marketplaceid` parameter). For a `SUPPLIER` organization, you must specify the default operator revenue share (`organization.operatorrevshare` parameter).
- `organization.marketplaceid`: ID of the marketplace to which the customer is to be directed.
- `organization.operatorrevshare`: Revenue share agreed between the operator and the supplier organization, which the operator is to receive from the supplier for selling services on the marketplaces managed on the operator's platform. The default operator revenue share applies to all services offered by the supplier as long as you do not define a service-specific revenue share. It also applies irrespective of whether the supplier sells his services himself, or uses brokers or resellers for selling them. For details, refer to *Defining Revenue Shares* on page 38.
- `organization.settings`: Applicable only if CT-MG is installed in INTERNAL authentication mode.

  Fully qualified path and name of an LDAP configuration file containing the parameters required for connecting to the organization's LDAP system. If no configuration file is specified, the organization does not use LDAP-based user management.

  Here is a sample configuration file:

```
LDAP_BASE_DN=ou\=people,dc\=est,dc\=mycompany,dc\=de
LDAP_CONTEXT_FACTORY=com.sun.jndi.ldap.LdapCtxFactory
LDAP_URL=ldap://myldapserver.lan.est.company.de:389
LDAP_ATTR_UID=uid
LDAP_ATTR_LAST_NAME=sn
LDAP_ATTR_FIRST_NAME=givenName
LDAP_ATTR_EMAIL=scalixEmailAddress
```

  For a list of LDAP keys, refer to *LDAP Keys* on page 74

2. Execute the `createorganization` command with the following syntax:

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

5.2 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:

- CT-MG administration portal - 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 command below. 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.

     The following entries cannot be changed:

     - `organization.roles`
     - `organization.marketplaceid`
     - `organization.settings`

  3. Use the `updateorganization` command with the following syntax:

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

5.3 Adding a Role to an Organization

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

To do so, choose one of the following options:

- CT-MG administration portal - 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_HOME>/bin/java -jar lib/fujitsu-adm-um-operatorsvc-client.jar
 <userkey> <password>
 addorganizationtorole orgid=<organizationID>
 role={SUPPLIER | TECHNOLOGY_PROVIDER | RESELLER | BROKER}
```

If the organization wants to act, for example, both as a supplier and a technology provider, separate the values by a comma without any blanks. The RESELLER and BROKER roles cannot be combined with each other or with the TECHNOLOGY_PROVIDER or SUPPLIER role.

5.4 Managing User Accounts

Note: This section is relevant only if CT-MG is installed in INTERNAL authentication mode. Refer to *Authentication* on page 17 for details on authentication modes.

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.

Note: If CT-MG is installed in INTERNAL authentication mode and an organization uses an external LDAP system for user management, only those fields that are not mapped to settings in the LDAP system can be edited. For example, passwords can only be changed in the LDAP system.

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

- CT-MG administration portal - 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_HOME>/bin/
java -jar lib/fujitsu-adm-um-operatorsvc-client.jar
 <userkey> <password>
 lockuser userid=<userID>
```

  - unlockuser for unlocking a user account:

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

  - resetpasswordforuser for resetting the password of a user:

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

6   Reporting

CT-MG 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**: Shows the accumulated revenues of all suppliers and resellers registered with the platform. A time frame can be specified. Dates are to be entered in the format `YYYY-MM-DD`.

  The report shows the name, ID, and revenue of each supplier and reseller, and one accumulated value per currency. A supplier's revenue includes all revenues generated by his authorized brokers.

- **External services report**: Shows all sellers (suppliers, brokers, resellers) who have published services with the external access type to a marketplace on the platform.

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

- **Supplier revenue share report**: Shows the revenue share data for all suppliers registered with the platform for a specified month. For each marketplace, the revenue share data is calculated from the accumulated charges for subscriptions which were due during the past calendar month, irrespective of supplier-specific billing periods. For each supplier, the data is broken down to the organizations which generated the revenue as well as to the individual services. The suppliers' liabilities to the other participating parties are calculated. The operator can specify the month for which to generate the report. The month is to be entered in the format `MM`, the year in the format `YYYY`.

- **Broker/reseller revenue share report**: Shows the revenue share data for all brokers and resellers registered with the platform. For each marketplace, the revenue share data is calculated from the accumulated charges for subscriptions which were due during the past calendar month. For each broker and reseller, the data is broken down to the suppliers who are providing the services to the brokers and resellers. The operator can specify the month for which to generate the report. The month is to be entered in the format `MM`, the year in the format `YYYY`.

- **Service report (of a supplier)**: Shows all marketable services of a supplier with their existing subscriptions. Subscriptions to services offered by brokers are also listed. The operator selects the supplier by specifying the supplier organization ID.

- **Customer report (of a supplier)**: Shows all customers of a supplier and his brokers with their organization ID and the services they have subscribed to. The report outputs whether a subscription is still active or has already ended. The operator selects the supplier by specifying the supplier organization ID.

- **Billing report (of a supplier)**: Shows a summary of all billing data for each customer of a supplier and his brokers. The operator selects the supplier by specifying the supplier organization ID. The billing data includes the billing data key of each subscription.

- **Detailed billing report for an existing invoice of a supplier's customer**: Shows the billing data of the current billing period for a selected subscription of a supplier's customer. With pro rata cost calculation, the costs for the real service usage are calculated based on milliseconds. With per time unit calculation, the report contains the subscription costs for the time units that ended in the billing period. The operator selects the subscription by specifying a billing data key. The billing data key of each subscription is output by the **Billing report (of a supplier)**.

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

- CT-MG administration portal - Choose **Create report** in the **Account** menu, and select the supplier revenue report.

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

  The generated report is instantly displayed at the CT-MG administration portal. 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 `getrevenuelist` command with the following syntax:

```
<JAVA_HOME>/bin/java -jar lib/fujitsu-adm-um-operatorsvc-client.jar
 <userkey> <password>
 getrevenuelist month=<YYYY-MM>
```

  The command returns a CSV list in the following format:

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

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

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

  `CURRENCY`: ISO currency code.

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

To create one of the other reports, choose the following option:

- CT-MG administration portal - 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 in the CT-MG administration portal. 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 *CT-MG Configuration Settings* on page 62.

---

7 Managing Billing and Payment

At daily intervals, the database content is checked for organizations which have produced billing-relevant data, the billing data is calculated, and the data is collected and stored in the database. These checks and calculations are called "billing runs for customer billing data".

Each supplier and reseller can define his preferred start day of the monthly billing periods. When a billing period ends, the costs for all customer subscriptions that were produced in the course of this period are calculated. This is done in the billing run that is executed on the start day of the next billing period plus the offset defined by the `TIMER_INTERVAL_BILLING_OFFSET` configuration setting. For example: A supplier defines that his billing period is to start on the 8th of a month. The `TIMER_INTERVAL_BILLING_OFFSET` is set to 5 days and 4 hours. The billing run calculating the costs for the supplier's customers is executed on the 13th of each month at 04:00:00.000.

Brokers and resellers as well as the platform operator and the owners of the marketplaces on which services are published, usually receive a share of the revenue for the services. The revenue shares for these organizations and their effects on suppliers' revenues are calculated every month based on the results of the daily checks for customer billing data. This calculation is called "billing run for revenue share data". The billing period for this billing run cannot be changed; it is always executed for the previous month on the first day of a month plus the offset defined by the `TIMER_INTERVAL_BILLING_OFFSET` configuration setting.

When a billing run for customer billing data is finished, the billing data that must be handled by a payment service provider (PSP) is processed. This transaction is called "payment processing".

You should check at regular intervals whether billing runs or payment processing fail, and, if yes, explicitly execute them.

This chapter describes how to:

- Define revenue shares.
- Integrate the services of a PSP with CT-MG.
- Assign payment types to suppliers and resellers.
- Handle billing data, including how to manually start a billing run or payment processing and retry failed payment processes, as well as how to preview or export billing data to an XML file.

7.1 Defining Revenue Shares

Suppliers may involve brokers and resellers in selling their services. The brokers and resellers as well as the platform operator and the owners of the marketplaces on which the services are published, usually receive a share of the revenue for the services. CT-MG calculates these revenue shares based on the billing data for the customers who use the services. The operator revenue share applies irrespective of whether services are sold by the supplier or by his resellers and brokers.

Suppliers, brokers, resellers, and marketplace owners can generate reports for their revenue shares and export the revenue share data for a specific month. As an operator, you can export the data for all the suppliers, brokers, resellers, or marketplace owners known to your platform installation. The exported data can be forwarded, for example, to an accounting system which continues to process it. For details, refer to *Handling Billing Data* on page 43.

As an operator, you are responsible for defining the revenue shares. You can define the following:

- Operator revenue share to be paid by each supplier for using the platform.
- Marketplace owner revenue share to be paid by a supplier for publishing services on a specific marketplace.

- Revenue shares for broker and reseller organizations to be paid by a supplier for selling his services.

The revenue shares specify the percentage of the revenue the operator, marketplace owners, brokers, or resellers are entitled to.

The values you enter for revenue share percentages are based on agreed conditions between the operator, marketplace owners, suppliers, resellers, and brokers. You as the operator are responsible for setting correct values. It is possible to set the percentages to a total of over 100%. This might be intentional. For example, a supplier who wants to strongly promote a service for a limited period of time may grant a broker or reseller a revenue share of 80%. In addition, the supplier may need to pay 30% of the revenue to the marketplace owner and 10% to the operator. This results in a total revenue share percentage of 120 and thus in a negative revenue for the supplier.

Revenue shares can be defined on the following levels:

1. **Operator revenue share**. It is defined for every supplier organization that wants to sell its services on your platform. The default operator revenue share is specified when you create a supplier organization. It is independent of the marketplace to which a service is published, and whether the supplier sells services himself or authorizes resellers and brokers to do so. The default operator revenue share applies to all services of a supplier as long as you do not define service-specific operator revenue shares. A service-specific operator revenue share applies to a specific marketable service and overrules the default operator revenue share defined for the respective supplier organization.

2. **Marketplace-specific revenue shares**. They are defined for one marketplace and comprise a percentage for the marketplace owner as well as a default for all broker and reseller organizations.

3. **Service-specific revenue shares**. For every marketable service a supplier offers to brokers or resellers for publishing, a service-specific revenue share can be defined. This revenue share applies irrespective of the marketplace the service is published to. It overrules any defined marketplace-specific revenue shares.

4. **Individual broker or reseller revenue shares**. An individual revenue share can be defined for each broker and reseller organization. This revenue share definition overrules the service-specific and marketplace-specific revenue share definitions.

To define operator revenue shares, proceed as follows:

- Default operator revenue share for a supplier organization:

  CT-MG administration portal - Choose **Create organization** in the **Operation** menu for a new supplier organization or **Manage organization** in the **Operation** menu for an existing supplier organization.

- To define a service-specific operator revenue share:

  CT-MG administration portal - Choose **Manage operator revenue share** in the **Operation** menu.

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

To define marketplace-specific revenue shares, choose the following option:

- CT-MG administration portal - Choose **Update marketplace** in the **Marketplace** menu.

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

To define revenue shares for specific services, brokers, or resellers, choose the following option:

- CT-MG administration portal - Choose **Manage broker revenue share** or **Manage reseller revenue share** in the **Marketplace** menu, respectively.

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

7.2   Integrating PSP Support

For payment collection, a supplier or reseller organization can use the facilities of a commercial payment service provider (PSP). CT-MG provides a Web service interface (payment service) to enable operator organizations to integrate payment service providers (PSPs) with CT-MG. The following figure illustrates the communication paths between the components involved in PSP integration:
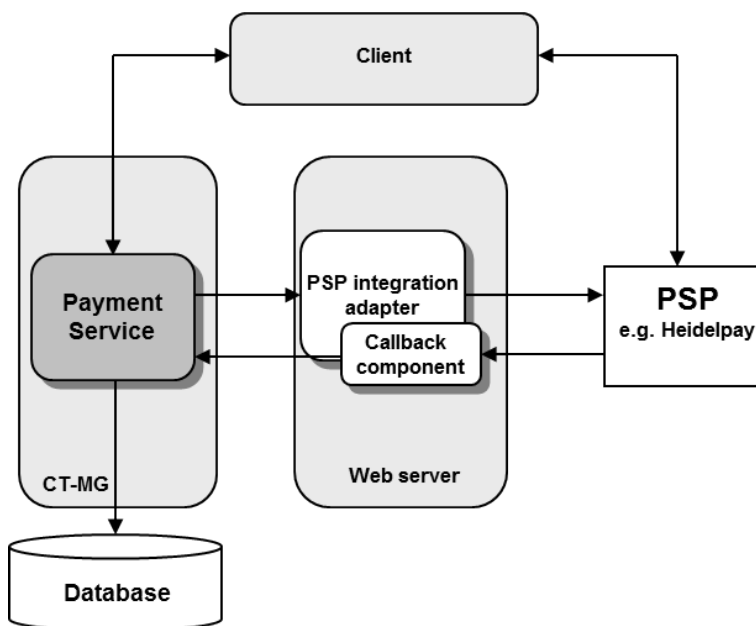
As a sample PSP integration, Heidelpay is supported.

The process of integrating PSP support for a supplier or reseller organization and the tasks to be performed by an operator are as follows:
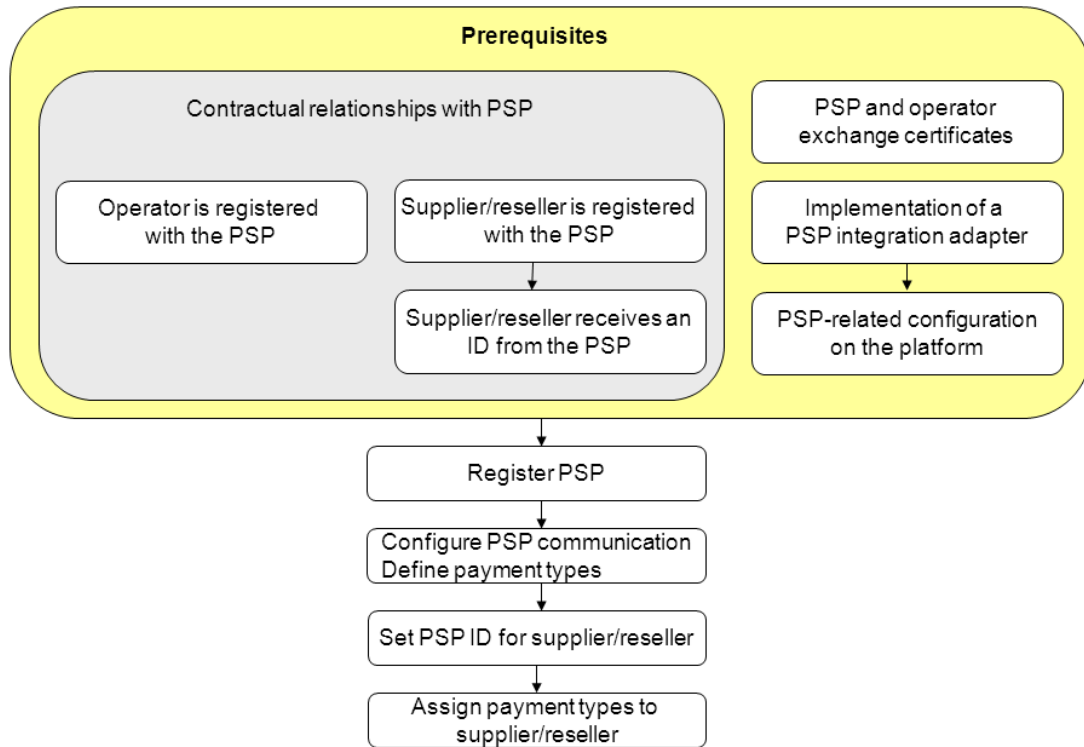
The subsequent sections describe the prerequisites and the operator tasks in detail.

7.2.1 Prerequisites

Before a supplier or reseller 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 or resellers who are to use the services of the PSP need to have a contractual relationship with the PSP. The supplier or reseller organizations must also be registered in the PSP system.

3. The PSP issues an identifier for the supplier or reseller so that payments can be immediately linked to the supplier's or reseller's account.

4. Your own organization and the PSP organization must have exchanged certificates so that trusted and secure communication between CT-MG and the PSP system is ensured.

   For details, refer to *Integrating Certificates for Trusted Communication* on page 46.

5. A PSP integration adapter must have been implemented as a Web service and deployed on an application server. This adapter is a Web service providing the methods for integrating the PSP with CT-MG. You need the URL pointing to the WSDL file of this service.

   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 CT-MG:

   `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 20.

7.2.2 Registering a Payment Service Provider With CT-MG

Before an organization can use the services of a commercial PSP, you need to register the PSP with CT-MG.

To register a PSP, you have the following option:

• CT-MG administration portal - 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 CT-MG and is visible to the operator only.

• WSDL URL: Link to the WSDL file containing the PSP integration adapter.

• 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 CT-MG 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 46.

7.2.3 Configuring PSP Communication and Payment Types

Before an organization 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:

• CT-MG administration portal - Choose **Manage payment service provider** in the **Operation** menu.

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

7.2.4 Setting the PSP Identifier for a Supplier or Reseller

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

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

To set the PSP identifier for an organization, choose one of the following options:

• CT-MG administration portal - 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 `setpspaccountfororganization` command with the following syntax:

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

7.3 Assigning Payment Types to Suppliers and Resellers

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

If a supplier or reseller 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 or reseller, choose one of the following options:

- CT-MG administration portal - 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_HOME>/bin/java -jar lib/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 42.

7.4 Handling Billing Data

This section describes how you can check whether billing runs have failed, re-invoke payment processes, or explicitly start a billing run. In addition, you can preview customer billing data for a specific period or export revenue share data to an XML file.

Use the CT-MG reporting facilities to retrieve detailed information on all billing-relevant data of the suppliers, resellers, brokers, and marketplace owners managed on your platform. For details, refer to *Reporting* on page 36.

The definition of revenue shares is described in detail in *Defining Revenue Shares* on page 38.

7.4.1 Start Billing Run

You can explicitly start a billing run for calculating the customer billing data for a billing period. The billing period for which the data is generated depends on the day when you start the billing run and on the `TIMER_INTERVAL_BILLING_OFFSET` configuration setting. Explicitly starting a billing run 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 calculation of the data.

If you start a billing run on the first day of a month plus the day(s) defined in the `TIMER_INTERVAL_BILLING_OFFSET` setting, the revenue share data is also calculated and stored.

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

- CT-MG administration portal - 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_HOME>/bin/java -jar lib/fujitsu-adm-um-operatorsvc-client.jar
 <userkey> <password>
 startbillingrun
```

7.4.2 Start Payment Processing

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

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

- CT-MG administration portal - 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_HOME>/bin/java -jar lib/fujitsu-adm-um-operatorsvc-client.jar
 <userkey> <password>
 startpaymentprocessing
```

7.4.3 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:

- CT-MG administration portal - 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_HOME>/bin/java -jar lib/fujitsu-adm-um-operatorsvc-client.jar
 <userkey> <password>
 retryfailedpaymentprocesses
```

7.4.4 Preview Billing Data

You can preview the billing data for a customer's subscriptions for a specified 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.

The billing data preview collects the billing-relevant data for the specified customer and accumulates it for every day within the specified time frame. The data is not stored in the database; the result is just a cost projection for the customer organization.

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

- CT-MG administration portal - Choose **Billing data preview** 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_HOME>/bin/java -jar lib/fujitsu-adm-um-operatorsvc-client.jar
 <userkey> <password>
 getbillingdata orgid=<organizationID> from=<startdate> to=<enddate>
```

The billing data is saved to an XML file (`<date>BillingData.xml`).

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

7.4.5 Export Revenue Share Data

You can export the revenue share data for all organizations with a specific role for a specific time frame. Based on the customer billing data calculated for the given time frame, the costs are analyzed to determine the revenue shares for the operator, marketplace owners, brokers, and resellers and their effects on the suppliers' revenues.

You can use the data to get an overview of who is to receive which revenue shares. The exported data can be forwarded, for example, to an accounting system which continues to process it. For example, you can invoice your revenue share to marketplace owners or suppliers. For details on defining revenue shares, refer to *Defining Revenue Shares* on page 38.

To export billing data, choose the following option:

- CT-MG administration portal - Choose **Export billing data** in the **Account** menu.

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

The billing data can be saved to an XML file (`<date>BillingData.xml`) and opened in an editor of your choice. You can edit the data and work with it as required.

For a detailed description of the elements and attributes that may occur in the XML file, refer to the *Developer's Guide*.

8 Integrating Certificates for Trusted Communication

Certificates are required for CT-MG to allow for trusted communication between CT-MG and an application underlying a technical service, a payment service provider (PSP), an Identity Provider (IdP) or a Security Token Service (STS).

The following organizations are involved when using certificates:

• Platform operator

• Technology providers integrating their applications with CT-MG.

• PSPs whose services are to be integrated with CT-MG for invoicing and payment collection.

• Organizations integrating an external process control system.

• Any other organization using Web service calls to or from CT-MG

8.1 Introduction

Web service calls coming from CT-MG (e.g. for provisioning application instances for services, or for integrating a process control system) or sent to it (e.g. by an application such as a PSP system) 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, CT-MG can act as a server (Web service calls to CT-MG) or as a client (Web service calls from CT-MG).

Every application integrated with CT-MG or accessing the platform services acts as a Web service client. CT-MG acts as the server, and the client must provide its authenticating data to the server. Three options are available:

• **Basic authentication**:

  The caller sends the key and password of a CT-MG user and addresses the CT-MG Web services with the `BASIC` suffix. This mechanism does not apply certificates for user authentication. However, certificates are involved because the communication between the client and CT-MG must be secured using TLS/SSL and HTTPS. You should also configure your network's firewall to block JNDI lookups from the outside when using basic authentication.

  Basic authentication must be used for calling CT-MG functions that require the specification of a user role. The user role determines whether the user is allowed to execute the function.

  Basic authentication can only be used if CT-MG is installed in INTERNAL authentication mode.

• **Certificate-based authentication**:

  The caller provides a certificate to the CT-MG server. It addresses the CT-MG Web services with the `CLIENTCERT` suffix. In this case, the following requirements must be fulfilled:

  • The distinguished name (DN) of the client's certificate must correspond to the distinguished name (DN) configured and stored in CT-MG for the corresponding organization.

  • CT-MG must trust the client's certificate: The CT-MG truststore must contain a certificate with a valid signing chain to the certificate presented by the client.

  Certificate-based authentication must be used when integrating a PSP such as Heidelpay. Web services calling PSP-related functions of CT-MG do not require the specification of a user role.

Certificate-based authentication can be used in both authentication modes (INTERNAL and SAML_SP). Refer to *Integrating Certificates for Trusted Communication* on page 46 for details.

- **Security token service authentication**:

  The caller addresses the CT-MG Web services with the `STS` suffix.

  Web service calls are authenticated by means of a security token service (STS). This is a Web service that issues security tokens as defined in the WS-Security/WS-Trust specification. WS-Security (Web Services Security) is an extension to SOAP to apply security to Web services.

  This mechanism does not apply certificates for user authentication. However, certificates are involved because the communication between the client and CT-MG must be secured using TLS/SSL and HTTPS.

  Security token service authentication can only be used if CT-MG is installed in SAML_SP authentication mode.

Note: Both, the server and the client certificates must be created and signed using the same JRE/JDK. Otherwise, the communication may fail.

Terminology

CT-MG 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 proofing 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 CT-MG

For the provisioning of application instances and for integrating external process control or PSP systems, CT-MG calls other Web services which can be addressed by HTTPS. In this scenario, CT-MG 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 (common name) must correspond to the server's domain name and it must be valid at the time of calling.

- The client (CT-MG) must trust the server's certificate. To this end, the server's certificate must have been imported into the client's truststore, or the client's truststore must contain a root certificate with a valid signing chain to the certificate presented by the server.

8.3 Requirements for Web Service Calls to CT-MG

CT-MG provides Web services (platform services) that can be called by other systems, such as an external process control or a PSP system. In this scenario, CT-MG is the HTTPS server while the other system is a Web service client. The following requirements must be met to establish a connection to CT-MG:

- The CT-MG server must present a valid certificate: The CN (common name) must correspond to the server's domain name and it must be valid at the time of calling.
- The clients must trust the CT-MG server certificate for SSL connections: The clients' truststore must contain the CT-MG server certificate.
- In case of certificate-based authentication, CT-MG must trust the client's certificate: The CT-MG truststore must contain a certificate with a valid signing chain to the certificate presented by the client.

8.4 Certificate Integration Procedures

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

The CT-MG operator creates the root certificate. This certificate is imported to the CT-MG truststore as well as handed out to all technology providers or other organizations calling CT-MG Web services.

With this type of implementation, new applications can be added as technical services without touching the CT-MG 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:

□ **Operator task**

▨ **Task of external organization**
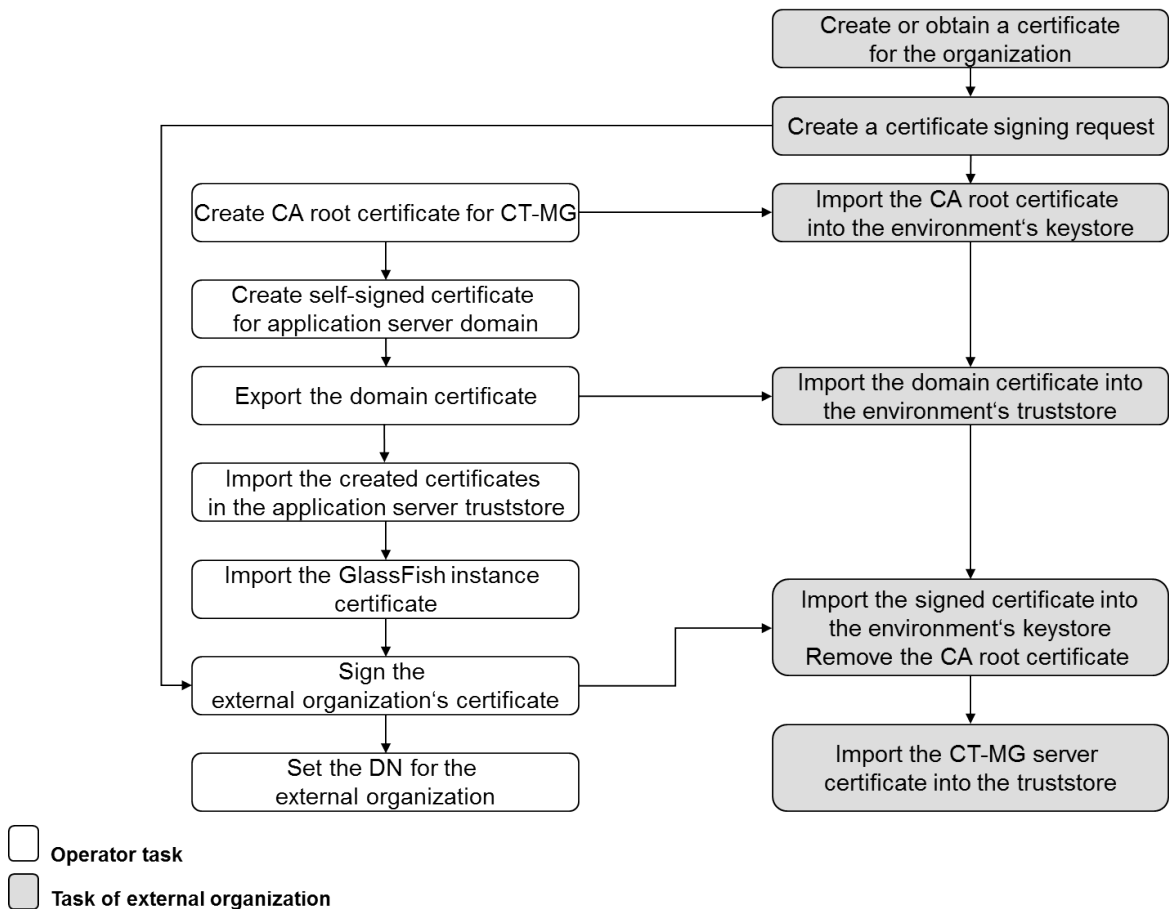
The operator tasks are described in detail in the subsequent sections. The tasks of an external organization that wants to make use of certificates are described in the *Developer's Guide*.

Prerequisites

- You have a Java environment installed of the Java version supported by CT-MG. 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.
- You have moved the keystore and truststore files in the application server domain where CT-MG has been deployed to a temporary directory. Check for the files

  `<GLASSFISH_HOME>/glassfish/domains/bes-domain/config/cacerts.jks` and

  `<GLASSFISH_HOME>/glassfish/domains/bes-domain/config/keystore.jks`.

  If they are present, move them to a temporary directory.

8.4.1 Creating a CA Root Certificate for CT-MG

The creation of the CA (Certification Authority) root certificate for CT-MG 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 CT-MG has been deployed, execute the following command of the JRE you are using:

```
<YourJRE>/bin/keytool -genkey -alias s1as -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 certificates.

Execute the following command to export the certificate:

```
<YourJRE>/bin/keytool -export -file bssDomain.crt
 -alias s1as -keystore
 <GLASSFISH_HOME>/glassfish/domains/bes-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 s1as -keystore
 <GLASSFISH_HOME>/glassfish/domains/bes-domain/config/cacerts.jks
```

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

8.4.5 Importing the GlassFish Instance Certificate

You need to import the Glassfish instance certificate with the `glassfish-instance` alias into the keystore and truststore of your application server. This certificate is available from the directory to which you moved the keystore and truststore files before starting with the certificate integration procedure.

Execute the following commands:

```
<YourJRE>/bin/keytool -importkeystore -srckeystore
    <TEMP_DIR>/keystore.jks -srcalias glassfish-instance
    -destkeystore keystore.jks -destalias glassfish-instance
```

```
<YourJRE>/bin/keytool -importkeystore -srckeystore
    <TEMP_DIR>/cacerts.jks -srcalias glassfish-instance
    -destkeystore cacerts.jks -destalias glassfish-instance
```

After having imported the certificates, restart the application server and delete the keystore and truststore files from your temporary directory.

8.4.6 Signing an Organization's Certificate

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

To sign a certificate, use `OpenSSL`. Make sure that you have access to the private key of the CT-MG 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 organization requesting certificate-based communication.

8.4.7 Setting the Distinguished Name for an Organization

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

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 CT-MG. For details, refer to *Registering a Payment Service Provider With CT-MG* on page 42.

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

- CT-MG administration portal - 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_HOME>/bin/java -jar lib/fujitsu-adm-um-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

This chapter describes

- How to create marketplaces.
- How to change the owner of a marketplace.
- How to delete a marketplace.
- How to configure marketplaces.

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

Use the CT-MG reporting facilities, for example, to retrieve information on the services published to your marketplace. For details, see chapter *Reporting* on page 36.

9.1 Creating a Marketplace

You are responsible for creating marketplaces for the organizations that want to authorize suppliers, brokers, and resellers to publish their services using the facilities of CT-MG. 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.

To create a marketplace, you have the following option:

- CT-MG administration portal - Choose **Create marketplace** in the **Marketplace** menu.

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

9.2 Changing the Owner of a Marketplace

The owner of a marketplace is responsible for administrating and customizing the marketplace to which suppliers, brokers, and resellers 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:

- CT-MG administration portal - Choose **Update marketplace** in the **Marketplace** menu.

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

9.3 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:

- CT-MG administration portal - Choose **Delete marketplace** in the **Marketplace** menu.

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

9.4 Configuring Marketplaces

CT-MG provides several configuration settings that influence the behavior of marketplaces. For example:

* `CUSTOMER_SELF_REGISTRATION_ENABLED`: Specifies whether customer organizations can register on a marketplace.

* `MP_ERROR_REDIRECT_HTTP` or `MP_ERROR_REDIRECT_HTTPS` : URL specifying a Web page that is to be displayed in case a visitor tries to access a marketplace without a valid marketplace ID.

* `TAGGING_MAX_TAGS`: The maximum number of tags composing the tag cloud.

* `TAGGING_MIN_SCORE`: The minimum number of times a tag must be used in services to be shown in the tag cloud.

For details on the settings, refer to *CT-MG Configuration Settings* on page 62.

For details on how to change configuration settings, refer to *Updating Configuration Settings* on page 20.

Appendix A: Application Server Resources

The following sections provide an overview of the resources which must be provided to CT-MG 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 CT-MG.

A.1 CT-MG Domain

The following sections give an overview of the resources which must be provided for the CT-MG domain (`bes-domain`) on the application server.

Data Source

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

Name	BSSDS
Class name	org.postgresql.xa.PGXADataSource
Resource type	javax.sql.XADataSource
Database schema	As distributed with CT-MG.

JMS Queues

The following JMS queues and corresponding connection factories are required for asynchronous processing:

1. **Trigger queue:**

    Queue name: `jms/bss/triggerQueue`

    Connection factory name: `jms/bss/triggerQueueFactory`

2. **Task queue:**

    Queue name: `jms/bss/taskQueue`

    Connection factory name: `jms/bss/taskQueueFactory`

3. **Indexer queue:**

    Queue name: `jms/bss/indexerQueue`

    Connection factory name: `jms/bss/indexerQueueFactory`

4. **Master indexer queue:**

    Queue name: `jms/bss/masterIndexerQueue`

    Connection factory name: `jms/bss/masterIndexerQueueFactory`

All queues and connection factories mentioned above use the following settings:

Initial and minimum pool size	1 connection
Maximum pool size	250 connections
Pool resize quantity	2 connections

Idle timeout	`600` seconds
Max wait time	`60000` milliseconds
On any failure	`Close all connections`
Transaction support	`XATransaction`
Connection validation	`Required`
UserName	`jmsuser`
Password	`jmsuser`

The master indexer queue / connection factory requires the following additional setting:

AddressList	`<master-indexer-host-name>:<port>`, where `port` is the port number of the search indexer application domain. Default: `8476`

Java Mail Session

A Java Mail Session is required for the application server to automatically send emails in case of specific user actions.

JNDI Name	`mail/BSSMail`
Store Protocol	`imap`
Transport Protocol	`smtp`
mail-smtp-auth	`false`
mail-user	`saas`
mail-smtp-port	`25`
mail-smtp-password	`password`

The settings are retrieved from the `glassfish.properties` file when running the installation scripts. The **JNDI Name**, **Store Protocol**, and **Transport Protocol** are mandatory and must not be changed.

Realm

CT-MG 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	`bss-realm`
JAAS context	`bssRealm`
Implementation class	`com.fujitsu.adm.um.security.ADMRealm`

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`.

A.2 Master Indexer Domain

The following sections provide an overview of the resources which must be provided for the `master-indexer-domain` on the application server.

Data Source

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

Name	BSSDS
Class name	org.postgresql.xa.PGXADataSource
Resource type	javax.sql.XADataSource
Database schema	As distributed with CT-MG.

JMS Queue

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

Queue name	jms/bss/masterIndexerQueue
Connection factory name	jms/bss/masterIndexerQueueFactory
Initial and minimum pool size	1 connection
Maximum pool size	250 connections
Pool resize quantity	2 connections
Idle timeout	600 seconds
Max wait time	60000 milliseconds
On any failure	Close all connections
Transaction support	XATransaction
Connection validation	Required
UserName	jmsuser
Password	jmsuser

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 CT-MG are provided in the following files in the directory to which you extracted the `fujitsu-bss-setup.zip` file (`<install_dir>/fujitsu-bss-setup`):

- `db.properties`
- `glassfish.properties`
- `glassfishJMSBroker.properties`
- `configsettings.properties`
- `sso.properties`

This appendix describes the settings in detail.

B.1 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`

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 domains for CT-MG.

Below you find a detailed description of the settings.

GLASSFISH_HOME

The absolute path and name of the GlassFish installation directory.

JDBC_DRIVER_JAR_NAME

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

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

MAIL_HOST

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

MAIL_RESPONSE_ADDRESS

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

Example: `saas@yourcompany.com`

MAIL_PORT

The port of your mail server.

Default: `25`

MAIL_USE_AUTHENTICATION

Optional. Defines whether mails can 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.

MAIL_TIMEOUT

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

Allowed values: Any value between `0` and `4924967296`

Default: `30000`

MAIL_CONNECTIONTIMEOUT

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

Allowed values: Any value between `0` and `4924967296`

Default: `30000`

glassfish.master.domain.host

Mandatory. The host name or IP address of the server where the domain for the search indexer application is deployed.

Example: `MyHibernateMasterIndexerHostName`

glassfish.master.domain.portbase

Mandatory. The base number for all ports used by the domain of the search indexer application. Make sure that the port base setting differs from the `glassfish.bes.domain.portbase` setting by at least 200.

Example: `8400`

glassfish.master.domain.jms.port

Mandatory. The port of the JMS broker used by the domain of the search indexer application.

Example: `8476`

glassfish.master.domain.port

Mandatory. The administration port of the search indexer domain.

Example: `8448`

glassfish.master.domain.name

Mandatory. The name of the domain for the search indexer application.

Default: `master-indexer-domain`

glassfish.master.domain.admin.user.name

Mandatory. The user name of the search indexer application domain administrator.

Default: `admin`

glassfish.master.domain.admin.user.pwd

Mandatory. The password of the search indexer application domain administrator.

Default: `adminadmin`

glassfish.master.domain.admin.master.pwd

Mandatory. The master password of the search indexer application domain administrator.

Default: `changeit`

master.slave.shared.hibernate.search.sourceBase

Mandatory. The shared directory hosting the search index for the master indexer node and the slave nodes, if any. From this directory, the master search index is replicated to the slave nodes, if any.

On Windows, double-escape the colon in the drive specification of the path to the shared directory.

Example: `C\\:/glassfish/masterSourceBase`

master.hibernate.search.default.refresh

Mandatory. The interval in seconds between refresh operations. At this interval, the search index is replicated and copied to the shared directory hosting the search index for the master indexer node.

Default: `300`

glassfish.bes.domain.portbase

Mandatory. The base number for all ports used by the domain of the CT-MG application. Make sure that the port base setting differs from the `glassfish.master.domain.portbase` setting by at least 200.

Example: `8000`

glassfish.bes.admin.domain.port

The administration port of the domain used for CT-MG.

Example: `8048`

glassfish.bes.domain.name

The name of the domain where you deployed CT-MG.

Example: `bes-domain`

glassfish.bes.domain.admin.user

The user name of the CT-MG domain administrator.

Default: `admin`

glassfish.bes.domain.admin.pwd

The password of the CT-MG domain administrator.

Default: `adminadmin`

glassfish.bes.domain.admin.master.pwd

The master password of the CT-MG domain administrator.

Default: `changeit`

glassfish.WS_PORT

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

Example: `8082`

glassfish.HIBERNATE_SLAVE_REFRESH_SEC

Mandatory. The interval in seconds between index data refresh operations. At this interval, the search index is read from the shared directory hosting the search index for the master indexer node and written to the index directory on the slave node.

Example: `300`

glassfish.domain.stop.waitSeconds

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

Default: `60`

glassfish.domain.start.maxWaitSeconds

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

Default: `600`

B.3 GlassFish JMS Configuration Settings

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

> **Note:** If you are using CT-MG in a multi-node installation, make sure to change the default setting `imq.brokerid=broker1` to a value that is unique for every node.

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

B.4 CT-MG Configuration Settings

The `configsettings.properties` file contains the configuration settings for the CT-MG services.

AUDIT_LOG_ENABLED

Optional. Specifies whether user operations related to subscriptions, marketable services, and price models are logged and stored in the database. If set to `true`, the operator can export audit log data to retrieve information on the user operations.

Allowed values: `true, false`

Default: `false`

AUDIT_LOG_MAX_ENTRIES_RETRIEVED

Optional. Specifies how many log entries are retrieved in one export of audit log data. If this number is exceeded, a warning is displayed asking the operator to change his filter criteria and start the export again. This setting is required to keep the number of SQL requests to the database low when audit log data is exported. Too many requests may lead to a decrease in system performance.

Allowed values: Any value between `1` and `1000`

Default: `100`

AUTH_MODE

Mandatory. Specifies whether CT-MG is used for user authentication or whether it acts as a SAML service provider. This configuration setting is evaluated at the first startup of CT-MG and must be defined in the `configsettings.properties` file BEFORE starting CT-MG for the first time. It cannot be changed.

Allowed values: `INTERNAL` (CT-MG user authentication is used) or `SAML_SP` (CT-MG shall act as SAML service provider and allow for Web browser single sign-on).

Default: `INTERNAL`

BASE_URL

Mandatory. The base URL is used to access the CT-MG home page. It is particularly required to create the URL for accessing the confirmation page for customers to confirm their registration.

Syntax: `http://<host>:<port>/<context-root>`

The default for `<context-root>` is `fujitsu-bss-portal`.

BASE_URL_HTTPS

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

Syntax: `https://<host>:<port>/<context-root>`

The default for `<context-root>` is `fujitsu-bss-portal`.

CUSTOMER_SELF_REGISTRATION_ENABLED

Optional. Specifies whether customer organizations can register on a marketplace. If set to `false`, the operator needs to create an organization for the customer who wants to register, or a seller (supplier, broker, reseller) needs to register the customer.

Allowed values: `true, false`

Default: `true`

DECIMAL_PLACES

Optional. Specifies the number of decimal places in which usage charges are calculated. This setting is needed only when migrating from CT-MG V14.1 to a higher version.

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

Default: `2`

HIDDEN_UI_ELEMENTS

Optional. Specifies user interface elements to be hidden from the CT-MG administration portal and the marketplaces operated on your platform. You can use this setting to hide user interface elements both from the administration portal and the marketplaces.

Marketplaces

If you want to hide a menu option from the **Account** menu of the marketplaces operated on your platform, enter one of the following values:

- `marketplace.navigation.Profile`: **Profile** menu

- `marketplace.navigation.Payment`: **Payment** menu
- `marketplace.navigation.Subscriptions`: **Subscriptions** menu
- `marketplace.navigation.Users`: **Users** menu
- `marketplace.navigation.Reports`: **Reports** menu
- `marketplace.navigation.Processes`: **Processes** menu

To hide several options from the **Account** menu, separate the options by a comma.

Administration Portal

If you want to hide a specific page from the CT-MG administration portal, you can find out which value needs to be specified here as follows:

1. Open the respective page at the administration portal.
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 administration portal page is `organization.manageVats`. You need to set the configuration key as follows:

`HIDDEN_UI_ELEMENTS=organization.manageVats`

To hide several pages from the administration portal, 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 from the administration portal, enter one of the following values:

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

> **Note:** The **Update configuration settings** page in the **Operation** menu is the default page the operator is directed to when logging in. If you hide the page from the menu or hide the complete menu, you are still directed to the **Update configuration settings** page where you can make changes, if required.

HTTP_PROXY

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

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

HTTP_PROXY_PORT

Optional. The proxy port to be used for PSP-related 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 CT-MG 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 CT-MG 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 CT-MG 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-3.1.2.2/glassfish/domains/bes-domain/config/cakey.der`

IDP_PUBLIC_CERTIFICATE_FILE_PATH

Mandatory. In case a technical service is defined with the user access type and uses CT-MG 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-3.1.2.2/glassfish/domains/bes-domain/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 management.

Allowed values: Any value between `1` and `9223372036854775807`

Default: `100`

LOG_CONFIG_FILE

Optional. The path to the `log4j` configuration file of CT-MG.

Default:

`./log4j.properties` in the `<GLASSFISH_HOME>/glassfish/domains/bes-domain/config` directory

LOG_FILE_PATH

Mandatory. The path to the CT-MG log files.

Default:

`../logs`, which is the `<GLASSFISH_HOME>/glassfish/domains/bes-domain/logs` directory

LOG_LEVEL

Optional. The log level for CT-MG. 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`

MAX_NUMBER_LOGIN_ATTEMPTS

Optional. The maximum number of allowed login attempts to CT-MG. 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`

MP_ERROR_REDIRECT_HTTP

Optional. The URL of a Web page that is to be displayed in case a visitor tries to access a marketplace without a valid marketplace ID by HTTP. This Web page will be shown instead of the default error message.

Syntax: `http://<your Web page>`

Be aware that internet domain names must follow the following rules:

• They must start with a letter and end with a letter or number.

• They may contain letters, numbers, or hyphens only. Special characters are not allowed.

• They may consist of a maximum of 63 characters.

MP_ERROR_REDIRECT_HTTPS

Optional. The URL of a Web page that is to be displayed in case a visitor tries to access a marketplace without a valid marketplace ID by HTTPS. This Web page will be shown instead of the default error message.

Syntax: `https://<your Web page>`

Be aware that internet domain names must follow the following rules:

• They must start with a letter and end with a letter or number.

• They may contain letters, numbers, or hyphens only. Special characters are not allowed.

• They may consist of a maximum of 63 characters.

PERMITTED_PERIOD_INACTIVE_ON_BEHALF_USERS

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

Allowed values: Any value between `1` and `9223372036854775807`

Default: `604800000`, i.e. 7 days

PERMITTED_PERIOD_UNCONFIRMED_ORGANIZATIONS

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

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 CT-MG reports. The URL template of the report engine. If you do not specify a correct URL template, CT-MG will not be able to generate any reports, since the Report Web service cannot be called correctly.

The required value is:

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

Note: The above value must be used as indicated. Do not change this value.

REPORT_SOAP_ENDPOINT

Mandatory if you want to use the CT-MG 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, CT-MG will not be able to generate any reports, since the Report Web service cannot be called correctly.

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

Note:	The above value must be used as indicated. Do not change this value.

REPORT_WSDLURL

Mandatory if you want to use the CT-MG 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, CT-MG 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`

Note:	The above value must be used as indicated. Do not change this value.

SEARCH_INDEX_MASTER_FACTORY_NAME

Mandatory. The name of the search indexer application's JMS connection factory. Stay with the entry `jms/bss/masterIndexerQueueFactory` and do not change it.

SEARCH_INDEX_MASTER_QUEUE_NAME

Mandatory. The name of the search indexer application's JMS queue. Stay with the entry `jms/bss/masterIndexerQueue` and do not change it.

SSO_IDP_TRUSTSTORE

Mandatory if you install CT-MG in SAML_SP authentication mode. The path to the truststore file holding the public key certificate of the IdP (Identity Provider).

Default: `<path>/keystore.jks`

This setting is a runtime property. If you change it in the administration portal, you need to restart CT-MG.

SSO_IDP_TRUSTSTORE_PASSWORD

Mandatory if you install CT-MG in SAML_SP authentication mode. The password of the IdP truststore.

Default: `changeit`

This setting is a runtime property. If you change it in the administration portal, you need to restart CT-MG.

SSO_IDP_URL

Mandatory if you install CT-MG in SAML_SP authentication mode. The SAML **Redirect** URL (location of the IdP service endpoint to be used).

This setting must be defined in the `configsettings.properties` file BEFORE starting CT-MG for the first time.

For security reasons, it is recommended to use the HTTPS protocol.

Syntax: `https://<RedirectServiceEndpoint>`

This setting is a runtime property. If you change it in the administration portal, you need to restart CT-MG.

SSO_ISSUER_ID

Mandatory if you install CT-MG in SAML_SP authentication mode. A unique ID for the CT-MG installation. The IdP or STS uses this ID for identifying incoming requests for SAML assertions.

This configuration setting cannot be changed after the first exchange of data between CT-MG and the IdP or STS.

Default: `CT-MG`

SSO_STS_METADATA_URL

Mandatory if you install CT-MG in SAML_SP authentication mode. The URL of the MEX address (Issuer Metadata Exchange) of the STS.

For security reasons, it is recommended to use the HTTPS protocol.

Syntax: `https://<MEXAddress>`

This setting is an installation property and cannot be changed in the administration portal. If you need to change the URL, proceed as follows:

1. On the machine where CT-MG is deployed, go to the directory

   `<GLASSFISH_HOME>/glassfish/domains/bes-domain/applications/fujitsu-bss`

2. Copy the `fujitsu-adm-um-webservices.jar` file to a different location.

3. Extract the contents of the copied `fujitsu-adm-um-webservices.jar` file and change the value for `SSO_STS_METADATA_URL` in all Web services configuration files (`wsit-com.fujitsu.bss*.xml`).

4. Repackage the `fujitsu-adm-um-webservices.jar` file and copy it back to the directory

   `<GLASSFISH_HOME>/glassfish/domains/bes-domain/applications/fujitsu-bss`

5. Restart the `bes-domain`.

SSO_STS_URL

Mandatory if you install CT-MG in SAML_SP authentication mode. The URL of the security token system service endpoint to be used.

This setting must be defined in the `configsettings.properties` file BEFORE starting CT-MG for the first time.

Be aware that if you provide an invalid URL or if the URL to the STS service endpoint changes, you can change this setting in the `bss` database only.

For security reasons, it is recommended to use the HTTPS protocol.

Syntax: `https://<ServiceEndpoint>`

This setting is an installation property and cannot be changed in the administration portal. If you need to change the URL, proceed as follows:

1. On the machine where CT-MG is deployed, go to the directory

   `<GLASSFISH_HOME>/glassfish/domains/bes-domain/applications/fujitsu-bss`

2. Copy the `fujitsu-adm-um-webservices.jar` file to a different location.

3. Extract the contents of the copied `fujitsu-adm-um-webservices.jar` file and change the value for `SSO_STS_URL` in all Web services configuration files (`wsit-com.fujitsu.bss*.xml`).

4. Repackage the `fujitsu-adm-um-webservices.jar` file and copy it back to the directory

   `<GLASSFISH_HOME>/glassfish/domains/bes-domain/applications/fujitsu-bss`

5. Restart the `bes-domain`.

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 in milliseconds for the timer for billing runs calculating subscription usage costs (customer billing data) or revenue share data. The interval for this timer is one day and cannot be changed. If no offset is defined, the default offset of 4 days is applied.

Customer billing data is calculated for a period of one month (billing period). Suppliers and resellers can define individual start days for their billing periods. Revenue share data is always calculated for the past month on the first day of a month.

The offset for the billing run timer defines the following:

- Number of days after which the billing run calculating the customer billing data or the revenue share data is executed.
- Time the timer for the daily billing runs expires on the current day.

Example:

A supplier defines the 10th of a month as the billing period start date. The offset is set to 4 days and 4 hours. The billing run that calculates the customer billing data for the past billing period of

this supplier is started on the 14th of the following month at 04:00:00.000. The revenue share data is calculated on the 5th of the following month at 04:00:00.000. The daily check whether a billing period of any supplier has ended is started at 04:00:00.000 every day.

Allowed values: Any value between `0` and `2419200000` (28 days)

Default: `345600000`, i.e. 4 days.

TIMER_INTERVAL_DISCOUNT_END_NOTIFICATION_OFFSET

Optional. The offset in milliseconds for the timer for terminating the discounts for all organizations. The timer interval is one day and cannot be changed.

Allowed values: Any value between `0` and `9223372036854775807`

Default: `0`

TIMER_INTERVAL_INACTIVE_ON_BEHALF_USERS

Optional. The time interval in milliseconds at which a check for non-existing users acting on behalf of another organization is executed. 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 CT-MG 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 in milliseconds for the timer for removing inactive "on behalf" users.

Allowed values: Any value between `0` and `9223372036854775807`

Default: `0`

TIMER_INTERVAL_ORGANIZATION

Optional. The time interval in milliseconds at which tasks related to organizations are executed. 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 in milliseconds for the timer for organization-related tasks.

Allowed values: Any value between `0` and `9223372036854775807`

Default: `0`

TIMER_INTERVAL_SUBSCRIPTION_EXPIRATION

Optional. The time interval in milliseconds at which a check for expired subscriptions is executed. 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 in milliseconds for the timer for subscription expiration checks.

Allowed values: Any value between `0` and `9223372036854775807`

Default: `0`

TIMER_INTERVAL_TENANT_PROVISIONING_TIMEOUT

Optional. The time interval in milliseconds at which a check for timed-out subscriptions is executed. 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 in milliseconds for the timer for pending subscription checks.

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 CT-MG. 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 CT-MG configuration settings.

B.5 SAML_SP Configuration Setting

The `sso.properties` file contains the configuration setting for creating the first platform operator in CT-MG when installing CT-MG in SAML_SP authentication mode.

ADMIN_USER_ID

Mandatory if you install CT-MG in SAML_SP authentication mode. The ID of the user who is to become the first CT-MG operator. Each user in CT-MG must relate to an existing user in the Identity Provider system.

User IDs are restricted to 100 characters and must not contain any of the following characters:

! " # $ % & ' * + , / : ; < = > ? \ ^ `

Appendix C: LDAP Keys

The following keys must be defined in a configuration file for enabling access to an organization's LDAP system:

Key	Description

LDAP_URL	Mandatory. Provider URL of the LDAP server. This LDAP server is used for user authentication.
	Example:
	`LDAP_URL=ldap://`
	`myldapserver.lan.est.company.de:389`

LDAP_BASE_DN	Mandatory. Position in the LDAP directory tree at which to start looking for users.
	Example:
	`LDAP_BASE_DN=ou=people,dc=est,dc=mycompany,`
	`dc=de`

LDAP_PRINCIPAL	Optional. Name of the user who is allowed to query the LDAP server. Example: `LDAP_PRINCIPAL=uid=admin,ou=system`
LDAP_CREDENTIALS	Optional. Password of the user who is allowed to query the LDAP server. Example: `LDAP_CREDENTIALS=secret`
LDAP_ATTR_UID	Mandatory. LDAP attribute from which a user ID is read. The default used when an organization is created is `uid` unless the operator has defined a different value in the platform LDAP settings. Example: `LDAP_ATTR_UID=uid`
LDAP_ATTR_EMAIL	Optional. LDAP attribute from which the email address of a user is read. Example: `LDAP_ATTR_EMAIL=scalixEmailAddress`
LDAP_ATTR_FIRST_NAME	Optional. LDAP attribute from which the first name of a user is read. Example: `LDAP_ATTR_FIRST_NAME=givenName`
LDAP_ATTR_LAST_NAME	Optional. LDAP attribute from which the last name of a user is read. Example: `LDAP_ATTR_LAST_NAME=sn`
LDAP_ATTR_ADDITIONAL_NAME	Optional. LDAP attribute from which any additional name of a user is read. Example: `LDAP_ATTR_LAST_NAME=sn`
LDAP_ATTR_LOCALE	Optional. LDAP attribute from which the default language to be stored for a user is read. Example: `LDAP_ATTR_LOCALE=locale`

LDAP_CONTEXT_FACTORY	Mandatory. Context factory which provides the API to query the LDAP server. The default used when an organization is created is `com.sun.jndi.ldap.LdapCtxFactory` unless the operator has defined a different value in the platform LDAP settings.
	Example: `LDAP_CONTEXT_FACTORY=com.sun.jndi.ldap.LdapCtxFactory`

Appendix D: Audit Log

The operator can view and export audit log data on all kinds of user operations related to subscriptions and marketable services including their price models. This may be useful, for example, to check when and by whom a price model was changed, when subscriptions were created, when and by whom a license description was changed.

Single entries of the audit 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.SSS`: The local server date and time.

- `FSP_SW/CT-MG_CTMG-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 level cannot be changed.

- `3<nnnn>`: The ID of the operation that was logged. This is a number between 30000 and 39999. The ID is unique for each operation. For example, the ID of the `Define service` operation is 30090, the ID of the `Subscribe to service` operation is 30000.

- `<operation>`: The operation that was logged, for example, `Define service`.

Following the header, the detailed log message is appended. It consists of name-value pairs with additional information, separated by vertical bars (|).

Every message starts with the following name-value pairs:

- `userId=<user ID>`: The ID of the user who executed the operation.

- `orgId=<organization ID>`: The ID of the organization the user belongs to.

- `orgName=<organization name>`: The name of the organization the user belongs to.

- `<additional name-value pairs>`: Additional name-value pairs specific to the logged operation, separated by vertical bars (|).

The log entries are categorized by the role of the user who executed an operation:

- *User Operations* on page 76
- *Administrator Operations* on page 77
- *Service Manager Operations* on page 79

The log entries show the information as stored in the database. If there are updated values, the initial values are not contained in the audit log file.

D.1 User Operations

This section explains the messages that are output in the audit log for operations executed by standard users on a marketplace.

The following `<additional name-value pairs>` are appended to the header and the name-value pairs common to all log entries:

- `serviceKey=<service key>`: The technical key of a marketable service as stored in the database.

- `serviceName=<service ID>`: The ID of a marketable service as entered during its creation.

- `subscriptionKey=<subscription key>`: The technical key of a subscription as stored in the database.

- `subscriptionName=<subscription ID>`: The name of a subscription as entered when subscribing to a service.

Additional name-value pairs may be appended depending on the logged operation.

Operation: **Execute service operation**

Description: A user selected an operation to be executed for the service he subscribed to. The operations are defined in the technical service underlying the marketable service.

Additional name-value pair:

`serviceOperation=<operation>`: The name of the service operation executed by the user.

D.2 Administrator Operations

This section explains the messages that are output in the audit log for operations executed by an administrator of an organization with any role on a marketplace. Most of these operations can also be executed by subscription managers.

The following `<additional name-value pairs>` are appended to the header and the name-value pairs common to all log entries:

- `serviceKey=<service key>`: The technical key of a marketable service as stored in the database.

- `serviceName=<service ID>`: The ID of a marketable service as entered during its creation.

- `subscriptionKey=<subscription key>`: The technical key of a subscription as stored in the database.

- `subscriptionName=<subscription name>`: The name of a subscription as entered when subscribing to a service.

Additional name-value pairs may be appended depending on the logged operation.

Operation: **Assign service role**

Description: An administrator or subscription manager set or changed a service role for a user assigned to a subscription.

Additional name-value pairs:

- `user=<user ID>`: The ID of the user who was assigned the service role.
- `userRole=<service role>`: The name of the service role as defined in the underlying technical service.

Operation: **Assign user to subscription**

Description: An administrator or subscription manager assigned a user to a subscription.

Additional name-value pair:

`user=<user ID>`: The ID of the user who was assigned to the subscription.

Operation: **Deassign service role**

Description: An administrator or subscription manager removed a service role from a user assigned to a subscription.

Additional name-value pairs:

- `user=<user ID>`: The ID of the user who was deassigned the service role.
- `userRole=<service role>`: The name of the service role as defined in the underlying technical service.

Operation: **Deassign user from subscription**

Description: An administrator or subscription manager removed the assignment of a user to a subscription.

Additional name-value pair:

`user=<user ID>`: The ID of the user who was deassigned from the subscription.

Operation: **Edit customer attribute by customer**

Description: An administrator or subscription manager defined or changed an attribute value for a customer attribute. For every attribute value, a separate log entry is written.

Additional name-value pairs:

- `attributeName=<attribute name>`: The name of the customer attribute.
- `attributeValue=<attribute value>`: The value of the customer attribute.

Operation: **Edit subscription attribute by customer**

Description: An administrator or subscription manager defined or changed an attribute value for a custom subscription attribute. For every attribute value, a separate log entry is written.

Additional name-value pairs:

- `attributeName=<attribute name>`: The name of the custom attribute.
- `attributeValue=<attribute value>`: The value of the custom attribute.

Operation: **Edit subscription owner**

Description: An administrator set or changed the owner of a subscription.

Additional name-value pair:

`subscriptionOwner=<user ID>`: The ID of the administrator or subscription manager who was set as the new owner of the subscription.

Operation: **Edit subscription parameter configuration**

Description: An administrator or subscription manager changed the parameter options for a subscription. For every parameter option, a separate log entry is written.

Additional name-value pairs:

- `parameterName=<option name>`: The name of the parameter option.
- `parameterValue=<option value>`: The new option value. For boolean parameter options, `ON` or `OFF` (`ON`: the option has been selected, `OFF`: the option has not been selected). For enumerations, the value set for the parameter option.

Operation: **Report issue**

Description: An administrator or subscription manager reported an issue on a subscription to the responsible supplier or reseller.

Additional name-value pair:

`issue=<text>`: The subject of the issue as entered by the administrator or subscription manager.

Operation: **Set subscription billing address**

Description: An administrator or subscription manager set or changed the billing address for a subscription.

Additional name-value pairs:

* `address=<display name>`: The name of the billing address as shown to the user.
* `addressDetails=<address details>`: The name of the organization, email, and postal address to which invoices are to be sent.

Operation: **Set subscription payment type**

Description: An administrator or subscription manager set or changed the payment type for a subscription.

Additional name-value pairs:

* `paymentName=<display name>`: The name of the payment type as shown to the user.
* `paymentType=<payment type>`: The payment type name as offered by the supplier or PSP.

Operation: **Subscribe to service**

Description: An administrator or subscription manager completed subscribing to a marketable service offered on a marketplace.

No additional name-value pairs.

Operation: **Unsubscribe from service**

Description: An administrator or subscription manager terminated a subscription.

No additional name-value pairs.

Operation: **Up/downgrade subscription**

Description: An administrator or subscription manager upgraded or downgraded a subscription.

Additional name-value pairs:

* `newServiceKey=<service key>`: The technical key of the service to which the subscription was upgraded or downgraded.
* `newServiceName=<service ID>`: The ID of the service to which the subscription was upgraded or downgraded.

D.3 Service Manager Operations

This section explains the messages that are output in the audit log for operations that can be executed in the administration portal by a service manager of an organization with the supplier role. Some of these operations can also be executed by brokers or resellers.

The following *<additional name-value pairs>* are appended to the header and the name-value pairs common to all log entries:

- `serviceKey=`*<service key>*: The technical key of a marketable service as stored in the database.

- `serviceName=`*<service ID>*: The ID of a marketable service as entered during its creation.

Additional name-value pairs may be appended depending on the logged operation.

Operation: **Activate/deactivate service**

Description: A service manager, broker, or reseller activated or deactivated a marketable service.

Additional name-value pairs:

- `marketplaceId=`*<ID>*: The ID of the marketplace to which the service is published.

- `marketplaceName=`*<name>*: The name of the marketplace to which the service is published.

- `activation=`*<on or off>*: Specifies whether the service was activated or deactivated. Can be `ON` or `OFF`.

- `inCatalog=`*<on or off>*: Specifies whether the service is to be displayed in the service catalog. Can be `ON` or `OFF`.

Operation: **Assign brokers**

Description: In the publishing options, a service manager authorized a broker to sell the supplier's services.

Additional name-value pair:

`BrokerId=`*<ID>*: The organization ID of the assigned broker.

Operation: **Assign categories**

Description: In the publishing options, a service manager, broker, or reseller assigned or deassigned one or several categories to/from a marketable service.

Additional name-value pair:

`listOfCategories=`*<names>*: The categories that are assigned, separated by commas.

Operation: **Assign resellers**

Description: In the publishing options, a service manager authorized a reseller to sell the supplier's services.

Additional name-value pair:

`ResellerId=`*<ID>*: The organization ID of the assigned reseller.

Operation: **Assign service to marketplace**

Description: In the publishing options, a service manager, broker, or reseller specified or changed the marketplace to which a marketable service is to be published.

Additional name-value pairs:

- `marketplaceId=`*<ID>*: The ID of the marketplace.

- `marketplaceName=`*<name>*: The name of the marketplace.

Operation: **Copy service**

Description: A service manager created a copy of a marketable service.

Additional name-value pairs:

- `copyId=<service ID>`: The ID of the marketable service copy as stored in the database.
- `copyName=<service name>`: The name of the marketable service copy as entered when the copy was created.

Operation: **Deassign brokers**

Description: In the publishing options, a service manager removed a broker's right to sell the supplier's services.

Additional name-value pair:

`BrokerId=<ID>`: The organization ID of the deassigned broker.

Operation: **Deassign resellers**

Description: In the publishing options, a service manager removed a reseller's right to sell the supplier's services.

Additional name-value pair:

`ResellerId=<ID>`: The organization ID of the deassigned reseller.

Operation: **Define service**

Description: A service manager defined a marketable service. One additional **Update service parameter** log entry is created for each parameter and parameter option defined in the service.

Additional name-value pairs:

- `technServiceName=<name>`: The name of the technical service the marketable service is based on.
- `shortDescription=<yes or no>`: Specifies whether the short description of the marketable service was entered. Can be `YES` or `NO`.
- `description=<yes or no>`: Specifies whether the description of the marketable service was entered. Can be `YES` or `NO`.
- `locale=<language code>`: The code of the language in which the descriptions were saved.
- `autoAssignUser=<yes or no>`: Specifies whether the user subscribing to the service is automatically assigned to the subscription. Can be `YES` or `NO`.

Operation: **Define up/downgrade options**

Description: A service manager defined options to which a subscription can be upgraded or downgraded, or he removed a service from the list of up/downgrade options.

Additional name-value pairs:

- `targetId=<service ID>`: The ID of the marketable service to which subscriptions can be up/downgraded as stored in the database.
- `targetName=<service name>`: The name of the marketable service to which subscriptions can be up/downgraded.
- `upDownGrade=<on or off>`: Specifies whether the service was added or removed as an up/downgrade option. Can be `ON` or `OFF`.

Operation: **Delete customer price model**

Description: A service manager deleted a customer-specific price model.

Additional name-value pairs:

- `customerKey=`*`<organization ID>`*: The ID of the customer organization.
- `customerName=`*`<organization name>`*: The name of the customer organization.

Operation: **Delete service**

Description: A service manager deleted a marketable service.

No additional name-value pairs.

Operation: **Edit event price in customer price model**

Description: A service manager defined or changed the price for an event in a customer-specific price model.

Additional name-value pairs:

- `customerKey=`*`<organization ID>`*: The ID of the customer organization.
- `customerName=`*`<organization name>`*: The name of the customer organization.
- `currency=`*`<currency code>`*: The ISO code of the currency in which a customer is charged for using the service.
- `eventName=`*`<name>`*: The name of the event as defined in the underlying technical service.
- `range=`*`<value>`*: If stepped prices are defined, this is the step limit up to which the price applies. If no stepped prices are defined, the value is `1-ANY`.
- `action=`*`<action>`*: If stepped prices are defined, action that was executed. Can be `INSERT` (when a new stepped price was defined), `UPDATE` (when a stepped price was updated), or `DELETE` (when a stepped price was deleted. If no stepped prices are defined, the `action` parameter is not written to the audit log.
- `price=`*`<price>`*: The price as defined in the price model.

Operation: **Edit event price in service price model**

Description: A service manager defined or changed the price for an event in a service price model.

Additional name-value pairs:

- `currency=`*`<currency code>`*: The ISO code of the currency in which a customer is charged for using the service.
- `eventName=`*`<name>`*: The name of the event as defined in the underlying technical service.
- `range=`*`<value>`*: If stepped prices are defined, this is the step limit up to which the price applies. If no stepped prices are defined, the value is `1-ANY`.
- `action=`*`<action>`*: If stepped prices are defined, action that was executed. Can be `INSERT` (when a new stepped price was defined), `UPDATE` (when a stepped price was updated), or `DELETE` (when a stepped price was deleted. If no stepped prices are defined, the `action` parameter is not written to the audit log.
- `price=`*`<price>`*: The price as defined in the price model.

Operation: **Edit event price in subscription price model**

Description: A service manager defined or changed the price for an event in a subscription-specific price model.

Additional name-value pairs:

- `customerKey=<organization ID>`: The ID of the customer organization.
- `customerName=<organization name>`: The name of the customer organization.
- `subscriptionKey=<subscription key>`: The technical key of the subscription as stored in the database.
- `subscriptionName=<subscription name>`: The name of the subscription as entered when subscribing to a service.
- `currency=<currency code>`: The ISO code of the currency in which a customer is charged for using the service.
- `eventName=<name>`: The name of the event as defined in the underlying technical service.
- `range=<value>`: If stepped prices are defined, this is the step limit up to which the price applies. If no stepped prices are defined, the value is `1-ANY`.
- `action=<action>`: If stepped prices are defined, action that was executed. Can be `INSERT` (when a new stepped price was defined), `UPDATE` (when a stepped price was updated), or `DELETE` (when a stepped price was deleted. If no stepped prices are defined, the `action` parameter is not written to the audit log.
- `price=<price>`: The price as defined in the price model.

Operation: **Edit one time fee in customer price model**

Description: A service manager defined or changed a one-time fee in a customer-specific price model.

Additional name-value pairs:

- `customerKey=<organization ID>`: The ID of the customer organization.
- `customerName=<organization name>`: The name of the customer organization.
- `currency=<currency code>`: The ISO code of the currency in which a customer is charged for using the service.
- `timeUnit=<time unit>`: The time unit defined for recurring charges. Can be `MONTH`, `WEEK`, `DAY`, or `HOUR`.
- `oneTimeFee=<fee>`: The one-time fee for a subscription.

Operation: **Edit one time fee in service price model**

Description: A service manager defined or changed a one-time fee in a service price model.

Additional name-value pairs:

- `currency=<currency code>`: The ISO code of the currency in which a customer is charged for using the service.
- `timeUnit=<time unit>`: The time unit defined for recurring charges. Can be `MONTH`, `WEEK`, `DAY`, or `HOUR`.
- `oneTimeFee=<fee>`: The one-time fee for a subscription.

Operation: **Edit price model type for customer price model**

Description: A service manager defined or changed the calculation mode, and/or a free trial period of a customer-specific price model.

Additional name-value pairs:

- `customerKey=<organization ID>`: The ID of the customer organization.
- `customerName=<organization name>`: The name of the customer organization.
- `currency=<currency code>`: The ISO code of the currency in which a customer is charged for using the service.
- `timeUnit=<time unit>`: The time unit for recurring charges and per time unit calculation. Can be `MONTH`, `WEEK`, `DAY`, or `HOUR`.
- `calculationMode=<calculation option>`: The way how charges for using a service are calculated. Can be `PRO_RATA` or `PER_UNIT`.
- `trialPeriod=<on or off>`: Specifies whether a free trial period has been defined in the price model. Can be `ON` or `OFF`.
- `daysOfTrial=<Number of days>`: The number of days defined for the free trial period.

Operation: **Edit price model type for service price model**

Description: A service manager defined or changed the calculation mode and/or a free trial period of a service price model.

Additional name-value pairs:

- `currency=<currency code>`: The ISO code of the currency in which a customer is charged for using the service.
- `timeUnit=<time unit>`: The time unit for recurring charges and per time unit calculation. Can be `MONTH`, `WEEK`, `DAY`, or `HOUR`.
- `calculationMode=<calculation option>`: The way how charges for using a service are calculated. Can be `PRO_RATA` or `PER_UNIT`.
- `trialPeriod=<on or off>`: Specifies whether a free trial period has been defined in the price model. Can be `ON` or `OFF`.
- `daysOfTrial=<Number of days>`: The number of days defined for the free trial period.

Operation: **Edit price model type to free of charge for customer**

Description: A service manager defined or changed a free-of-charge, customer-specific price model so that no costs are charged for using the service.

Additional name-value pairs:

- `customerKey=<organization ID>`: The ID of the customer organization.
- `customerName=<organization name>`: The name of the customer organization.

Operation: **Edit price model type to free of charge for service**

Description: A service manager defined or changed a free-of-charge service price model so that no costs are charged for using the service.

No additional name-value pairs.

Operation: **Edit price per subscription parameter in customer price model**

Description: A service manager defined or changed the price per subscription for a parameter or a parameter option in a customer-specific price model.

Additional name-value pairs:

- `customerKey=<organization ID>`: The ID of the customer organization.

- `customerName=<organization name>`: The name of the customer organization.

- `currency=<currency code>`: The ISO code of the currency in which a customer is charged for using the service.

- `parameterName=<name>`: The name of the parameter as defined in the underlying technical service.

- `range=<value>`: For numeric parameters, stepped prices can be defined. If this is the case, this is the step limit up to which the price applies. If no stepped prices are defined, the value is `1-ANY`.

- `optionName=<name>`: The name of the parameter option as defined in the underlying technical service.

- `action=<action>`: If stepped prices are defined, action that was executed. Can be `INSERT` (when a new stepped price was defined), `UPDATE` (when a stepped price was updated), or `DELETE` (when a stepped price was deleted. If no stepped prices are defined, the `action` parameter is not written to the audit log.

- `price=<price>`: The price as defined in the price model.

Operation: **Edit price per subscription parameter in service price model**

Description: A service manager defined or changed the price per subscription for a parameter or a parameter option in a service price model.

Additional name-value pairs:

- `currency=<currency code>`: The ISO code of the currency in which a customer is charged for using the service.

- `parameterName=<name>`: The name of the parameter as defined in the underlying technical service.

- `range=<value>`: For numeric parameters, stepped prices can be defined. If this is the case, this is the step limit up to which the price applies. If no stepped prices are defined, the value is `1-ANY`.

- `optionName=<name>`: The name of the parameter option as defined in the underlying technical service.

- `action=<action>`: If stepped prices are defined, action that was executed. Can be `INSERT` (when a new stepped price was defined), `UPDATE` (when a stepped price was updated), or `DELETE` (when a stepped price was deleted. If no stepped prices are defined, the `action` parameter is not written to the audit log.

- `price=<price>`: The price as defined in the price model.

Operation: **Edit price per subscription parameter in subscription price model**

Description: A service manager defined or changed the price per subscription for a parameter or a parameter option in a subscription-specific price model.

Additional name-value pairs:

- `customerKey=<organization ID>`: The ID of the customer organization.
- `customerName=<organization name>`: The name of the customer organization.
- `subscriptionKey=<subscription key>`: The technical key of the subscription as stored in the database.
- `subscriptionName=<subscription name>`: The name of the subscription as entered when subscribing to a service.
- `currency=<currency code>`: The ISO code of the currency in which a customer is charged for using the service.
- `parameterName=<name>`: The name of the parameter as defined in the underlying technical service.
- `range=<value>`: For numeric parameters, stepped prices can be defined. If this is the case, this is the step limit up to which the price applies. If no stepped prices are defined, the value is `1-ANY`.
- `optionName=<name>`: The name of the parameter option as defined in the underlying technical service.
- `action=<action>`: If stepped prices are defined, action that was executed. Can be `INSERT` (when a new stepped price was defined), `UPDATE` (when a stepped price was updated), or `DELETE` (when a stepped price was deleted. If no stepped prices are defined, the `action` parameter is not written to the audit log.
- `price=<price>`: The price as defined in the price model.

Operation: **Edit price per user parameter in customer price model**

Description: A service manager defined or changed the price per user for a parameter or a parameter option in a customer-specific price model.

Additional name-value pairs:

- `customerKey=<organization ID>`: The ID of the customer organization.
- `customerName=<organization name>`: The name of the customer organization.
- `currency=<currency code>`: The ISO code of the currency in which a customer is charged for using the service.
- `parameterName=<name>`: The name of the parameter as defined in the underlying technical service.
- `optionName=<name>`: The name of the parameter option as defined in the underlying technical service.
- `price=<price>`: The price as defined in the price model.

Operation: **Edit price per user parameter in service price model**

Description: A service manager defined or changed the price per user for a parameter or a parameter option in a service price model.

Additional name-value pairs:

*   `currency=<currency code>`: The ISO code of the currency in which a customer is charged for using the service.
*   `parameterName=<name>`: The name of the parameter as defined in the underlying technical service.
*   `optionName=<name>`: The name of the parameter option as defined in the underlying technical service.
*   `price=<price>`: The price as defined in the price model.

Operation: **Edit price per user parameter in subscription price model**

Description: A service manager defined or changed the price per user for a parameter or a parameter option in a subscription-specific price model.

Additional name-value pairs:

*   `customerKey=<organization ID>`: The ID of the customer organization.
*   `customerName=<organization name>`: The name of the customer organization.
*   `subscriptionKey=<subscription key>`: The technical key of the subscription as stored in the database.
*   `subscriptionName=<subscription name>`: The name of the subscription as entered when subscribing to a service.
*   `currency=<currency code>`: The ISO code of the currency in which a customer is charged for using the service.
*   `parameterName=<name>`: The name of the parameter as defined in the underlying technical service.
*   `optionName=<name>`: The name of the parameter option as defined in the underlying technical service.
*   `price=<price>`: The price as defined in the price model.

Operation: **Edit price per user role parameter in customer price model**

Description: A service manager defined or changed the price per user with a specific service role for a parameter or a parameter option in a customer-specific price model.

Additional name-value pairs:

- `customerKey=<organization ID>`: The ID of the customer organization.
- `customerName=<organization name>`: The name of the customer organization.
- `currency=<currency code>`: The ISO code of the currency in which a customer is charged for using the service.
- `timeUnit=<time unit>`: The time unit defined for recurring charges. Can be `MONTH`, `WEEK`, `DAY`, or `HOUR`.
- `userRole=<role name>`: The name of the service role as defined in the underlying technical service.
- `parameterName=<name>`: The name of the parameter as defined in the underlying technical service.
- `optionName=<name>`: The name of the parameter option as defined in the underlying technical service.
- `price=<price>`: The price as defined in the price model.

Operation: **Edit price per user role parameter in service price model**

Description: A service manager defined or changed the price per user with a specific service role for a parameter or a parameter option in a service price model.

Additional name-value pairs:

- `currency=<currency code>`: The ISO code of the currency in which a customer is charged for using the service.
- `timeUnit=<time unit>`: The time unit defined for recurring charges. Can be `MONTH`, `WEEK`, `DAY`, or `HOUR`.
- `userRole=<role name>`: The name of the service role as defined in the underlying technical service.
- `parameterName=<name>`: The name of the parameter as defined in the underlying technical service.
- `optionName=<name>`: The name of the parameter option as defined in the underlying technical service.
- `price=<price>`: The price as defined in the price model.

Operation: **Edit price per user role parameter in subscription price model**

Description: A service manager defined or changed the price per user with a specific service role for a parameter or a parameter option in a subscription-specific specific price model.

Additional name-value pairs:

- `customerKey=<organization ID>`: The ID of the customer organization.
- `customerName=<organization name>`: The name of the customer organization.
- `subscriptionKey=<subscription key>`: The technical key of the subscription as stored in the database.
- `subscriptionName=<subscription name>`: The name of the subscription as entered when subscribing to a service.
- `currency=<currency code>`: The ISO code of the currency in which a customer is charged for using the service.
- `timeUnit=<time unit>`: The time unit defined for recurring charges. Can be MONTH, WEEK, DAY, or HOUR.
- `userRole=<role name>`: The name of the service role as defined in the underlying technical service.
- `parameterName=<name>`: The name of the parameter as defined in the underlying technical service.
- `optionName=<name>`: The name of the parameter option as defined in the underlying technical service.
- `price=<price>`: The price as defined in the price model.

Operation: **Edit recurring charge for subscription in customer price model**

Description: A service manager defined or changed the recurring charge per subscription in a customer-specific price model.

Additional name-value pairs:

- `customerKey=<organization ID>`: The ID of the customer organization.
- `customerName=<organization name>`: The name of the customer organization.
- `currency=<currency code>`: The ISO code of the currency in which a customer is charged for using the service.
- `timeUnit=<time unit>`: The time unit defined for recurring charges. Can be MONTH, WEEK, DAY, or HOUR.
- `oneTimeFee=<fee>`: The one-time fee for a subscription.
- `recurringCharge=<charge>`: The recurring charge per subscription.

Operation: **Edit recurring charge for subscription in service price model**

Description: A service manager defined or changed the recurring charge per subscription in a service price model.

Additional name-value pairs:

- `currency=<currency code>`: The ISO code of the currency in which a customer is charged for using the service.
- `timeUnit=<time unit>`: The time unit defined for recurring charges. Can be `MONTH`, `WEEK`, `DAY`, or `HOUR`.
- `oneTimeFee=<fee>`: The one-time fee for a subscription.
- `recurringCharge=<charge>`: The recurring charge per subscription.

Operation: **Edit recurring charge for subscription in subscription price model**

Description: A service manager defined or changed the recurring charge per subscription in a subscription-specific price model.

Additional name-value pairs:

- `customerKey=<organization ID>`: The ID of the customer organization.
- `customerName=<organization name>`: The name of the customer organization.
- `subscriptionKey=<subscription key>`: The technical key of the subscription as stored in the database.
- `subscriptionName=<subscription name>`: The name of the subscription as entered when subscribing to a service.
- `currency=<currency code>`: The ISO code of the currency in which a customer is charged for using the service.
- `timeUnit=<time unit>`: The time unit defined for recurring charges. Can be `MONTH`, `WEEK`, `DAY`, or `HOUR`.
- `oneTimeFee=<fee>`: The one-time fee for a subscription.
- `recurringCharge=<charge>`: The recurring charge per subscription.

Operation: **Edit recurring charge for users in customer price model**

Description: A service manager defined or changed the recurring charge for users in a customer-specific price model.

Additional name-value pairs:

- `customerKey=<organization ID>`: The ID of the customer organization.

- `customerName=<organization name>`: The name of the customer organization.

- `currency=<currency code>`: The ISO code of the currency in which a customer is charged for using the service.

- `timeUnit=<time unit>`: The time unit defined for recurring charges. Can be `MONTH`, `WEEK`, `DAY`, or `HOUR`.

- `range=<value>`: If stepped prices are defined, this is the step limit up to which the price applies. If no stepped prices are defined, the value is `1-ANY`.

- `action=<action>`: If stepped prices are defined, action that was executed. Can be `INSERT` (when a new stepped price was defined), `UPDATE` (when a stepped price was updated), or `DELETE` (when a stepped price was deleted. If no stepped prices are defined, the `action` parameter is not written to the audit log.

- `recurringCharge=<charge>`: The recurring charge for users.

Operation: **Edit recurring charge for users in service price model**

Description: A service manager defined or changed the recurring charge for users in a service price model.

Additional name-value pairs:

- `currency=<currency code>`: The ISO code of the currency in which a customer is charged for using the service.

- `timeUnit=<time unit>`: The time unit defined for recurring charges. Can be `MONTH`, `WEEK`, `DAY`, or `HOUR`.

- `range=<value>`: If stepped prices are defined, this is the step limit up to which the price applies. If no stepped prices are defined, the value is `1-ANY`.

- `action=<action>`: If stepped prices are defined, action that was executed. Can be `INSERT` (when a new stepped price was defined), `UPDATE` (when a stepped price was updated), or `DELETE` (when a stepped price was deleted. If no stepped prices are defined, the `action` parameter is not written to the audit log.

- `recurringCharge=<charge>`: The recurring charge for users.

Operation: **Edit recurring charge for users in subscription price model**

Description: A service manager defined or changed the recurring charge for users in a subscription-specific price model.

Additional name-value pairs:

- `customerKey=<organization ID>`: The ID of the customer organization.

- `customerName=<organization name>`: The name of the customer organization.

- `subscriptionKey=<subscription key>`: The technical key of the subscription as stored in the database.

- `subscriptionName=<subscription name>`: The name of the subscription as entered when subscribing to a service.

- `currency=<currency code>`: The ISO code of the currency in which a customer is charged for using the service.

- `timeUnit=<time unit>`: The time unit defined for recurring charges. Can be `MONTH`, `WEEK`, `DAY`, or `HOUR`.

- `range=<value>`: If stepped prices are defined, this is the step limit up to which the price applies. If no stepped prices are defined, the value is `1-ANY`.

- `action=<action>`: If stepped prices are defined, action that was executed. Can be `INSERT` (when a new stepped price was defined), `UPDATE` (when a stepped price was updated), or `DELETE` (when a stepped price was deleted. If no stepped prices are defined, the `action` parameter is not written to the audit log.

- `recurringCharge=<charge>`: The recurring charge for users.

Operation: **Edit service role prices for customer price model**

Description: A service manager defined or changed the recurring charge for users with a given service role in a customer-specific price model.

Additional name-value pairs:

- `customerKey=<organization ID>`: The ID of the customer organization.

- `customerName=<organization name>`: The name of the customer organization.

- `currency=<currency code>`: The ISO code of the currency in which a customer is charged for using the service.

- `timeUnit=<time unit>`: The time unit defined for recurring charges. Can be `MONTH`, `WEEK`, `DAY`, or `HOUR`.

- `userRole=<role name>`: The name of the service role as defined in the underlying technical service.

- `price=<price>`: The recurring charge for users having the selected role.

Operation: **Edit service role prices for service price model**

Description: A service manager defined or changed the recurring charge for users with a given service role in a service price model.

Additional name-value pairs:

- `currency=<currency code>`: The ISO code of the currency in which a customer is charged for using the service.

- `timeUnit=<time unit>`: The time unit defined for recurring charges. Can be `MONTH`, `WEEK`, `DAY`, or `HOUR`.

- `userRole=<role name>`: The name of the service role as defined in the underlying technical service.

- `price=<price>`: The recurring charge for users having the selected role.

Operation: **Edit service role prices for subscription price model**

Description: A service manager defined or changed the recurring charge for users with a given service role in a subscription-specific price model.

Additional name-value pairs:

- `customerKey=<organization ID>`: The ID of the customer organization.

- `customerName=<organization name>`: The name of the customer organization.

- `subscriptionKey=<subscription key>`: The technical key of the subscription as stored in the database.

- `subscriptionName=<subscription name>`: The name of the subscription as entered when subscribing to a service.

- `currency=<currency code>`: The ISO code of the currency in which a customer is charged for using the service.

- `timeUnit=<time unit>`: The time unit defined for recurring charges. Can be `MONTH`, `WEEK`, `DAY`, or `HOUR`.

- `userRole=<role name>`: The name of the service role as defined in the underlying technical service.

- `price=<price>`: The recurring charge for users having the selected role.

Operation: **Edit subscription attribute by service manager**

Description: A service manager defined or changed an attribute value for a custom subscription attribute. For every attribute value, a separate log entry is written.

Additional name-value pairs:

- `subscriptionKey=<subscription key>`: The technical key of the subscription as stored in the database.

- `subscriptionName=<subscription name>`: The name of the subscription as entered when subscribing to a service.

- `attributeName=<attribute name>`: The name of the custom attribute.

- `attributeValue=<attribute value>`: The value of the custom attribute.

Operation: **Localize price model for customer**

Description: A service manager translated the price model elements for a customer into a given language.

Additional name-value pairs:

- `customerKey=<ID>`: The ID of the customer organization.

- `customerName=<name>`: The name of the customer organization.

- `locale=<language code>`: The code of the language in which the texts were saved.

- `description=<yes or no>`: Specifies whether the description of the price model was changed. Can be `YES` or `NO`.

- `license=<yes or no>`: Specifies whether the license information was changed. Can be `YES` or `NO`.

Operation: **Localize price model for service**

Description: A service manager translated the price model elements for a service into a given language.

Additional name-value pairs:

- `locale=<language code>`: The code of the language in which the texts were saved.

- `description=<yes or no>`: Specifies whether the description of the price model was changed. Can be `YES` or `NO`.

- `license=<yes or no>`: Specifies whether the license information was changed. Can be `YES` or `NO`.

Operation: **Localize price model for subscription**

Description: A service manager translated the price model elements for a subscription into a given language.

Additional name-value pairs:

- `customerKey=<ID>`: The ID of the customer organization.

- `customerName=<name>`: The name of the customer organization.

- `subscriptionKey=<subscription key>`: The technical key of the subscription as stored in the database.

- `subscriptionName=<subscription name>`: The name of the subscription as entered when subscribing to a service.

- `locale=<language code>`: The code of the language in which the texts were saved.

- `description=<yes or no>`: Specifies whether the description of the price model was changed. Can be `YES` or `NO`.

- `license=<yes or no>`: Specifies whether the license information was changed. Can be `YES` or `NO`.

Operation: **Localize service**

Description: A service manager added a service description or short description in another language.

Additional name-value pairs:

- `shortDescription=<yes or no>`: Specifies whether the short description of the marketable service was changed. Can be `YES` or `NO`.

- `description=<yes or no>`: Specifies whether the description of the marketable service was changed. Can be `YES` or `NO`.

- `locale=<language code>`: The code of the language in which the descriptions were saved.

Operation: **Set service as public**

Description: In the publishing options, a service manager, broker, or reseller specified that a marketable service is to be public or that a public marketable service is no longer to be public.

Additional name-value pair:

`public=<true or false>`: Specifies whether the service has been marked as public. Can be `true` (service is public) or `false` (service is not public).

Operation: **Terminate subscription**

Description: A service manager or reseller explicitly terminated a customer's subscription.

Additional name-value pairs:

- `subscriptionKey=<subscription key>`: The technical key of the subscription as stored in the database.

- `subscriptionName=<subscription name>`: The name of the subscription as entered when subscribing to a service.

- `reason=<termination reason>`: Text explaining the reason for terminating the subscription.

Operation: **Update service**

Description: A service manager changed a marketable service.

Additional name-value pairs:

- `shortDescription=<yes or no>`: Specifies whether the short description of the marketable service was changed. Can be `YES` or `NO`.

- `description=<yes or no>`: Specifies whether the description of the marketable service was changed. Can be `YES` or `NO`.

- `locale=<language code>`: The code of the language in which the descriptions were saved.

- `autoAssignUser=<yes or no>`: Specifies whether the user subscribing to the service is automatically assigned to the subscription. Can be `YES` or `NO`.

Operation: **Update service parameter**

Description: A service manager defined or changed a parameter or parameter option.

Additional name-value pairs:

- `parameterName=<name>`: Name of the parameter as defined in the underlying technical service.
- `userOption=<on or off>`: Specifies whether the parameter is offered as an option to customers who subscribe to the marketable service. Can be `ON` or `OFF`.
- `parameterValue=<value>`: For a parameter, the actual string or number; for a parameter option, this can be `YES` or `NO`.

Operation: **View subscription**

Description: A service manager, broker, or reseller displayed the details of a subscription.

Additional name-value pairs:

- `subscriptionKey=<subscription key>`: The technical key of the subscription as stored in the database.
- `subscriptionName=<subscription name>`: The name of the subscription as entered when subscribing to a service.
- `customerKey=<organization ID>`: The ID of the customer organization.
- `customerName=<organization name>`: The name of the customer organization.

Appendix E: Menu Options and Required Roles

The roles of an organization determine which functions are available to its users at the CT-MG 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 organizations of all roles.

Function	Organization Role	User Role
Edit profile	Any	Any
Import users (if the organization uses LDAP-based user management)	Any	Administrator
Change password (if the organization does not use LDAP-based user management)	Any	Any
Register new users (if the organization does not use LDAP-based user management)	Any	Administrator
Manage users	Any	Administrator
LDAP settings (if the organization uses LDAP-based user management)	Any	Administrator
Create report	Any	Administrator
Manage suppliers	Technology provider	Technology manager
Manage custom attributes	Supplier	Service manager
Process triggers	Any	Administrator
Manage processes (if the organization is connected to an external process control system)	Any	Any
Export billing data	Supplier, reseller, broker, marketplace owner, operator	Service manager, reseller, broker, marketplace manager, operator

Function	Organization Role	User Role
Define billing period	Supplier, reseller	Service manager, reseller

Customer Menu

This menu is available to supplier, reseller, and broker organizations.

Function	Organization Role	User Role
Register customer	Supplier, reseller, broker	Service manager, reseller, broker
View customer	Reseller, broker	Reseller, broker
Manage customer	Supplier	Service manager
Manage payment types	Supplier, Reseller	Service manager, reseller
Manage VAT rates	Supplier	Service manager
View subscription	Supplier, reseller, broker	Service manager, reseller, broker
Manage subscription attributes	Supplier	Service manager
Terminate subscription	Supplier, reseller	Service manager, reseller

Marketable Service Menu

This menu is available to supplier, reseller, and broker organizations.

Function	Organization Role	User Role
Define service	Supplier	Service manager
Update service	Supplier	Service manager
View service	Broker	Broker
Manage service	Reseller	Reseller
Copy service	Supplier	Service manager
Delete service	Supplier	Service manager
Define up/downgrade options	Supplier	Service manager
Define publishing options	Supplier, reseller, broker	Service manager, reseller, broker
Activate or deactivate services	Supplier, reseller, broker	Service manager, reseller, broker

Marketplace Menu

This menu is available to marketplace owner and operator organizations.

Function	Organization Role	User Role
Manage categories	Marketplace owner	Marketplace manager
Manage sellers	Marketplace owner	Marketplace manager
Create marketplace	Operator	Operator
Update marketplace	Marketplace owner, operator	Marketplace manager or operator. The operator can assign a different marketplace owner and define revenue shares; the marketplace manager can change the name and settings of the marketplace.
Delete marketplace	Operator	Operator
Manage broker revenue share	Operator	Operator
Manage reseller revenue share	Operator	Operator
Add tracking code	Marketplace owner	Marketplace manager
Define featured services	Marketplace owner	Marketplace manager
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 operator revenue share	Operator	Operator
Manage timers	Operator	Operator
Manage payment service provider	Operator	Operator
Manage currencies	Operator	Operator

Function	Organization Role	User Role
Manage LDAP settings	Operator	Operator
Update configuration settings	Operator	Operator
Billing data preview	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 service definition	Technology provider	Technology manager
Import service definition	Technology provider	Technology manager
Update service definition	Technology provider	Technology manager
Export service definition	Technology provider	Technology manager
Delete service definition	Technology provider	Technology manager

Glossary

Administrator

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

Application

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

Broker

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

Cloud

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

Cloud Computing

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

Customer

An organization which subscribes to one or more marketable services in CT-MG 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 CT-MG, based on a technical service. A marketable service defines prices, conditions, and restrictions for using the underlying application.

Marketplace

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

Marketplace Owner

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

Marketplace Manager

A privileged user role within a marketplace owner organization.

Operator

An organization or person responsible for maintaining and operating CT-MG.

Organization

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

Payment Service Provider (PSP)

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

Payment Type

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

Platform as a Service (PaaS)

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

Price Model

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

Reseller

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

Role

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

Seller

Collective term for supplier, broker, and reseller organizations.

Service

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

Subscription Manager

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

Supplier

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

Technical Service

The representation of an application in CT-MG. 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 CT-MG.