FUJITSU

FUJITSU Software Enterprise Service Catalog Manager V17.6

Supplier's Guide

January 2018

Trademarks	LINUX is a registered trademark of Linus Torvalds. Open Service Catalog Manager is a registered trademark of FUJITSU LIMITED. The OpenStack Word Mark and OpenStack logo are registered trademarks/service marks or trademarks/ service marks of the OpenStack Foundation in the United States and other countries. Apache Tomcat, Tomcat, and Apache are trademarks of The Apache Software Foundation. Java is a registered trademark of Oracle and/or its affiliates. UNIX is a registered trademark of the Open Group in the United States and in other countries. Other company names and product names are trademarks or registered trademarks of their respective owners.
Copyright FUJITSU ENABLING SOFTWARE TECHNOLOGY GMBH 2018	All rights reserved, including those of translation into other languages. No part of this manual may be reproduced in any form whatsoever without the written permission of FUJITSU ENABLING SOFTWARE TECHNOLOGY GMBH.

High Risk Activity

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

Export Restrictions

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

Contents

About this Manual 5

1 Introduction 7

1.1 The Supplier's Tasks in ESCM 8

1.2 Accessing ESCM 9

2 Marketable Services 11

2.1 Defining Marketable Services 11

2.1.1 Creating Marketable Services 12

2.1.2 Defining Upgrade and Downgrade Options 13

2.2 Publishing Marketable Services 14

2.3 Working with Brokers and Resellers 17

2.4 Maintaining Marketable Services 18

3 Price Models 21

3.1 Price Model Scope 21

3.2 Price Model Elements 21

3.2.1 One-Time Fee 22

3.2.2 Recurring Charge per Subscription 23

3.2.3 Recurring Charge for Users 23

3.2.4 Prices for Parameters and Options 25

3.2.5 Prices for Events 26

3.2.6 Role-Based Prices 27

3.2.7 Stepped Prices 28

3.2.8 Free Trial Period 30

3.2.9 Discounts 31

3.3 Defining Price Models 31

3.4 Maintaining Price Models 32

4 Customer Management 34

4.1 Registering Customers 34

4.2 Configuring an External User Authentication System 34

4.3 Defining Custom Attributes 35

4.4 Handling Subscriptions....36

5 Billing and Payment.... 37

5.1 Managing Billing and Payment for Customers....37
5.1.1 Managing Payment Types.... 38
5.1.2 Managing VAT Rates.... 38
5.1.3 Defining the Billing Period.... 39
5.1.4 Exporting Billing Data.... 39

5.2 Handling Revenue Shares.... 40

6 Reporting.... 41

7 Integrating External Process Control.... 42

Appendix A LDAP Keys....44

Appendix B Customer Billing Data.... 46

Glossary 63

About this Manual

This manual describes how service providers can define and manage service offerings for applications with) FUJITSU Software Enterprise Service Catalog Manager, hereafter referred to as ESCM.

This manual is structured as follows:

Chapter	Description
Introduction on page 7	Outlines the role of a supplier in ESCM and gives an overview of the supplier's tasks.
Marketable Services on page 11	Describes the basic concepts of marketable services and explains how to define and manage marketable services as well as how to involve brokers and resellers in offering and selling the services.
Price Models on page 21	Describes the basic concepts of price models and explains how to define and maintain price models.
Customer Management on page 34	Describes the different tasks involved in customer management.
Billing and Payment on page 37	Describes how ESCM supports suppliers in collecting payments from customers and calculating their revenue shares.
Reporting on page 41	Describes the reports available for suppliers in ESCM.
Integrating External Process Control on page 42	Describes the tasks involved in connecting ESCM with an external process control system.
Menu Options and User Roles	Gives an overview of the menu options relevant for suppliers.
LDAP Keys on page 44	Lists the keys to be defined for enabling access to a customer's LDAP system.
Customer Billing Data on page 46	Describes the elements of an XML file created by exporting customer billing data.

Readers of this Manual

This manual is directed to people who want to offer services to customers for subscription and use. It assumes that you are familiar with the ESCM concepts as explained in the *Overview* manual.

Notational Conventions

This manual uses the following notational conventions:

Add	Names of graphical user interface elements.

`init`	System names, for example command names and text that is entered from the keyboard.
`<variable>`	Variables for which values must be entered.
`[option]`	Optional items, for example optional command parameters.
`one | two`	Alternative entries.
`{one | two}`	Mandatory entries with alternatives.

Abbreviations

This manual uses the following abbreviations:

ESCM	Enterprise Service Catalog Manager
IaaS	Infrastructure as a Service
LDAP	Lightweight Directory Access Protocol
PaaS	Platform as a Service
SaaS	Software as a Service

Available Documentation

The following documentation on ESCM is available:

- *Overview:* A PDF manual introducing ESCM. It is written for everybody interested in ESCM and does not require any special knowledge.
- *Operator's Guide:* A PDF manual for operators describing how to administrate and maintain ESCM.
- *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 ESCM.
- *Supplier's Guide:* A PDF manual for suppliers describing how to define and manage service offerings for applications that have been integrated with ESCM.
- *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 ESCM.
- *OpenStack Integration:* A PDF manual for operators describing how to offer and use virtual systems controlled by OpenStack through services in ESCM.
- *Amazon Web Services Integration:* A PDF manual for operators describing how to offer and use virtual servers controlled by the Amazon Elastic Compute Cloud Web service through services in ESCM.
- *Online Help:* Online help pages describing how to work with the administration portal of ESCM. The online help is intended for and available to everybody working with the administration portal.

1 Introduction

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

With its components, ESCM supports software vendors as well as their customers in leveraging the advantages of Cloud Computing.

The basic scenario of deploying and using applications as services in the ESCM framework involves the following organizations:

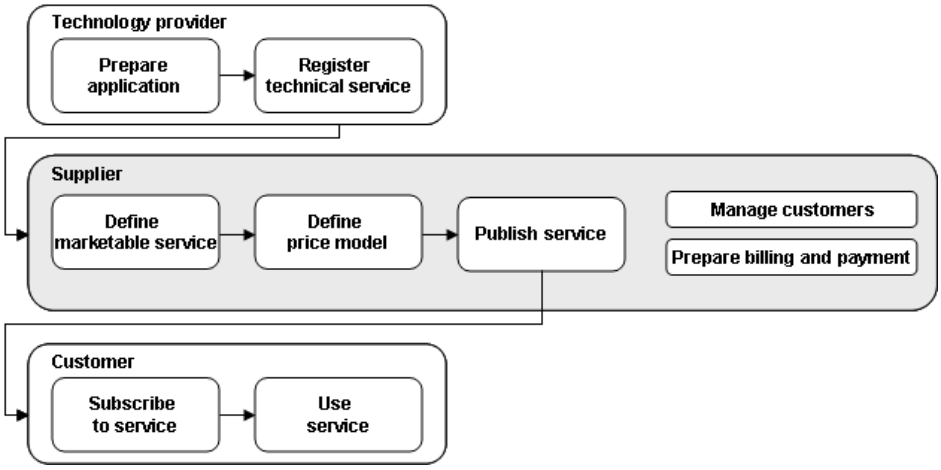
- **Technology providers** (e.g. independent software vendors) technically prepare their applications for usage in the Cloud and integrate them with ESCM. They register the applications as technical services in ESCM.
- **Suppliers** (e.g. independent software vendors or sales organizations) define service offerings, so-called marketable services, for the technical services in ESCM. They publish the services to a marketplace.
- **Customers** register themselves or are registered by an authorized organization in ESCM and subscribe to one or more services. Users appointed by the customers work with the underlying applications under the conditions of the corresponding subscriptions.
- **Marketplace owners** are responsible for administrating and customizing the marketplaces to which services are published.
- **Operators** are responsible for deploying and maintaining ESCM.

  ESCM is provided in Docker containers and deployed in a container environment such as OpenStack or a Kubernetes cluster providing for optimum performance, scalability, failover, and non-stop operation. The applications integrated with ESCM and their data may be hosted in the same Virtual Machine (VM) as ESCM or in different locations.

In extended usage scenarios, the suppliers who define marketable services may involve additional users and organizations in offering and selling these services:

- **Brokers** support suppliers in establishing relationships to customers by offering the suppliers' services on a marketplace. A service subscription is a contract between the customer and the supplier.
- **Resellers** offer services defined by suppliers to customers applying their own terms and conditions. A service subscription establishes a contract between the customer and the reseller.

1.1 The Supplier's Tasks in ESCM



As a supplier, you define marketable services for applications that have been prepared and registered as technical services in ESCM by technology providers.

For each technical service, you can create any number of marketable services, applying different price models, configurations, and restrictions. A price model defines how much a customer will be charged for using a service.

Once you have defined a marketable service and its associated price model, you can make the service available to customers by publishing it to a marketplace. Instead of or in addition to offering services yourself, you can appoint brokers or resellers to do so on the same or different marketplaces.

For subscribing to services you offer and using the underlying applications in the Cloud, customers need to register themselves with ESCM, or you can register them. The customers can subscribe to any number of your services on the basis of the terms and conditions and the price models you have defined. They can register or import any number of users who work with the services.

Apart from providing marketable services with their price models, your role as a supplier involves:

- Managing your customers. This includes, for example, explicitly terminating a customer's subscription.
- Preparing billing and payment. As users work with a service, the usage charges are calculated based on the fees defined in the related price model. This billing data can be used for generating invoices. You are responsible for the processing of the billing data and payment collection.
- Handling revenue shares. The platform operator, the owners of the marketplaces where you publish your services as well as the brokers and resellers who support you are usually entitled to certain shares of your revenues. The revenue shares are defined by the operator. They may differ for individual marketplaces, brokers or resellers, or even specific services. You can at any time retrieve billing data and information on the actual revenues depending on the usage of the services. This data can be used as the basis for collecting or paying the shares, depending on your contracts with the participating parties.

ESCM offers comprehensive reports for different purposes and at different levels of detail. Supplier reports focus on retrieving business data that enable you, for example, to recognize specific patterns of customer behavior.

As a supplier, you can carry out specific actions under the control of an external process control system. ESCM provides a way to interact with such an external process control system by means of triggers and dedicated Web services.

If required, you can also act as a customer in ESCM. As a supplier, you are automatically privileged to subscribe to services and work with the services you have subscribed to.

If you need to perform technology provider or marketplace owner tasks, your organization must be assigned the corresponding roles by the operator. Note that a supplier organization cannot be assigned the reseller or broker role in addition.

1.2 Accessing ESCM

To perform your tasks, you use the ESCM user interface. The role of your organization as a supplier and your user role within the organization determine which features are available to you at the ESCM user interface.

ESCM distinguishes between the following user roles within supplier organizations:

- **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.
- **Service manager**: This role allows a user to define marketable services and price models as well as publish marketable services.

To access the ESCM user interface, you use the login information provided by ESCM in the email confirming the creation of your user account. If your organization uses an external authentication system, passwords are managed in this system. This means that you log in with the password as stored in this system, and the email sent by ESCM does not contain a password.

To log in to the administration portal where you will perform most of your tasks:

1. Click the link provided in the email, or type the access URL in your Web browser's address bar.

   The access URL has the following format:

   `https://<hostname.fqdn>:<port>/oscm-portal`

   `<hostname.fqdn>` is the name and the fully qualified domain name of the machine where ESCM has been deployed. `<port>` is the port to address the machine (default `8081`), `oscm-portal` is the default context root of ESCM and cannot be changed.
2. On the **Login** page, type your user ID and password.
3. Click **Login**, or press **Return**.

You are either logged in directly, or you are prompted to change your initial password when you log in for the first time. It is highly recommended to change the initial password.

If you try to log in with a wrong password, your account may be locked after the third attempt. This depends on whether your organization maintains its user data in an external authentication system. In this case, passwords can only be changed or reset in this system. If user data are maintained in the platform, contact your administrator who can reset your password. You will get a new temporary password for your next login.

If you have forgotten your password, click **Forgot your password?** on the **Login** page. This allows you to define a new password for your user ID. Defining a new password is not possible if your account is locked or if your organization maintains its user data in an external authentication system.

If you have forgotten your user ID, contact your administrator who can look up the IDs of all users registered for your organization.

If your session expires, you have to log in again.

2 Marketable Services

A marketable service is a service offering to customers based on a technical service provided by a technology provider. It specifies parameters, conditions, and restrictions for using the underlying application.

You can define marketable services for all the technical services for which you have been granted permissions by the relevant technology providers. For each technical service, any number of marketable services can be defined, applying different price models, configurations, and upgrade and downgrade options.

You could, for example, define three marketable services based on one technical service, thus offering three different editions of an application to your customers. The services could be defined as follows:

	Configuration	**Price**
Trial edition	Limited subscription for 4 weeks Limited set of features Number of users limited to 1	Free of charge
Standard edition	No limit of subscription lifetime Standard set of features Number of users limited to 25	Basic charge of 45 € per month
Enterprise edition	No limit of subscription lifetime Full set of features Unlimited number of users	Basic charge of 60 € per month 1 € per user login

If defined for the underlying application, specific parameters can take on different values which you can make available to customers as different options. When a customer subscribes to the service, he can choose between the options. This makes service definitions flexible, since there is no need for defining separate services for every option you want to offer.

For example, you could offer a parameter defining the required disk space with three options for the customer to choose from:

- Minimum space of 100 GB
- Medium space of 200 GB
- Maximum space of 400 GB

You can define different prices for the three options. For details, refer to *Price Model Elements* on page 21.

In the following sections, you will find an overview of the elements that make up the definition of a marketable service and the functions that support you in defining, maintaining, and publishing marketable services. In addition, the sections describe how you can use the support of brokers and resellers.

2.1 Defining Marketable Services

The general procedure for defining a marketable service in ESCM is the following:

1. Create the marketable service. For details, refer to *Creating Marketable Services* on page 12.

2. Define a price model for the service. For details, refer to *Price Models* on page 21.
3. Define upgrade and downgrade options (optional). For details, refer to *Defining Upgrade and Downgrade Options* on page 13.

2.1.1 Creating Marketable Services

Creating a marketable service means specifying its parameters, conditions, and restrictions.

To create a marketable service, use the **Define a service** menu option in the **Marketable service** menu of the ESCM administration portal.

When you create a marketable service, you first have to select the technical service the marketable service is to be based on. You can then specify the following elements:

- A **service identifier**. This is offered for selection when you update or delete the service, or when you specify upgrade and downgrade options or a price model for it.

  The service identifier is mandatory. It must be unique within the marketable services of your organization.

- An **image** representing the service. This may be useful, for example, for customers to identify the service.

  The image is shown in the service catalog on the marketplace to which you publish the service.

- A **service name for customers**. The name should be meaningful and should have a high recognition value.

  The service name for customers is shown in the service catalog on the marketplace to which you publish the service. If you do not specify it, a string is displayed saying that it is not defined.

- A **short description** that briefly introduces the service. It may, for example, provide a short overview of the features the service provides.

  The short description is shown in the service catalog on the marketplace to which you publish the service.

- A **service description** that provides, for example, details of the features offered by the service.

  The underlying technical service comes with a description that is displayed by default for the marketable service. It is up to you whether you use the description as defined by the technical service, or whether you enter a different one.

  The service description is shown to customers on a marketplace when they request details of a service.

- **Service parameters** related to subscription and usage.

  Subscription parameters include, for example, the maximum number of users per subscription or the maximum lifetime of a subscription. It depends on the underlying technical service which subscription parameters are available.

  Usage parameters include, for example, the enablement or disablement of specific features of the underlying application, or the maximum number of folders, files, or objects that can be created. The usage parameters are defined and provided by the underlying technical service.

- An external **parameter configuration tool** that your customers can use for configuring the service parameters and for viewing their prices per subscription when they subscribe to the service on a marketplace.

  You can integrate a third-party or proprietary parameter configuration tool of your choice with ESCM. During the subscription process, the tool replaces the default user interface for

configuring service parameters. To link the tool with ESCM, you have to specify the URL of the tool in the service definition.

ESCM provides a generic interface by which you can integrate the tool.

- A **custom tab** for integrating a Web page or Web application for providing detailed information on a service and its operation to your customers. Customers may open the custom tab when they display details on a subscription on a marketplace. Particularly users of IaaS services are interested in additional details on the service and the operation of the service they subscribed to. Using a custom tab, you can provide the information needed for accessing provisioned instances, such as IP addresses, network information, or user credentials.

  To link the Web page or Web application with ESCM, you have to specify its URL in the service definition.

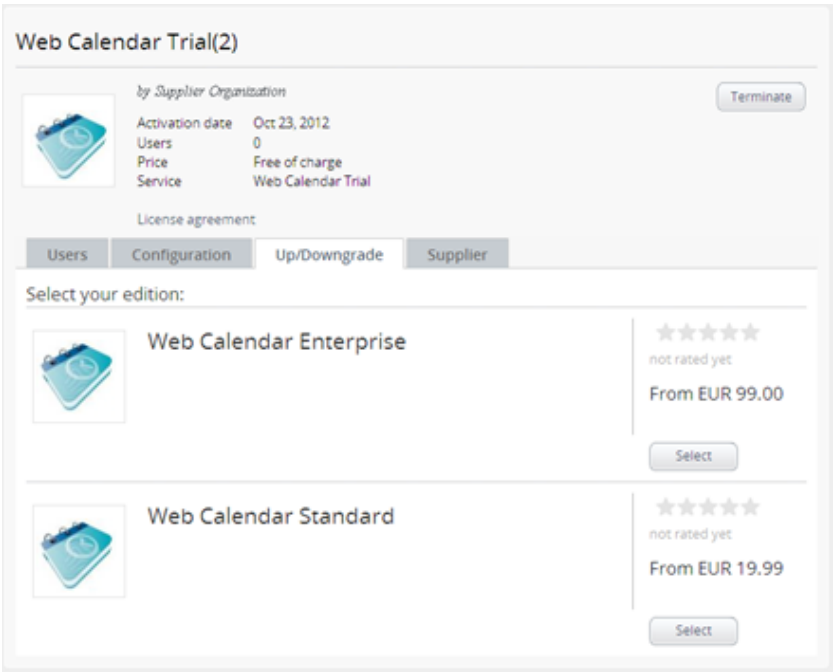
Additionally, you can specify whether a user subscribing to the service is automatically assigned to the corresponding subscription.

The service name for customers, the short description, the description, and the name of the custom tab can be specified in one or several languages. For details, refer to *Maintaining Marketable Services* on page 18.

2.1.2 Defining Upgrade and Downgrade Options

Upgrade and downgrade options allow customers to migrate between marketable services. They enable you to flexibly meet the changing needs of your customers. A subscription for a marketable service defining a trial edition could be upgraded, for example, to a marketable service defining a standard or enterprise edition. A subscription for a marketable service with an unlimited number of users could be downgraded to a marketable service with a limited number of users.

The following example shows upgrade options for a service on a marketplace:



You can specify any number of upgrade and downgrade options for a marketable service. The options must all be based on the same technical service and be published to the same marketplace. If customers are charged for using the marketable service, the price models of the upgrade and downgrade options must use the same currency.

When a customer upgrades or downgrades a subscription, the price model of the new service is instantly applied for calculating the charges. Users assigned to the subscription, service roles assigned to the users, or application data already stored for the old service are retained. All parameters and their options as well as the payment information are automatically applied to the new service, unless explicitly changed during the upgrade or downgrade.

Be aware of the effects of up/downgrade options on the price model elements that determine how the charges for a customer are calculated(see *Price Model Elements* on page 21 for details):

- The prices for using a service may be calculated and charged based on time units. When a customer upgrades or downgrades a subscription with a price calculation based on time units, the costs for the unit in which the subscription is upgraded or downgraded are charged twice.
- A one-time fee that is defined for a service you provide as an upgrade or downgrade option is added to the total charges for the customer, even if the service from which the customer migrates also defines a one-time fee.
- A free trial period that is defined for a service you provide as an upgrade or downgrade option starts immediately after the customer's upgrade or downgrade. It is independent of whether the service from which the customer migrates also defines a free trial period.

To define upgrade and downgrade options, use the **Define up/downgrade options** menu option in the **Marketable service** menu of the ESCM administration portal.

For defining upgrade and downgrade options, the relevant marketable services must be deactivated. As soon as you have published the services again, your customers can subscribe to them and are free to upgrade and downgrade their subscriptions as required.

2.2 Publishing Marketable Services

Publishing a marketable service means making it available to customers for subscription on a marketplace of your choice.

The following prerequisites must be fulfilled for publishing a service:

- A price model must have been defined for the service. For details, refer to *Price Models* on page 21.
- Access to the marketplace and the permission to publish services on it must have been granted by the marketplace owner.

Publishing a service involves selecting the marketplace, activating the service, and publishing the URL of the marketplace. These tasks are described in more detail in the sections below.

Specifying a Marketplace

As a prerequisite for publishing a service, you need to specify the desired marketplace.

A service can be published to exactly one marketplace. If you want to publish it to a different marketplace, you have to deactivate it before you can select the new marketplace.

Specifying a marketplace includes:

- Deciding whether the service is to be public.

A public service is visible in the service catalog for anybody accessing the marketplace. A service that has not been marked as public is available for subscription to registered customers only.

- Selecting the marketplace.

  You can publish the service to a marketplace that is open to any seller, or to a marketplace that is not open but for which you have been admitted as a seller by the marketplace owner. The marketplace must be public or you need to be granted access by the marketplace owner.

- Assigning categories.

  Provided that categories are defined for the marketplace you select, you can assign one or multiple categories to your service. The marketplace owner is responsible for creating the categories you can choose from. Customers can use the categories for browsing the service catalog and filtering the services on the marketplace. The marketplace owner may also decide to display the services on the marketplace home page grouped by categories.

To specify the marketplace, use the **Define publishing options** menu option in the **Marketable service** menu of the ESCM administration portal.

Activating a Service

To make a service available for subscription, you must activate it. To do this, use the **Activate or deactivate services** menu option in the **Marketable service** menu of the ESCM administration portal.

The menu option can also be used for deactivating a service. This is required if you do not want to offer it any longer or if you want to change the marketplace. If you deactivate a service, customers can no longer subscribe to it. Existing subscriptions are not affected.

Publishing the URL of a Marketplace

To make your service offering known to customers, you publish the URL of the marketplace. You can do this, for example, by email or on your website or portal. You can obtain the URL from the marketplace owner, who receives it from ESCM via email when the marketplace is created.

The URL has the following format:

`https://<hostname.fqdn>:<port>/oscm-portal?marketplaceId=<mId>`

`<hostname.fqdn>` is the name and the fully qualified domain name of the machine where ESCM has been deployed. `<port>` is the port to address the machine (default `8081`), `oscm-portal` is the default context root of ESCM and cannot be changed. `<marketplaceId>` is the ID of the marketplace.

People who use the URL for accessing a marketplace can view all services that are marked as public. When subscribing to a service, they are either redirected to the registration Web page or they can log in to ESCM directly, provided they are already registered as a customer.

You can also publish the URL of the registration Web page of ESCM. This allows customers to directly register with ESCM. Registered customers can see all services, no matter whether they are marked as public or not. A registered customer is sent an email with the login data and an initial password to be used for the first login.

The URL of the registration Web page has the following format:

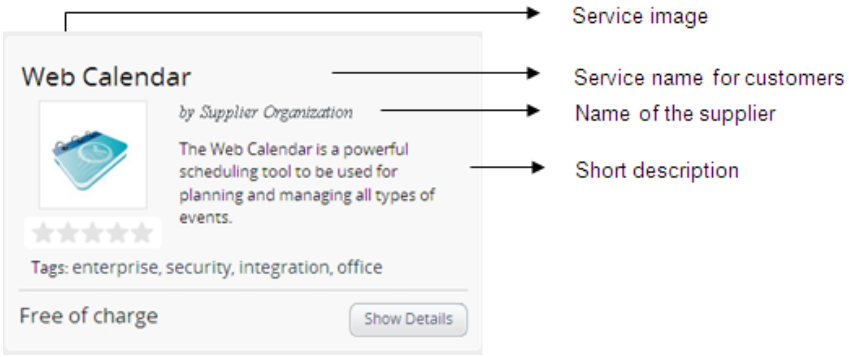
`https://<hostname.fqdn>:<port>/oscm-portal/marketplace/registration.jsf?marketplaceId=<mId>`

Note:	The marketplace owner may restrict the access to a marketplace to the users of specific organizations. In this case, the users need to log in to be able to see, subscribe to, and use the services published on the marketplace. New customers cannot register themselves on the marketplace but need to be registered by a supplier, broker, reseller, or the platform operator.

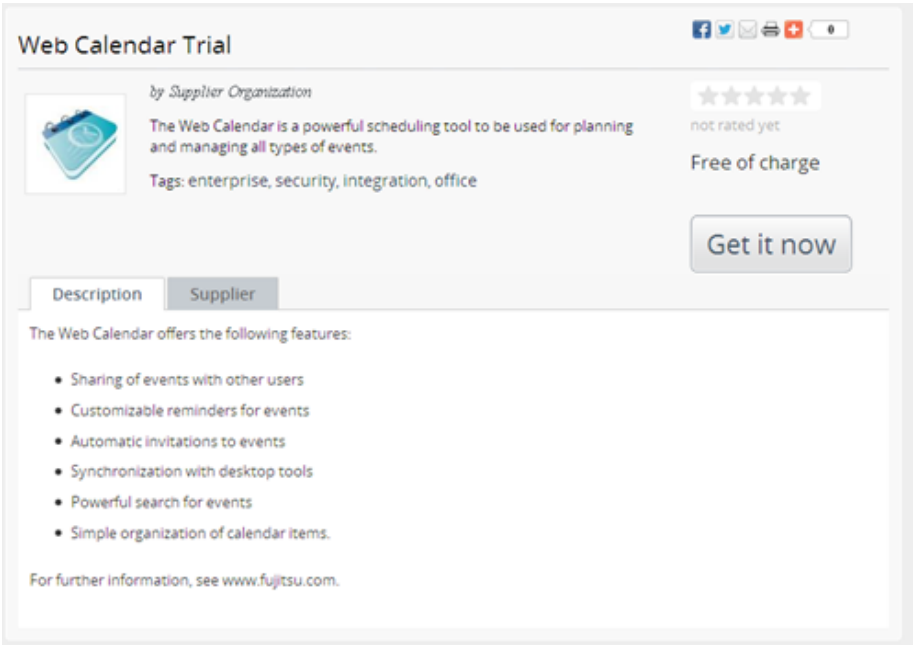
Example

The following example shows how a published service is presented on a marketplace.

In the service catalog on the marketplace, the service image, the service name for customers, the short description, and the name of the supplier are displayed:



From the service catalog, a customer can request details of the service. This allows the customer to view the service description you have provided:



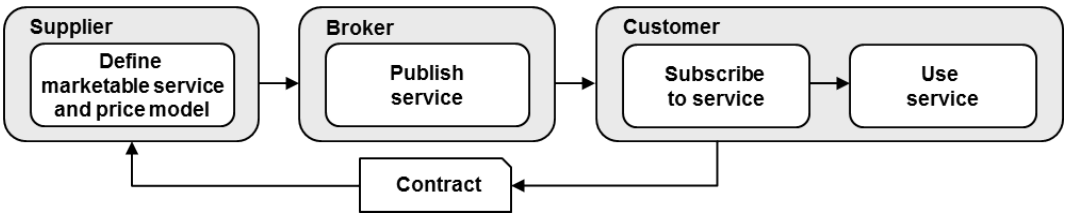
2.3 Working with Brokers and Resellers

As a supplier, you can involve brokers and resellers in offering and selling your services. This requires contractual relationships between you and your partners. The operator is responsible for creating the broker and reseller organizations in ESCM. He also needs to have a contractual relationship with them.

The following sections describe the different usage scenarios and your tasks as a supplier within these scenarios.

Broker Usage Scenario

A broker can support you in promoting a marketable service you have defined. To this end, the broker publishes the service to a marketplace. A subscription to a service establishes a contract between you and the customer.



For offering a service on a marketplace, the broker needs your explicit permission. You can grant this permission if you have fulfilled the following prerequisites:

- The service must be defined, and its parameters, conditions, and restrictions must be specified. For details, refer to *Creating Marketable Services* on page 12.
- The price model, including a license agreement, must be defined for the service. For details, refer to *Price Models* on page 21.

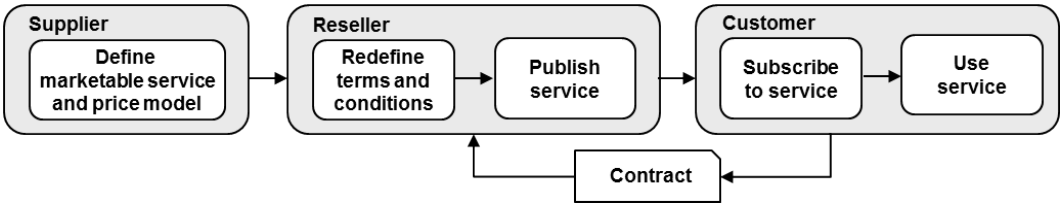
The broker cannot change the service definition, the price model, and the license agreement. He can view these elements and publish the service to a marketplace of his choice.

As a supplier, you are responsible for preparing billing and payment for your broker's customers and collecting the payments. For details, refer to *Billing and Payment* on page 37.

A broker usually receives a share of the revenue for the services he offers. ESCM allows a broker to retrieve information on his actual revenue. As a supplier, you can also retrieve information on the revenue accumulated by your brokers. For details, refer to *Billing and Payment* on page 37.

Reseller Usage Scenario

A reseller can offer and sell a service you have defined, applying his own terms and conditions. A subscription to the service establishes a contract between the reseller and the customer.



For offering a service on a marketplace, the reseller needs your explicit permission. You can grant this permission if you have fulfilled the following prerequisites:

- The service must be defined and its parameters, conditions, and restrictions must be specified. For details, refer to *Creating Marketable Services* on page 12.
- The price model must be defined for the service. For details, refer to *Price Models* on page 21.

The reseller can view the service definition and the price model, but he cannot change them. The contract between him and the customer is based on your prices. The reseller can, however, define his own license agreement. It is up to him whether he uses your license agreement or enters a different one.

A reseller who has been granted your permission can publish the service to a marketplace of his choice.

Since a subscription establishes a contract between the reseller and the customer, you are not involved in billing and payment. The reseller is responsible for preparing billing and payment for his customers and collecting the payments.

A reseller is usually not entitled to keep the entire revenue collected from his customers, but only a certain share thereof. ESCM allows a reseller to retrieve information on his actual revenue. As a supplier, you can also retrieve information on the revenue accumulated by your resellers. For details, refer to *Reporting* on page 41.

Granting Brokers and Resellers the Resale Permission

To grant brokers and resellers the permission to sell a service, use the **Define publishing options** menu option in the **Marketable service** menu of the ESCM administration portal. Your resale permission is not bound to an individual marketplace. The brokers and resellers can choose any marketplace to which they have been granted access by the marketplace owner.

The menu option can also be used for revoking the resale permission. This may be required, for example, if a reseller is in arrears with his payments to the other participating parties. If you revoke the permission, a service is instantly withdrawn from the marketplace to which it was published. Customers can no longer subscribe to it. Yet, existing subscriptions of the broker's or reseller's customers are not affected.

2.4 Maintaining Marketable Services

ESCM provides various features and functions for maintaining marketable services. You can:

- Deactivate a published service.
- Update a service.
- Localize elements of a service definition.
- Copy a service.
- Delete a service.

Deactivating a Marketable Service

You can deactivate a published service if you want to withdraw it from a marketplace or if you want to update it. A deactivated service is no longer displayed in the service catalog. Thus, customers cannot subscribe to it any longer. Existing subscriptions are not affected.

You can only deactivate services you have published. You cannot deactivate services published by your brokers or resellers.

To deactivate a marketable service, use the **Activate or deactivate services** menu option in the **Marketable service** menu of the ESCM administration portal.

Note: A marketplace manager of the marketplace owner organization is also allowed to deactivate a service you have published, for example, if you are in arrears with your payments. An email from the marketplace manager informs you about the reason for a deactivation. As soon as the problem is solved, the marketplace manager can reactivate the service again.

Updating a Marketable Service

You can update all elements of a marketable service's definition except for the technical service.

For updating a marketable service, it must be deactivated. Existing subscriptions are not affected by the update. Any changes you make are effective for new subscriptions only.

To update a service, use the **Update a service** menu option in the **Marketable service** menu of the ESCM administration portal.

If you update a service for which you have granted resale permissions, be aware of the fact that the brokers and resellers as well as their customers can instantly view any changes you make. Activated services are immediately offered with the updated definition. Existing subscriptions to services are not affected by your changes. The changes are effective for new subscriptions only.

Localizing Elements of a Service Definition

You can localize the following elements of a marketable service's definition to make them available in all languages supported by ESCM:

- Service name for customers
- Short description
- Description

For localizing the elements, the marketable service must be deactivated. Existing subscriptions are not affected by the localization. Any changes you make are effective for new subscriptions only.

To localize the elements, use the **Update a service** menu option in the **Marketable service** menu of the ESCM administration portal, and click **Localize** on the **Update a service** page that is displayed.

If you localize the elements for a service for which you have granted resale permissions, be aware of the fact that the brokers and resellers as well as their customers can immediately view the localized elements.

Copying a Marketable Service

Copying a marketable service may be useful for offering an existing service with slightly different properties, for example, a higher service level with a higher price, or special discounts for a promotion campaign.

Copying a marketable service means:

- A new marketable service is created that is based on the same technical service as the copied service.
- The service definition is copied to the new service.
- The price model is copied to the new service.

To copy a service, use the **Copy a service** menu option in the **Marketable service** menu of the ESCM administration portal. Once you have created the copy, you can proceed with updating it as required.

Deleting a Marketable Service

You can delete a marketable service which you do not want to offer to customers any longer.

For deleting a marketable service, it must be deactivated and all resale permissions must be revoked. Existing subscriptions are not affected by the deletion. Your customers and your partners' customers can continue to use their subscriptions.

To delete a service, use the **Delete a service** menu option in the **Marketable service** menu of the ESCM administration portal.

3 Price Models

A price model specifies whether and how much customers subscribing to a service will be charged for:

- Using the subscription as such.
- Each user assigned to the subscription.
- Individual events. These include, for example, login and logout by users to the underlying application through ESCM, the completion of specific transactions, or the creation or deletion of specific data.
- Parameters and their options.

It depends on the implementation of the underlying application and on its integration with ESCM whether fees for events or parameters can be defined.

In the following sections, you will find detailed information on the price model scope and the price model elements. In addition, you will find an overview of the functions that support you in defining and maintaining price models.

3.1 Price Model Scope

Price models can have different scopes.

Service price models are the basic price models. A service price model must be defined for every marketable service. Without defining a service price model, you cannot publish a service. You can define exactly one service price model for each service.

The service price model defines the prices to which brokers and resellers are bound if they offer and sell a service.

In addition to service price models, ESCM allows you to define **customer price models**. A customer price model can be used, for example, to offer lower charges to a long-time customer, or to make a new service available to a pilot customer at no charge.

A customer price model is specific to one customer and to one service that is used by the customer. It is not available for any other services offered to the same or a different customer. When you define a customer price model, ESCM automatically creates a marketable service for this specific customer only.

For even greater flexibility, you can define **subscription price models**. A subscription price model is specific to one subscription of an individual customer. It is not available for any other subscriptions of the same or a different customer.

3.2 Price Model Elements

Price models consist of the following elements that determine how the charges are calculated:

- One-time fee
- Recurring charge per subscription
- Recurring charge for users
- Prices for parameters and options
- Prices for events
- Role-based pricing

Stepped prices are supported which allow the definition of ranges for which different price model elements apply. In addition, a free trial period can be defined and discounts can be granted to individual customers.

The calculation of the actual charges for a service varies depending on the combination of the price model elements that are defined. You can find a detailed description of each price model element together with an example for the cost calculation in the subsequent sections.

Cost Calculation Options

The charges for a service can be calculated pro rata or per time unit. This affects the recurring charges for subscriptions and users as well as the charges for parameters, options, and service roles. The one-time fee and event prices are not affected.

- The **pro rata calculation** is based on milliseconds.

  Customers are charged exactly for the time a service was used.

  Switches to daylight saving time are considered in the cost calculation: When daylight saving time begins, the respective day has 23 hours. The related week and month time units are also shortened by one hour. When daylight saving time ends, the respective day has 25 hours. The related week and month time units are also extended by one hour.

- The **per time unit calculation** is based on fixed time units (month, week, day, or hour).

  Customers are charged for the time units during which they use a service. The time units are always fully charged, even if they are not fully used. The costs for a time unit are calculated for the billing period in which the unit ends.

  The time unit can be defined as **Hour**, **Day**, **Week**, or **Month**. The start and end of each unit are pre-defined as follows:

  Hour: From 00:00.000 to 59:59.999.

  Day: From 00:00:00.000 to 23:59:59.999.

  Week: From Monday, 00:00:00.000, to Sunday, 23:59:59.999.

  Month: From the first day of the month, 00:00:00.000, to the last day of the month, 23:59:59.999.

- In service price models and customer-specific price models, you can define and change the cost calculation option. When updating a price model, you can switch between the pro rata and the per time unit calculation. The prices are kept in this case. The prices get lost when a price model is changed to free of charge.

  The cost calculation options **cannot** be changed for subscription price models.

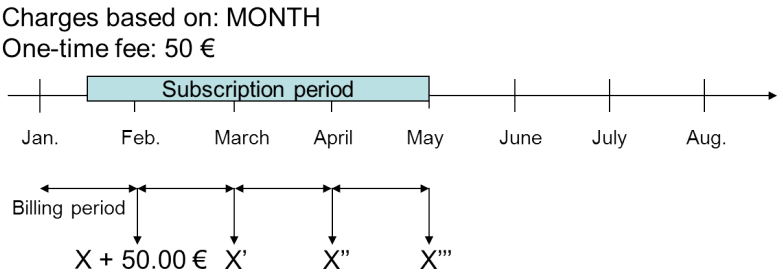
3.2.1 One-Time Fee

A one-time fee defines an amount a customer has to pay for a subscription in the first billing period.

A one-time fee is added to the total charges for the first billing period. It is independent of the number of users, events, or other usage data.

If a one-time fee is defined for a service to which a customer upgrades or downgrades a subscription, it is added to the total charges for the customer, even if the service from which the customer migrates also defines a one-time fee.

Example



In the example above, 50.00 € are added to the charges of the first billing period.

3.2.2 Recurring Charge per Subscription

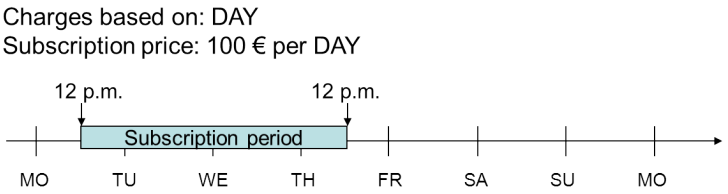
For each subscription, a charge can be defined that a customer has to pay on a recurring basis. Monthly, weekly, daily, or hourly periods are supported. The recurring charge for a subscription is independent of the amount of users, events, or other usage data.

Example

A customer subscribes to a service on Monday, 12 o'clock p.m. The subscription ends on Thursday, 12 o'clock p.m.

The price model specifies a recurring charge per subscription of 100.00 € per DAY.

The recurring charge for the subscription is calculated as follows:



Pro rata calculation: 0.5 + 1.0 + 1.0 + 0.5 = 3 * 100.00 € = 300.00 €

Per time unit calculation: 1.0 + 1.0 + 1.0 + 1.0 = 4 * 100.00 € = 400.00 €

With pro rata calculation, the customer is charged with 300 €. With per time unit calculation, the costs are 400 € because each day is fully charged.

3.2.3 Recurring Charge for Users

For the users assigned to a subscription, a charge can be defined that a customer has to pay on a recurring basis. Monthly, weekly, daily, or hourly periods are supported. The charge depends on the amount of time units one or more users are assigned to the subscription. This type of charge can only be defined for services with the login or user access type.

The recurring charge for users is independent of the recurring charge per subscription or other usage data.

For this type of charge, stepped prices can be applied: Recurring charges can be defined that depend on the sum of the time units of all user assignments.

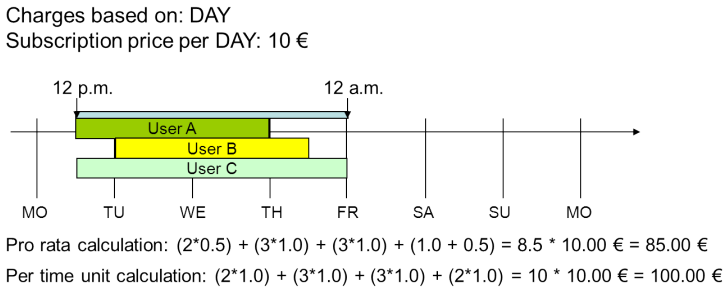
Refer to *Stepped Prices* on page 28 for details.

With per time unit calculation, the costs for a time unit in which a user is assigned to a subscription are always fully charged. There is no difference in the costs between a user who is assigned from the start until the end of the time unit and a user who is assigned for a part of the time unit only. A time unit is charged only once if a user is deassigned from and re-assigned to a subscription within the same time unit. Yet, canceling an assignment, deleting the user, and then creating a new user with the same user ID is treated as if two different users are assigned to the subscription. The time unit is charged twice, accordingly.

Example

3 users are assigned to a subscription. The price model specifies a recurring charge for users of 10.00 € per DAY. User A and User B use the service for 2.5 days, User C for 3.5 days.

The charges are calculated as follows:



With pro rata calculation, the customer is charged 85 €. With per time unit calculation, the costs are 100 € because each day is fully charged.

Example: Combination of Price Model Elements

Below you find a price model that combines the elements "one-time fee", "recurring charge per subscription", and "recurring charge for users":

- Charges based on: MONTH
- One-time fee: 30.00 €
- Recurring charge per subscription: 10.00 € per MONTH
- Recurring charge for users: 20.00 € per MONTH

The price model is applied to a service subscription that has 5 users assigned. 2 of them are assigned for half a month only. The other 3 users are assigned for the whole month.

With pro rata calculation, the costs for the first billing period of one month are calculated as follows:

Pro rata calculation

Usage time:	2 users * 0.5 + 3 users * 1 = **4**	
Recurring charge for users:	Charge 20 € * 4	80.00 €
One-time fee:	30.00 €	30.00 €
Recurring charge per subscription:	10.00 €	10.00 €
TOTAL for first month:		**120.00 €**

With per time unit calculation, the costs for the first billing period of one month are calculated as follows:

Per time unit calculation

Time units:	5 users * 1 = 5	
Recurring charge for users:	Charge 20 € * 5	100.00 €
One-time fee:	30.00 €	30.00 €
Recurring charge per subscription:	10.00 €	10.00 €
TOTAL for first month:		**140.00 €**

For subsequent billing periods, the charges are calculated without the one-time fee.

3.2.4 Prices for Parameters and Options

A price model can define prices for service parameters and options. It depends on the implementation and integration of the underlying application whether and which parameters and options are available.

A supplier can define a price for every parameter and option, and specify whether this price is to be charged per subscription or per user assigned to the subscription. Numeric parameters are a multiplier for the price. For boolean parameters, the multiplier is `1` if the value is `true`. In all other cases, the multiplier is `0`.

If the charges for a subscription are calculated per time unit and a customer changes a parameter value within a time unit, the affected time unit is charged pro rata. This means that the customer is charged exactly for the time each parameter value is set.

For numeric parameters, stepped prices can be applied per subscription: Different prices can be defined depending on the parameter values.

For details, refer to *Stepped Prices* on page 28.

The prices for parameters and options are independent of other price model elements.

Example

In the following example, the price for the parameter "number of folders that can be created" is set to 4.00 € per folder per subscription. The renaming of folders costs 1.00 € per day for each user assigned to the subscription. It is assumed that 2 users are assigned to the subscription for one day, which corresponds to the subscription period in this example. The customer chose to restrict the number of new folders to 45 when subscribing to the service.

The costs are calculated as follows with both pro rata and per time unit calculation:

(45 * 4.00 €) + (2 * 1.00 €) = 182.00 €

Total amount: 182.00 €

If the users are not assigned for the entire subscription period, but for a few hours only, the costs are calculated as follows:

Assume that user A is assigned for 2 hours, user B for 4 hours. This is a total of 6 hours, i.e. 25 % of the entire subscription period (DAY). This results in the following costs with pro rata calculation:

(45 * 4.00 €) + (**0.25** * 1.00€) = 180.25 €

Total amount: 180.25 €

With per time unit calculation, the costs are again **182.00 €** because the day is fully charged.

3.2.5 Prices for Events

A price model can define prices for individual, billable events. These include, for example, login and logout by users to the underlying application, the completion of specific transactions, or the creation or deletion of specific data. It depends on the implementation and integration of the underlying application which events are available.

For this type of charge, stepped prices can be applied: Different prices for an event can be defined depending on the number of occurrences per billing period.

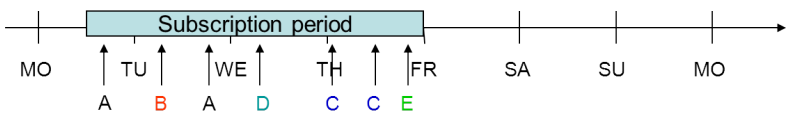
For details, refer to *Stepped Prices* on page 28.

The costs for events are independent of the selected calculation option (pro rata, per time unit).

Example

The following prices are defined for five types of event:

	Description	Price
A	Login of a user to the service.	1.00
B	Logout of a user from the service.	0.50
C	File download	1.50
D	File upload	1.00
E	New folder	0.50



The charges for the events are calculated based on their occurrence. The subscription period is within one billing period.

Event A:	2 * 1.00 €	2.00 €
Event B:	1 * 0.50 €	0.50 €
Event C:	2 * 1.50 €	3.00 €
Event D:	1 * 1.00 €	1.00 €
Event E:	1 * 0.50 €	0.50 €
TOTAL for subscription period:		**7.00 €**

3.2.6 Role-Based Prices

If defined for the underlying application, roles can be used to grant specific privileges to different users. The roles are specified in the technical service definition as service roles. Service roles can be mapped to corresponding permissions in the application.

Service roles can be used for role-specific prices: For each role, a price can be defined. This price is added to the base price per user in the cost calculation for a billing period.

Regarding the cost calculation options, role-based prices are handled the same way as the recurring charges for users. With per time unit calculation, the costs for a time unit in which a user with a specific role is assigned to a subscription are always fully charged. There is no difference in costs between a user who is assigned from the start until the end of the time unit and a user who is assigned for a part of the time unit only.

If the charges are calculated per time unit and the role assignment of a user is changed within a time unit, the affected time unit is charged pro rata. This means that the customer is charged exactly for the time each user role is assigned.

If the charges are calculated per time unit and a user with a specific role is removed from the subscription and assigned to it again with a different role in the same time unit, the customer is also charged for the time during which the user is not assigned to the subscription. This means that he is charged with the price for the first service role until the user is assigned to the subscription with the second service role.

Example

The sample calculation below uses the following data:

Role-specific prices:

User role	Description		Price
Administrator			2.00
User			3.00
Guest			5.00

The customer has 100 users assigned to the subscription. All of them are assigned for an entire billing period. 5 users have the Administrator service role, 80 the User service role, and 15 the Guest service role for the subscription.

The costs are calculated as follows:

Role-specific charges: (5 * 2.00 €) + (80 * 3.00 €) + (15 * 5.00 €) = 325.00 €

Total amount: 325.00 €

3.2.7 Stepped Prices

Stepped prices allow for the definition of ranges for which different price factors apply. Step limits, i.e. the upper limits of ranges, can be set for:

- The **sum of the time units** users are assigned and work with a subscription in a billing period. For example, up to 10 hours one user is assigned to a subscription cost 10.00 € per hour, every additional hour the user is assigned costs 8.00 €.
- **Values of numeric parameters**. For example, uploading up to 100 files costs 1.00 € per file, any additional upload costs 0.50 € per file.
- The **number of events** occurring in the usage of a subscription. For example, up to 10 file downloads cost 1.00 € per download, any additional download costs 0.50 €.

Defining a price model with stepped prices can be useful if you want to provide:

- A price for users depending on the amount of time users are assigned to a subscription.
- A price per event based on the number of events.
- A price for a parameter depending on its value. This is possible for numeric parameters (of type Integer or Long) only.

Stepped prices are independent of any other price model elements.

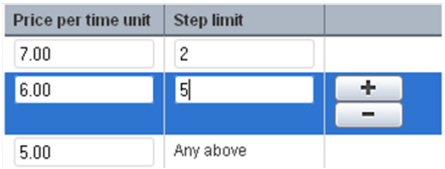
Example: Stepped Prices for Users

In the example below, the price model specifies a recurring charge for users per HOUR. Stepped prices are applied as follows: The first 2 hours a user is assigned to the subscription costs 7.00 € each. The 3rd, 4th, and 5th hour costs 6.00 € each. Every additional hour costs 5.00 €.

For the sample calculation, the customer assigns 4 users to the subscription. Each user uses the subscription for exactly one hour. The calculation would be the same if the customer assigns 1 user and this user uses the subscription for four hours.

The example assumes pro rata calculation.

Definition of stepped prices for users:



For billing, an entire period of one month is assumed. The charge for using the subscription is calculated as follows:

(2 * 7.00 €) + (2 * 6.00 €) = 26.00 €

Total amount: 26.00 €

Calculation with users being assigned for different time spans:

Calculating the costs gets more complex as soon as users are assigned to a subscription for different time spans.

In the example below, the recurring charge for users is again based on HOUR:

- 3 users are assigned to the subscription for 30 minutes (= 1.5 users / HOUR)
- 2 users are assigned to the subscription for 3.5 hours (= 7 users / HOUR)
- 3 users are assigned to the subscription for 2 hours (= 6 users / HOUR)

The sum of 14.5 hours (1.5 + 7 + 6) the users are assigned to the subscription is taken as the basis for the calculation. As defined in the first example, 2 hours cost 7 € each, the next 3 hours cost 6 € each, any above cost 5 €.

The prices are calculated as follows:

(2 * 7.00 €) + (3 * 6.00 €) + (9.5 * 5.00 €) = **79.50 €**

Total amount: 79.50 €

The same example, but with per time unit calculation based on HOUR, results in the following:

- 3 users are assigned to the subscription for 30 minutes (= 3 users / HOUR)
- 2 users are assigned to the subscription for 3.5 hours (= 8 users / HOUR)
- 3 users are assigned to the subscription for 2 hours (= 6 users / HOUR)

The sum of 17 hours (3 + 8 + 6) the users are assigned to the subscription is taken as the basis for the calculation. As defined in the first example, 2 hours cost 7 € each, the next 3 hours cost 6 € each, any above cost 5 €.

The prices are calculated as follows:

(2 * 7.00 €) + (3 * 6.00 €) + (12 * 5.00 €) = **92.00 €**

Total amount: 92.00 €

Example: Stepped Prices Depending on Parameter Values

In the example below, stepped prices are defined for the parameter "Number of folders that can be created". The first 40 folders cost 4.00 € each, the next 10 folders cost 3.50 € each, every folder that is created in addition costs 3.00 €. In the example, the customer chose to restrict the creation of folders to 45 when subscribing to the service.

Definition of stepped prices for a parameter:

Description	Price per user	Price per subscription	Step limit
Number of folders that can be created.	0.00	4.00	40 12-500
		3.50	50
		3.00	Any above

Customer choice when subscribing to the service:

Description	Value [minimum - maximum]	Price per user	Price per subscription	Step limit
Number of folders that can be created.	45 12-500	0.00	4.00	40
			3.50	50
			3.00	Any above

For billing, an entire period of one month is assumed. The charge for creating up to 45 folders is calculated as follows:

(40 * 4.00 €) + (5 * 3.50 €) = 177.50 €

Total amount: 177.50 €

Example: Stepped Prices Depending on the Number of Events

In the example below, different prices for events are defined depending on the number of occurrences per billing period. Stepped prices are defined for the following events:

- Login of a user to the service
- File download
- File upload

Logout from the service and creating new folders is free of charge.

Description	Price	Step limit
Login of a user to the service.	1.00	100
	0.50	200
	0.25	300
	0.20	Any above
Logout of a user from the service.	0.00	
File download	0.25	100
	0.20	Any above
File upload	1.00	100
	0.80	Any above
New folder	0.00	

For billing, an entire period of one month is assumed. The following events occur in the billing period:

- 500 times a user logs in to the service
- 300 files are downloaded
- 200 files are uploaded

The charge for these events is calculated as follows:

Login: (100 * 1.00 €) + (100 * 0.50 €) + (100 * 0.25 €) + (200 * 0.20 €) = 215.00 €

Download: (100 * 0.25 €) + (200 * 0.20 €) = 65.00 €

Upload: (100 * 1.00 €) + (100 * 0.80 €) = 180.00 €

Total amount: 460.00 €

3.2.8 Free Trial Period

A free trial period defines a limited period of time in which a service is offered for free. You can, for example, define a free trial period to give your customers time for testing a service. After this period, the customers are automatically charged according to the price model elements you define.

If a subscription is suspended and resumed during the free trial period, the free trial period is not prolonged although the users cannot work with the underlying service during this time.

If the charges for a service are calculated per time unit, the time unit in which the free trial period ends is always fully charged, no matter how much of the time unit falls into the free trial period. The charges for this time unit are calculated based on the price model elements that apply after the end of the free trial period, for example, the number of users assigned at that point of time, the defined parameter values, or the roles that are assigned to the users.

A free trial period that is defined for a service you provide as upgrade or downgrade option starts immediately after an upgrade or downgrade. It is independent of whether the service from which the customer migrates also defines a free trial period.

3.2.9 Discounts

A discount can be defined for a customer which applies to all subscriptions of the customer to services. A discount may be valid as of the current or a future month. It can be restricted to a certain period of time. Before the time expires, the customer is notified by email so that he can react and contact the supplier.

The discount is defined as a percentage that is subtracted from the regular total price for a subscription. It is granted for all costs of a customer that incur in a billing period in which the discount is valid. It does not matter whether the discount is valid for the whole billing period or only a part of it.

A discount is completely independent of what a customer might purchase. If a discount is changed, the new discount is valid the next time the billing data is generated. Usually, a discount is only changed in agreement with the relevant customer.

Discounts can be granted to a supplier's direct customers. They are not supported for customers of resellers and brokers who offer and sell the supplier's services.

Note that discounts are deducted from any agreed revenue shares.

3.3 Defining Price Models

To define a service price model, use the **Define for service** menu option in the **Price model** menu of the ESCM administration portal. Use the **Define for customer** menu option or the **Define for subscription** menu option to define a customer price model or a subscription price model, respectively.

Note:	Discounts are specified with the basic information for a customer. Use the **Manage customer** menu option in the **Customer** menu of the ESCM administration portal.

For each price model you define for services and subscriptions, you can decide whether the service is free of charge or whether and how the customer will be charged for using it.

If you decide to charge your customers, you can specify the following elements:

- A **price model description** that provides, for example, details of each element the price model consists of.

  The description is displayed to customers on a marketplace when they request details of the service from the service catalog. It can be specified in one or several languages. For details, refer to *Maintaining Price Models* on page 32.

- A **free trial period** if you want to offer the service to your customers for free for a limited period of time. For details, refer to *Free Trial Period* on page 30.
- **Charging conditions**, **prices for billable events**, **prices for parameters and options**, and **prices for roles** as described in *Price Model Elements* on page 21.

License Agreement

For each price model, you have to enter a license agreement. It is presented to customers and needs to be accepted by them. The underlying technical service comes with licensing information specified by the technology provider. You are free to use this information or to update it, if required.

The license agreement is only relevant for service price models and customer price models. A subscription price model automatically uses the license agreement defined for the relevant service price model.

The license agreement can be specified in one or several languages. For details, refer to *Maintaining Price Models* on page 32.

3.4 Maintaining Price Models

ESCM provides various features and functions for maintaining price models. You can:

- Update the definition of a price model.
- Localize elements of the price model definition.
- Delete a customer price model.

Updating the Definition of a Price Model

You can update the definition of a price model, but not change the service identifier, the customer in the case of a customer price model, and the customer subscription in the case of a subscription price model.

The update options depend on the billing system with which the price model is defined and used:

- You can update all elements of a price model definition in ESCM. Be aware of the following:

  In service price models and customer-specific price models, you can define and change the cost calculation option. When updating a price model, you can switch between the pro rata and the per time unit calculation. The prices are kept in this case. The prices get lost when a price model is changed to free of charge.

  If you change the price model of a specific subscription, The modified settings are valid for the current and future billing periods. This means that, for example, all user assignments or billable events that occurred in using the subscription in the current billing period before the change will be charged according to the new price model.

For updating a service price model or a customer price model, the corresponding marketable service must be deactivated. Changing a service price model or a customer price model does not affect existing subscriptions. The changes will only take effect for new subscriptions after the service has been activated again. Usually, you will change a customer price model only in agreement with the relevant customer.

To update a service price model, use the **Define for service** menu option in the **Price model** menu of the ESCM administration portal. To update a customer price model or a subscription price model, use the **Define for customer** or **Define for subscription** menu option, respectively.

If you update a price model for a service for which you have granted resale permissions, the brokers and resellers as well as their customers can instantly view any changes you make. A service that is already published is immediately offered with the new prices. Existing subscriptions to services are not affected by the update. Any changes you make are effective for new subscriptions only.

Resellers can provide their own license agreement. If you update the license agreement after granting a reseller the permission, the reseller's license agreement is not updated.

Localizing Elements of a Price Model

You can localize the description and license agreement of a price model to make them available in all languages supported by ESCM.

For localizing these elements, the corresponding marketable service must be deactivated. Existing subscriptions are not affected by the localization. Any changes you make are effective for new subscriptions only.

To localize the elements:

1. Use the **Define for service**, **Define for customer**, or **Define for subscription** menu option in the **Price model** menu of the ESCM administration portal depending on the type of price model whose elements you want to localize.
2. Select the price model to be localized on the **Define for service**, **Define for customer**, or **Define for subscription** page that is displayed.
3. Click **Localize** to enter the translations.

If you localize the elements for a service for which you have granted resale permissions, the brokers and resellers as well as their customers can instantly view the price model elements in the languages in which you provide them. Existing subscriptions to services sold by brokers or resellers are not affected by the localization. Any changes you make are effective for new subscriptions only.

Resellers can provide their own license agreement. If you localize the license agreement after granting a reseller the permission, the reseller's license agreement is not localized.

Deleting a Customer Price Model

You can delete a customer price model which you do not want to offer to a customer any longer. Usually, you delete a customer price model only in agreement with the relevant customer.

For deleting a customer price model, the relevant marketable service must be deactivated. Existing subscriptions are not affected by the deletion. The corresponding service price model is automatically applied for new subscriptions to the service.

To delete a customer price model, use the **Delete for customer** menu option in the **Price model** menu of the ESCM administration portal.

4 Customer Management

ESCM offers various features for managing your organization's interaction with customers. Customer management involves the following tasks:

- Registering customers.
- Configuring an external user authentication system, if required.
- Defining custom attributes.
- Handling subscriptions.

These tasks are described in more detail in the sections below.

4.1 Registering Customers

You can register customers with ESCM so that they do not need to do this themselves. The new customer automatically receives the login information that enables him to access the given marketplace and subscribe to services.

Registering a customer includes entering the following user data for the first administrator of the new customer organization:

- **Title**, **first name**, and **last name** of the administrator.
- **Email address** that is used for notifying the customer about the registration.
- **User ID** to be used for the administrator.
- **Language** in which the user interface, available documentation, and emails are presented to the administrator.

Registering a customer includes entering the data for the new organization:

- **Name**, **address**, and **phone number**.
- **Email address**.
- **Website**.
- **Country** where the customer organization is located.
- **Description** that provides, for example, information from the company's profile.
- **Marketplace** to which the customer is directed.

To register a customer, use the **Register customer** menu option in the **Customer** menu of the ESCM administration portal.

Once you have registered a customer, the organization data can only be edited by the customer's administrators. If you want to update the organization data, you must be registered as an administrator of your customer.

As a supplier, you can view the organization data that is defined for your customers. If you have granted brokers the permission to offer your services, you can also view the organization data that is defined for your brokers' customers. To view the data, use the **View customer** menu option in the **Customer** menu of the ESCM administration portal.

4.2 Configuring an External User Authentication System

Instead of maintaining user IDs and passwords in ESCM, your customer may want to connect to his organization's own identity management system for user authentication. ESCM supports user authentication against remote LDAP systems via secure VPN connections.

When using an LDAP system, your customer does not need to register his users manually with ESCM. The administrator of the customer organization can import the users, thus automatically registering them.

To enable access to the customer's LDAP system, you need to perform the following steps:

1. Create an LDAP configuration file on your file system.
2. Define the LDAP connection settings in the file. For a list of the relevant keys, refer to *LDAP Keys* on page 44.
3. Activate the LDAP-based authentication when registering the customer, and use the **LDAP settings** option to store the connection settings specified in the configuration file in ESCM.

   If you do not store any organization-specific connection settings in ESCM, ESCM will use default settings for LDAP connections.

Access to the LDAP system is enabled for the customer as soon as the customer is registered with ESCM. The administrator of the new customer organization can then import the required users from the LDAP system. The administrator can choose to import all users from the LDAP system or apply filter criteria to select specific users only. The imported users are sent an email with the URL for accessing the provided services. If required, the administrator of the customer organization can change the LDAP settings you defined for the organization.

User data maintained in the LDAP system such as the password, first name, or last name, can only be changed in the LDAP system. If the data is changed, ESCM automatically synchronizes its own data with it. An automatic synchronization is not possible in case a user ID is changed in the LDAP system. Such a user has to be imported again.

4.3 Defining Custom Attributes

Custom attributes are a means to store additional customer information such as data that is needed for controlling the cost and activity accounting. For example, you may want to specify the profit center to which a particular customer's revenue is to be accounted. Since there is no standard for such data, you can define your own custom attribute for storing the profit center. The custom attributes will be included in the billing data that is generated every month. The information can then be used, for example, as a reference for an external accounting, billing, or CRM system.

Custom attributes can be defined for customers and for subscriptions. For subscription attributes, suppliers can decide whether they themselves want to specify the value, or whether the customer must enter a value when he subscribes to a corresponding service. For customer attributes, suppliers can decide whether they themselves want to specify the value, or whether the customer administrator must enter a value by editing the customer organization's profile on a marketplace.

To define custom attributes, use the **Manage custom attributes** menu option in the **Account** menu of the ESCM administration portal.

To specify the value for a custom attribute for an individual customer, use the **Manage customer** menu option in the **Customer** menu of the ESCM administration portal. To specify the value for a custom attribute for an individual subscription, use the **Manage subscription attributes** menu option in the same menu.

Custom attributes are available for customers of your brokers when you grant them the permission to offer and sell your services. Custom attributes are not available for customers of your resellers.

4.4 Handling Subscriptions

You can view the details of all subscriptions customers have created for your services. The following information on a subscription is available:

- Activation date and number of users assigned to the subscription.
- Details of the marketable service to which the customer subscribed.
- Details of the underlying technical service.
- Details of the price model defined for the marketable service.

If you have granted brokers the permission to publish your services, you can also view the details of subscriptions of your brokers' customers. To view the details of a subscription, use the **View subscription** menu option in the **Customer** menu of the ESCM administration portal.

To offer support to customers, ESCM enables you to enter a support email address for your organization. This allows administrators and subscription managers of customer organizations to directly contact you in case they have questions or want to report issues on a service they have subscribed to. If you have granted brokers the permission to publish your services, your brokers' customers can also contact you via this email address. To enter a support email address, use the **Edit profile** menu option in the **Account** menu of the ESCM administration portal.

You can explicitly terminate a customer subscription. This may be required, for example, if a customer does not comply with a license agreement or does not pay for a subscription. Terminating a subscription is possible even if users are currently using the underlying service.

If you have granted brokers the permission to publish your services, you can also terminate subscriptions of your brokers' customers.

To terminate a subscription, use the **Terminate a subscription** menu option in the **Customer** menu of the ESCM administration portal. As soon as you terminate a subscription, the customer is sent an email informing him about the termination and the reason you have entered.

5 Billing and Payment

As users work with the services, ESCM automatically calculates the charges based on the related price models. It aggregates all the events, users, and configuration data associated with a subscription and rates them according to the associated price model.

At daily intervals, the database content is checked for organizations which have produced billing-relevant data. The data is collected and stored in the database. At monthly intervals (billing periods), the billing data is calculated. These checks and calculations are called "billing runs for customer billing data". The costs for all customer subscriptions that were produced in the course of a billing period are calculated by the billing run executed on the start day of the next billing period plus the offset defined by the operator.

Examples:

- You define that the billing period is to start on the 8th of a month, and the offset is set to 5 days and 4 hours. In this case, the billing run is executed on the 13th of each month at 04:00:00.000. The start day of the billing period, and thus the billing period itself, may be different for each supplier and reseller.
- Assume, the offset is set to 0 and the billing period starts on the 8th of a month. On January 5th, a customer subscribes to a service. The price model of this service defines per time unit calculation with charges based on MONTH as well as a one-time fee. On January 20th, the customer terminates this subscription.

  In this case:

  - The billing run executed on January 8th calculates the one-time fee only.
  - The billing run executed on February 8th calculates the time unit that ended on January 31st, i.e. the month of January is fully charged.

The customer billing data can be used for creating invoices. You are responsible for charging your customers and for collecting the payments.

If you grant a reseller the permission to sell your services, it is the reseller's responsibility to manage billing and payment for his customers. If you grant a broker the permission to offer your services, it is your responsibility to manage billing and payment for your broker's customers.

As a supplier, you are usually entitled to most of the revenue from your customers' service usage. ESCM offers various features that support you in retrieving information on the revenue shares you can claim from or need to pay to others, for example, brokers, resellers, marketplace owners, or the platform operator. The operator is responsible for defining the revenue shares. They may differ for individual marketplaces, brokers or resellers, or even specific services. You are not authorized to change any revenue shares.

The revenue shares 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 calendar month on the first day of a month plus the offset defined by the operator.

The following sections describe which preparatory steps you have to take for the billing and payment services in ESCM, how ESCM supports you in creating invoices and collecting payments, and which tasks are involved in handling revenue shares.

5.1 Managing Billing and Payment for Customers

Managing the billing and payment for customers involves the following tasks:

- Specify the payment types to be offered to customers.

- Define the VAT rates to be used for calculating the charges.
- Define your own billing period.
- Export billing data to make it available to external accounting systems.

5.1.1 Managing Payment Types

You are responsible for specifying how your customers and the customers of your brokers may pay for their subscriptions:

- On receipt of invoice.
- With credit card.
- By direct debit.

Your operator is responsible for configuring the payment types you can offer. Payment on receipt of invoice is automatically set up for all suppliers and resellers. This means that you can always allow your customers to pay on receipt of invoice without additional preparations by your operator. Payment on receipt of invoice implies that you are responsible for creating invoices and collecting payments.

ESCM allows you to specify the payment types for individual customers and individual services used by the customers. If you decide to offer more than one payment type to a customer, the customer is free to choose the type he prefers when subscribing to a service. If you remove a used payment type, the customer must specify a new one before he can continue using the service.

Be aware of the following effects of removing a used payment type:

- If you remove a used payment type during a customer's free trial period, the free trial period is not extended by the time the service could not be used.
- If you remove a used payment type for a subscription for which the charges are calculated per time unit, the customer may be charged with additional costs from stepped prices. Any customer who does not continue using the service in the time unit in which you removed the payment type, is affected. The resumed subscription is treated like a new subscription. This means that additional costs result, for example, from the sum of time units the users are assigned to the subscription or from the number of events that occur.

To specify the payment types, use the **Manage payment types** menu option in the **Customer** menu of the ESCM administration portal.

Note:	In specific scenarios, your operator may deactivate any possibilities for customers to select payment types and specify billing addresses. If this is the case, the definition of payment types is not possible.

5.1.2 Managing VAT Rates

In case you want to invoice usage charges for subscriptions as gross prices, you can enable VAT rate support for your supplier organization.

You can define a default VAT rate that applies to all prices for all customers. In addition, you can define country-specific or even customer-specific VAT rates. You can:

- Define a country-specific VAT rate for every country where you want to sell your services.
- Define a customer-specific VAT rate, for example, in case a customer organization has a subsidiary located in another country than its parent organization.

The VAT rate settings have the following effects on the cost calculation for a customer:

- If VAT rate support is disabled, prices are calculated as net prices; no VAT is added to the overall costs.
- A customer-specific VAT rate takes priority over any default or country-specific VAT rate.
- The country-specific VAT rate for the country where the customer organization is located is applied to the cost calculation when no customer-specific VAT rate is defined.
- The default VAT rate is used in all other cases.

The VAT rate does not affect any price model elements. The calculated VAT amount is added to the overall costs and results in the gross price to be paid by a customer.

If you grant a reseller the permission to sell your services, the customers of your reseller are not affected by the VAT rate you define. Any cost calculation for a reseller's customers results in net prices. If you grant a broker the permission to offer your services, the VAT rates you define are used.

To enable VAT rate support and set the VAT rates for your organization, use the **Manage VAT rates** menu option in the **Customer** menu of the ESCM administration portal.

5.1.3 Defining the Billing Period

A billing period is the time interval for which billing data for customer subscriptions is generated. It is always one month.

By default, a billing period starts on the 1st day of a month at 00:00:00.000 and ends on the last day of a month at 23:59:59.999. However, this is not always suitable in real business. For this reason, you can define your desired start day of the billing period.

When you change the start day of the billing period, this affects all subscriptions which are created or reactivated afterwards. Existing subscriptions are not affected.

The billing period start and end date and time are displayed in the **Customer billing report** and the **Revenue share report**.

To define the start day for billing periods, use the **Define billing period** menu option in the **Account** menu of the ESCM administration portal.

5.1.4 Exporting Billing Data

You can export the billing data from ESCM and process it using billing and payment facilities that have already been established in your organization. This is the standard procedure for all customers who have decided to pay on receipt of invoice.

You can export the data of one or several customers for a specific time frame. You can export the data of customers who subscribed to services you yourself have published or your brokers have published. The data can be saved to an XML file or opened in an editor of your choice. You can edit it and work with it as required. The data can be forwarded, for example, to an accounting system.

To export billing data, use the **Export billing data** menu option in the **Account** menu of the ESCM administration portal.

Refer to *Customer Billing Data* on page 46 for a detailed description of the elements and attributes that may occur in the generated XML file.

5.2 Handling Revenue Shares

ESCM offers the following options for handling your revenue share from your customers' service usage:

- You can view the revenue shares that have been defined by the operator.
- You can retrieve and export information on the actual revenue.

For details, refer to the following sections. You can also use reports for retrieving information on your actual revenue. For details, refer to *Reporting* on page 41.

Viewing the Revenue Share

You can view the percentage you are entitled to keep from the total revenue generated by a service, as well as the percentage the operator, marketplace owner, brokers, and resellers are entitled to receive.

For this purpose, use the **Define publishing options** menu option in the **Marketable service** menu of the ESCM administration portal.

Exporting Revenue Share Data

At the end of each calendar month (plus an offset defined by the operator), ESCM calculates the revenue share data for a marketplace.

To retrieve information on your actual revenue for a specific time frame, ESCM allows you to export the corresponding data. To do this, use the **Export billing data** menu option in the **Account** menu of the ESCM administration portal.

The data can be saved to an XML file or opened in an editor of your choice. You can edit it and work with it as required. The data can, for example, be forwarded to an accounting system. Depending on your contracts with the operator, the marketplace owner, and the brokers or resellers, you can use it for paying them their respective shares, or for forecasting the shares that they will collect from you.

Refer to *Customer Billing Data* on page 46 for a detailed description of the elements and attributes that may occur in the generated XML file.

6 Reporting

ESCM offers comprehensive reports for different purposes and at different levels of detail.

The following reports are available for suppliers:

- **Service report.** It shows all your marketable services with their existing subscriptions. Subscriptions to services offered by your brokers are also listed.

  A service report supports you, for example, in identifying unsold services or in gaining a quick overview of the services that need special promotion.

- **Customer report.** It shows all your customers and your brokers' customers with basic customer data and the services they subscribed to. The report outputs whether a subscription is still active or has already ended.

  A customer report supports you, for example, in gaining an overview of your customers and in quickly accessing customer data.

- **Billing report.** It shows a summary of all billing data for your customers and your brokers' customers.
- **Payment processing status report.** It shows the status of billing data calculations for your customers and your brokers' customers, their subscriptions, and the services they subscribed to.
- **Customer billing report.** It shows the billing data of the current billing period for a selected subscription of a customer.

  In combination with the payment processing status report, this report provides a detailed audit log for billing and payment issues. The data can be used, for example, to verify payment or service access issues.

- **Revenue share report.** It shows your actual revenue depending on the usage of your services. The report breaks down the data to the organizations which generated the revenue as well as to the individual services.

  The report outputs your liabilities to the other participating parties. The data can be used as the basis for collecting or paying the shares, depending on your contracts with the other participating parties.

To create a report, use the **Create report** menu option in the **Account** menu of the ESCM administration portal and choose the desired report. Depending on the report type, you may have to enter additional report parameters.

The generated report is instantly displayed at the ESCM administration portal. You can choose to print the report or save it in several formats.

Note:	Contact your platform operator if the reporting functionality is not available. He is responsible for defining the respective configuration parameters.

7 Integrating External Process Control

Organizations often have specific processes for registering users, subscribing to services, or defining prices. Usually, such processes include approval processes and are modeled and automated with a process control system.

Certain actions of customers and suppliers can be carried out under the control of an external process control system. You can configure so-called triggers which are invoked when these actions are carried out. The triggers start the corresponding process in the process control system. If approval for the action is required, it is suspended until it is approved or rejected in the process control system. If no approval is required, the action is instantly executed.

As a prerequisite for controlling actions by processes, a notification service must exist and be deployed. This service forms the interface between the platform and the process control system.

For details on implementing such a service, contact the platform operator.

Users can see all pending actions at the ESCM user interface, cancel them, or delete aborted ones.

Process-Controlled Actions

The following actions may be subject to process control and thus to approval in an external process control system before they are executed in ESCM:

- For any type of organization, the following triggers can be configured:
  - A subscription is to be changed, for example, renamed (`Modify subscription`).
  - A billing run is completed, and the billing data for a billing period is calculated (`Billing run finished`).
  - A subscription is to be added (`Subscribe to service`).
  - A user is to be assigned to or removed from a subscription (`Assign users to subscription`).

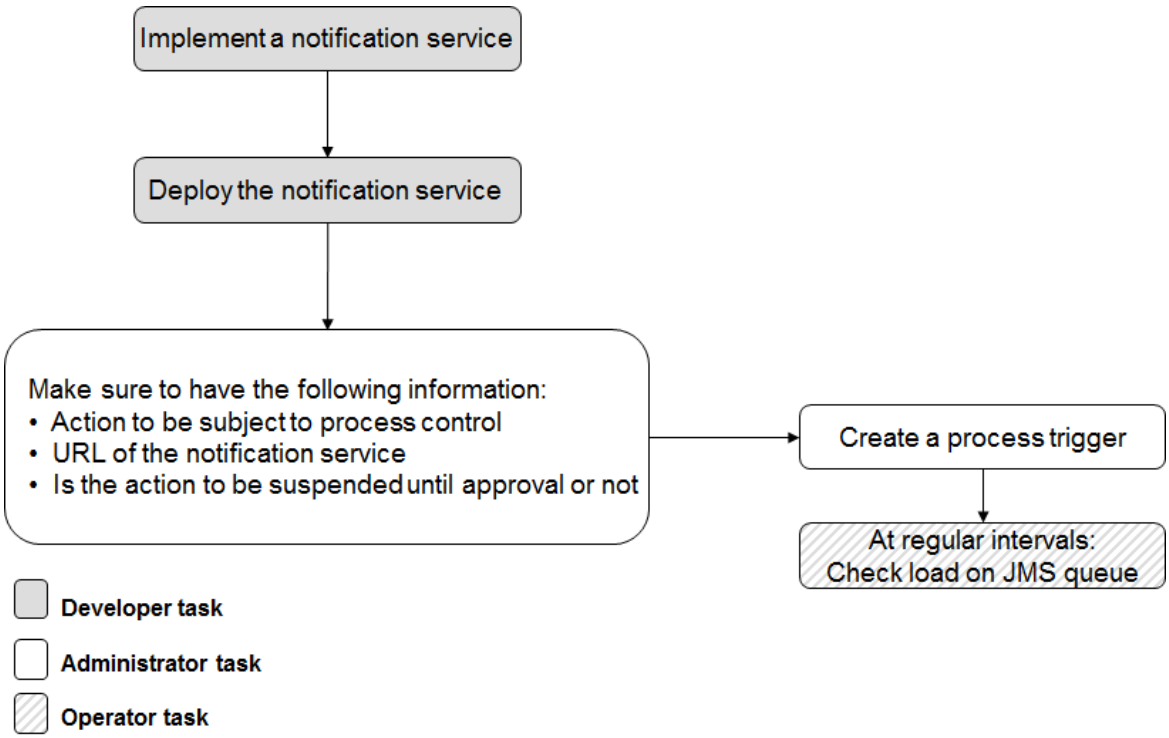
    For the case that a user is automatically assigned to a new subscription when subscribing to a service, no trigger can be configured. The `Assign users to subscription` trigger only applies to users that are manually assigned to a subscription.
  - A subscription is to be upgraded or downgraded (`Up/Downgrade subscription`).
  - A subscription is to be terminated (`Terminate subscription`).
  - A user is to be registered (`Register user`).

    Triggers can only be configured for registering single users. The import of multiple users in one operation from an LDAP system or a user data file cannot be executed under process control.
- For suppliers, the following additional triggers can be configured:
  - A customer is to be registered (`Register customer`).
  - The payment types for a customer are to be changed (`Manage payment types for customer`).
  - A marketable service is to be activated (`Activate service`).
  - A marketable service is to be deactivated (`Deactivate service`).
  - A user of an organization subscribes to a service offered by the supplier (`Subscription created (any user)`).

- A user of a customer modifies a subscription to a service of the supplier (`Subscription modified (any user)`).
- A user of a customer terminates a subscription to a service of the supplier (`Subscription terminated (any user)`).

Involved Tasks

The following figure provides an overview of the steps required for integrating an external process control system with ESCM:



Creating a Process Trigger

To create a process trigger, use the **Process triggers** menu option in the **Account** menu of the ESCM administration portal.

For configuring the triggers for your organization, you need the following information:

- Which action is to be subject to process control?
- How can the notification service of the external process control system be reached? You need the URL of the notification service. This URL depends on the application programming interface used for implementing the notification service.
- Is the action to be suspended until it is approved or rejected? If not, it will instantly be executed in ESCM. The process control system is informed if the action for a non-suspending trigger is executed successfully. If the action is to be suspended, ESCM will wait for a reply from the external system before the action is executed.

Appendix A: 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	Currently not used.
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`
LDAP_ATTR_REFERRAL	Optional. Property defining how LDAP referrals are to be processed. If an organization uses an Active Directory with sub-domains from which users are to be imported into ESCM, the sub-domains can be modeled as referrals. In this sense, a referral is a reference to another directory partition or sub-domain. By default, values from referrals are not retrieved. This property can take on the following values: `follow`: Referrals are followed, i.e. users are imported from all referenced directory partitions or sub-domains. `ignore`: Referrals are ignored (default), i.e. users are imported from the current domain directory only. Example: `LDAP_ATTR_REFERRAL=ignore`

Appendix B: Customer Billing Data

The charges for the usage of a service in ESCM are calculated based on the price model defined for the service, customer, or subscription.

A supplier or reseller can export the billing data for one or several of his customers for a specific time. Suppliers also have access to the billing data of customers of broker organizations that sell their services. The exported data can be forwarded, for example, to an accounting system for further processing.

The result of the export is stored in an XML file, the customer billing data file. The billing data file conforms to the XML schema `BillingResult.xsd`, which can be found in the ESCM integration package.

The billing data file is named `<date>BillingData.xml`, where `<date>` represents the creation date.

This appendix describes the meaning of the elements and attributes that may occur in a billing data file.

BillingDetails

Top-level container element of a billing data file. For each subscription, a `BillingDetails` element is added to the billing data file.

A `BillingDetails` element contains the following subelements:

- `Period` (see *Period* on page 46)
- `OrganizationDetails` (see *OrganizationDetails* on page 47)
- `Subscriptions` (see *Subscriptions* on page 47)
- `OverallCosts` (see *OverallCosts* on page 60)

A `BillingDetails` element has the following attributes:

`key` - (optional, data type `long`) Unique identifier allowing, for example, accounting systems to relate billing data to an invoice. The billing data key is printed on the invoice. Suppliers and customers may use the billing data key to create a detailed billing report for an existing invoice or subscription. A supplier can retrieve the key from a billing report, a customer gets the billing data key from the corresponding invoice.

`timezone` - (required, data type `string`) Time zone based on the UTC time standard. It reflects the standard server time without daylight saving time. For example, 18:00:00 o'clock on June 1st in Berlin (UTC+1) will be output as follows in the XML file:

```
<BillingDetails timezone="UTC+01:00" key="31122">
  <Period startDate="1370106000000"
  startDateIsoFormat="2013-06-01T16:00:00.000Z" ..."/>
```

The time zone is relevant for price models with costs (see *PriceModel* on page 49).

Period

Specifies the billing period for which the data is exported. The start and end time of the billing period are output according to the start day of the billing period which was defined by the supplier or reseller.

A `Period` element has the following attributes:

- **`startDate`** - (data type `long`) Start time of the period. The start time is specified in milliseconds, the starting point for the calculation is `1970-01-01, 00:00`.

- **startDateIsoFormat** - (optional, data type dateTime) Same as startDate, but specified in ISO 8601 format (YYYY-MM-DDThh:mm:ss.fffZ) .
- **endDate** - (data type long) End time of the period. The end time is specified in milliseconds, the starting point for the calculation is 1970-01-01, 00:00.
- **endDateIsoFormat** - (optional, data type dateTime) Same as endDate, but specified in ISO 8601 format (YYYY-MM-DDThh:mm:ss.fffZ).

Example:

```
<Period startDateIsoFormat="2012-08-31T22:00:00.000Z"
  startDate="1346450400000"
  endDateIsoFormat="2012-09-30T22:00:00.000Z"
  endDate="1349042400000"/>
```

OrganizationDetails

Provides details of the customer for which the billing data have been exported. The details may include a Udas element with custom attributes that store additional information on the customer organization.

An OrganizationDetails element contains the following subelements:

Email

Specifies the email address of the organization (data type string).

Name

Specifies the name of the organization (data type string).

Address

Specifies the address of the organization (data type string).

Paymenttype

Specifies the payment type used for subscriptions of the organization (data type string).

Example:

```
<BillingDetails key="10002" timezone="UTC+01:00">
...
  <OrganizationDetails>
    <Email>info@company.com</Email>
    <Name>company</Name>
    <Address>Street</Address>
    <Paymenttype>INVOICE</Paymenttype>
  </OrganizationDetails>
...
</BillingDetails>
```

Subscriptions

Contains the billing data for the subscriptions of the customer which are relevant for the current billing period.

For every subscription of an organization, the Subscriptions element contains a Subscription element. In this element, the costs of the affected subscription are specified.

A Subscription element has the following attributes:

- **id** - (required, data type string) Unique subscription name.

- **purchaseOrderNumber** - (data type string) Optional reference number as specified by the customer when subscribing to a service.

A Subscription element contains a PriceModel element with the billing data for the price model of the subscription (see *PriceModel* on page 49). A Udas element with custom attributes that store additional information on the subscription may also be included (see *Udas* on page 48).

If a subscription is assigned to an organizational unit at the end of the billing period, the Subscription element also contains an OrganizationalUnit element with the following attributes:

- **name** - (required, data type string) Required name of the organizational unit to which the subscription is assigned.
- **referenceID** - (optional, data type string) Optional reference ID of the organizational unit to which the subscription is assigned.

Be aware that the organizational units to which the subscription may have been assigned before the generation of the billing data are not shown.

Example:

```
<BillingDetails key="10002" timezone="UTC+01:00">
...
  <Subscriptions>
    <Subscription id="Mega Office Basic" purchaseOrderNumber="12345">
      <OrganizationalUnit name="ProjectTeam" referenceID="123abc"/>
      <PriceModels>
        <PriceModel calculationMode="PRO_RATA" id="14001">
...
        </PriceModel>
      </PriceModels>
    </Subscription>
  </Subscriptions>
...
</BillingDetails>
```

Udas

Contains custom attributes that store additional information on an organization or subscription. This could be, for example, the profit center to which a customer's revenue is to be accounted.

A Udas element may be included in an OrganizationDetails or a Subscription element.

For every custom attribute, a Uda element is included in the Udas element.

A Uda element has the following attributes:

- **id** - (required, data type string) ID of the custom attribute.
- **value** - (required, data type string) Value of the custom attribute.

Example:

```
<BillingDetails key="10002" timezone="UTC+01:00">
...
  <Subscriptions>
    <Subscription id="Mega Office Basic" purchaseOrderNumber="12345">
...
      <Udas>
        <Uda id="Profit Center" value="My Company"/>
      </Udas>
    </Subscription>
  </Subscriptions>
```

```
...
</BillingDetails>
```

PriceModel

Contains the billing data for a price model used to calculate the utilization charges for a subscription.

A `PriceModel` element is included in every subscription element. It contains the following subelements:

- `UsagePeriod` (see *UsagePeriod* on page 49)
- `GatheredEvents` (see *GatheredEvents* on page 50)
- `PeriodFee` (see *PeriodFee* on page 51)
- `UserAssignmentCosts` (see *UserAssignmentCosts* on page 52)
- `OneTimeFee` (see *OneTimeFee* on page 53)
- `PriceModelCosts` (see *PriceModelCosts* on page 53)
- `Parameters` (see *Parameters* on page 54)

A `PriceModel` element has the following attributes:

`id` - (required, data type `string`) Unique name identifying the price model.

`calculationMode` - (required, data type `string`) Cost calculation option of the price model. Can be set to one of the following values: `FREE_OF_CHARGE` (the service is free of charge), `PRO_RATA` (the costs are calculated exactly for the time a service is used, based on milliseconds), `PER_UNIT` (the costs are calculated based on fixed time units).

UsagePeriod

Specifies the actual period in which a price model is used for calculating the charges of a subscription.

A usage period usually begins when a customer subscribes to a service and ends when the subscription is terminated. In case a free trial period is defined for the service, the usage period begins when the free trial period has ended. When the customer upgrades or downgrades the subscription, a new usage period is started in which the price model of the new service is applied. If the customer changes elements that determine how the charges for the service are calculated (e.g. the number of assigned users, the service roles of the assigned users, or the parameter values), a new usage period is started in which the updated elements are applied.

A `UsagePeriod` element is contained in a `PriceModel` element.

A `UsagePeriod` element has the following attributes:

- **`startDate`** - (data type `long`) Start time of the period. The start time is specified in milliseconds, the starting point for the calculation is `1970-01-01, 00:00`.
- **`startDateIsoFormat`** - (optional, data type `dateTime`) Same as `startDate`, but specified in ISO 8601 format (`YYYY-MM-DDThh:mm:ss.fffZ`) .
- **`endDate`** - (data type `long`) End time of the period. The end time is specified in milliseconds, the starting point for the calculation is `1970-01-01, 00:00`.
- **`endDateIsoFormat`** - (optional, data type `dateTime`) Same as `endDate`, but specified in ISO 8601 format (`YYYY-MM-DDThh:mm:ss.fffZ`).

Example:

```
<PriceModel calculationMode="PRO_RATA" id="14001">
```

```
  <UsagePeriod endDate="1306879200000"
   endDateIsoFormat="2011-05-31T22:00:00.000Z"
   startDate="1304755088065"
   startDateIsoFormat="2011-05-07T07:58:08.065Z"/>
...
</PriceModel>
```

GatheredEvents

Specifies the costs for all chargeable events that occurred in the current usage period of the subscription. These include, for example, login and logout by users to the underlying application, the completion of specific transactions, or the creation or deletion of specific data. It depends on the implementation and integration of the underlying application which events are available.

A `GatheredEvents` element is contained in a `PriceModel` element.

A `GatheredEvents` element contains the following subelements:

- `Event`
- `GatheredEventsCosts`

Event

For every event, an `Event` element is included in the `GatheredEvents` element.

An `Event` element has the following attribute:

`id` - (required, data type `string`) Event ID as specified in the technical service definition.

An `Event` element contains the following subelements:

- `Description`
- `SingleCost`
- `NumberOfOccurence`
- `CostForEventType`

Description

Contains the description of the event.

SingleCost

Specifies the price for the event as defined in the price model. If an event has stepped prices, this element is omitted. A `SteppedPrices` element is included instead (see *SteppedPrices* on page 59).

A `SingleCost` element has the following attribute:

`amount` - (required, data type `decimal`) Price for a single event.

NumberOfOccurence

Specifies how often the event occurred.

A `NumberOfOccurence` element has the following attribute:

`amount` - (required, data type `long`) Number of times the event occurred.

CostForEventType

Specifies the total costs for the event in the billing period.

A `CostForEventType` element has the following attribute:

`amount` - (required, data type `decimal`) Total costs for the event. The total costs for an event are calculated from the singular costs (`SingleCost`) multiplied with the number of occurrences (`NumberOfOccurence`). If role-based costs and/or stepped prices are specified for events, these

costs are added (see *RoleCosts* on page 58 and *SteppedPrices* on page 59). The value is rounded to two decimal places.

GatheredEventsCosts

Specifies the total costs for all events in the current `GatheredEvents` element.

A `GatheredEventsCosts` element has the following attribute:

`amount` - (required, data type `decimal`) Total costs for events. The value is rounded to two decimal places.

Example:

```
<PriceModel calculationMode="PRO_RATA" id="14001">
...
  <GatheredEvents>
    <Event id="USER_LOGOUT_FROM_SERVICE">
      <Description xml:lang="en">Logout from the service.</Description>
      <SingleCost amount="100.00"/>
      <NumberOfOccurrence amount="3"/>
      <CostForEventType amount="300.00"/>
    </Event>
...
    <GatheredEventsCosts amount="1200.00"/>
  </GatheredEvents>
...
</PriceModel>
```

PeriodFee

Specifies the costs for using the subscription in the given usage period.

For each subscription, a charge can be defined that a customer has to pay on a recurring basis. Monthly, weekly, daily, or hourly periods are supported. The recurring charge for a subscription is independent of the amount of users, events, or other usage data.

The calculation of the charges depends on the cost calculation option which was chosen for the price model (see *PriceModel* on page 49 for details).

A `PeriodFee` element is contained in a `PriceModel` element.

A `PeriodFee` element has the following attributes:

- **`basePeriod`** - (required, data type `string`) Period on which the charges are based. Can be set to one of the following values: `MONTH`, `WEEK`, `DAY`, `HOUR`.
- **`basePrice`** - (required, data type `decimal`) Recurring charge per base period according to the price model.
- **`factor`** - (required, data type `decimal`) Factor used to calculate the period fee for the subscription. The factor is calculated from the usage period of the subscription divided by the base period (`basePeriod`). The recurring charge is multiplied with this factor to calculate the total costs (`price`).
- **`price`** - (required, data type `decimal`) Total period fee for the subscription. This value is calculated from the recurring charge (`basePrice`) multiplied with the factor (`factor`). The value is rounded to two decimal places.

Example:

```
<PriceModel calculationMode="PRO_RATA" id="14001">
...
  <PeriodFee basePeriod="MONTH" basePrice="10.00"
 factor="0.4020212567204301" price="4.02"/>
```

```
...
</PriceModel>
```

UserAssignmentCosts

Specifies the costs for the user assignments to the subscription.

For the users assigned to a subscription, a charge can be defined that a customer has to pay on a recurring basis. Monthly, weekly, daily, or hourly periods are supported. The charge depends on the amount of time units one or more users are assigned to the subscription. This type of charge can only be defined for services with the login or user access type.

The recurring charge for users is independent of the recurring charge per subscription or other usage data.

For this type of charge, stepped prices can be applied: Recurring charges can be defined that depend on the sum of the time units of all user assignments.

The calculation of the charges depends on the cost calculation option which was chosen for the price model (see *PriceModel* on page 49 for details).

With per time unit calculation, the costs for a time unit in which a user is assigned to a subscription are always fully charged. There is no difference in the costs between a user who is assigned from the start until the end of the time unit and a user who is assigned for a part of the time unit only. A time unit is charged only once if a user is deassigned from and re-assigned to a subscription within the same time unit. Yet, canceling an assignment, deleting the user, and then creating a new user with the same user ID is treated as if two different users are assigned to the subscription. The time unit is charged twice, accordingly.

A `UserAssignmentCosts` element is contained in a `PriceModel` element.

A `UserAssignmentCosts` element has the following attributes:

- **`basePeriod`** - (optional, data type `string`) Period on which the charges are based. Can be set to one of the following values: `MONTH`, `WEEK`, `DAY`, `HOUR`.
- **`basePrice`** - (optional, data type `decimal`) Recurring charge for users per base period according to the price model. If the charge for users has stepped prices, this attribute is omitted.
- **`factor`** - (optional, data type `decimal`) Factor used to calculate the costs for the user assignments. The factor is calculated by summing up the factors for each user specified in the `UserAssignmentCostsByUser` element. The recurring charge (`basePrice`) is multiplied with this factor to calculate the costs (`price`).
- **`numberOfUsersTotal`** - (optional, data type `long`) Number of users assigned to the subscription in the usage period.
- **`total`** - (data type `decimal`) Total costs for the user assignments including role-based costs and stepped prices. The value is rounded to two decimal places. For details on role-based costs and stepped prices, refer to *RoleCosts* on page 58 and *SteppedPrices* on page 59.
- **`price`** - (optional, data type `decimal`) Costs for the user assignments. This value is calculated from the recurring charge (`basePrice`) multiplied with the factor (`factor`). The value is rounded to two decimal places.

A `UserAssignmentCosts` element contains the following subelement. If role-based costs and/or stepped prices are specified, a `RoleCosts` element and/or `SteppedPrices` element is also present (see *RoleCosts* on page 58 and *SteppedPrices* on page 59).

UserAssignmentCostsByUser

Specifies the fraction of the usage period a user was assigned to the subscription.

A `UserAssignmentCostsByUser` element has the following attributes:

- **`factor`** - (required, data type `decimal`) Fraction of the usage period the given user was assigned to the subscription. The factors of the single user assignments are summed up to calculate the total costs for the user assignments.
- **`userId`** - (required, data type `string`) User ID.

Example:

```
<PriceModel calculationMode="PRO_RATA" id="18000">
...
  <UserAssignmentCosts basePeriod="MONTH" basePrice="19.00"
   factor="0.5337726052867383" numberOfUsersTotal="2"
   total ="50.00" price="10.14">
     <UserAssignmentCostsByUser factor="1.0499215949820788E-4"
     userId="admin"/>
     <UserAssignmentCostsByUser factor="0.5336676131272401"
     userId="miller"/>
   </UserAssignmentCosts>
...
</PriceModel>
```

OneTimeFee

Specifies the one-time fee for the subscription.

A one-time fee defines the amount a customer has to pay for a subscription in the first billing period of the subscription. It is added to the total charges for the first billing period. It is independent of the number of users, events, or other usage data.

If a one-time fee is defined for a service to which a customer upgrades or downgrades a subscription, it is added to the total charges for the customer, even if the service from which the customer migrates also defines a one-time fee.

A `OneTimeFee` element is contained in a `PriceModel` element.

A `OneTimeFee` element has the following attributes:

- **`amount`** - (required, data type `decimal`) Total costs for the one-time fee. The value is rounded to two decimal places.
- **`baseAmount`** - (required, data type `decimal`) One-time fee as defined in the price model.
- **`factor`** - (required, data type `long`) Factor used for calculating the one-time fee. Since this charge occurs only once, the factor is 1 for the first billing period, and 0 in case the one-time fee has already been charged in a previous billing period.

Example:

```
<PriceModel calculationMode="PRO_RATA" id="14001">
...
  <OneTimeFee amount="10.00" baseAmount="10.00" factor="1"/>
...
</PriceModel>
```

PriceModelCosts

Specifies the total costs for the subscription in the current usage period.

A `PriceModelCosts` element is contained in a `PriceModel` element.

A `PriceModelCosts` element has the following attributes:

- **`currency`** - (required, data type `string`) ISO code of the currency in which the costs are calculated.
- **`amount`** - (required, data type `decimal`) Total amount of the costs for the subscription. The value is rounded to two decimal places.

Example:

```
<PriceModel calculationMode="PRO_RATA" id="14001">
...
  <PriceModelCosts currency="EUR" amount="990.00"/>
</PriceModel>
```

Parameters

Specifies the costs for parameters defined for the service underlying the subscription.

A price model can define prices for service parameters and options. It depends on the implementation and integration of the underlying application whether and which parameters and options are available.

A price can be defined for every parameter and option, and the price can be charged per subscription or per user assigned to the subscription. Numeric parameters are a multiplier for the price. For boolean parameters, the multiplier is `1` if the value is `true`. In all other cases, the multiplier is `0`.

The calculation of charges for parameters and options depends on the cost calculation option which was chosen for the price model (see *PriceModel* on page 49 for details).

If the charges for a subscription are calculated per time unit and a customer changes a parameter value within a time unit, the affected time unit is charged pro rata. This means that the customer is charged exactly for the time each parameter value is set.

For numeric parameters, stepped prices can be applied per subscription: Different prices can be defined depending on the parameter values.

The prices for parameters and options are independent of other price model elements.

A `Parameters` element is contained in a `PriceModel` element.

A `Parameters` element contains the following subelements:

- `Parameter`
- `ParametersCosts`

Parameter

For every parameter, a `Parameter` element is included in the `Parameters` element.

A `Parameter` element has the following attribute:

`id` - (required, data type `string`) Parameter ID.

A `Parameter` element contains the following subelements:

- `ParameterUsagePeriod`
- `ParameterValue`
- `PeriodFee`
- `UserAssignmentCosts`
- `ParameterCosts`
- `Options`

ParameterUsagePeriod

Specifies the actual usage period for the parameter.

The usage period for a parameter begins when a customer subscribes to the service with the given parameter definition in the price model and ends when the subscription is terminated.

In case a free trial period is defined for the service, the usage period for the subscription begins when the free trial period has ended. If a parameter value is changed, a new usage period is started in which the updated value is applied for calculating the costs.

A `ParameterUsagePeriod` element has the following attributes:

- **`startDate`** - (data type `long`) Start time of the period. The start time is specified in milliseconds, the starting point for the calculation is `1970-01-01, 00:00`.
- **`startDateIsoFormat`** - (optional, data type `dateTime`) Same as `startDate`, but specified in ISO 8601 format (`YYYY-MM-DDThh:mm:ss.fffZ`) .
- **`endDate`** - (data type `long`) End time of the period. The end time is specified in milliseconds, the starting point for the calculation is `1970-01-01, 00:00`.
- **`endDateIsoFormat`** - (optional, data type `dateTime`) Same as `endDate`, but specified in ISO 8601 format (`YYYY-MM-DDThh:mm:ss.fffZ`).

ParameterValue

Specifies the costs and data type for the parameter.

A `ParameterValue` element has the following attributes:

- **`amount`** - (required, data type `string`) Costs for the parameter as defined in the price model.
- **`type`** - (required, data type `string`) Data type of the parameter. Can be set to one of the following values: `BOOLEAN`, `INTEGER`, `LONG`, `STRING`, `ENUMERATION`, `DURATION`.

PeriodFee

Specifies the costs for using the parameter in the given usage period. If a parameter has stepped prices, a `SteppedPrices` element is included in the `PeriodFee` element.

A `PeriodFee` element has the following attributes:

- **`basePeriod`** - (required, data type `string`) Period on which the charges are based. Can be set to one of the following values: `MONTH`, `WEEK`, `DAY`, `HOUR`.
- **`basePrice`** - (optional, data type `decimal`) Recurring charge per base period according to the price model. If a parameter has stepped prices, this attribute is omitted.
- **`factor`** - (required, data type `decimal`) Factor used to calculate the costs for using the parameter. The factor is calculated from the usage period divided by the base period (`basePeriod`). The recurring charge (`basePrice`) is multiplied with this factor to calculate the costs (`price`).
- **`price`** - (required, data type `decimal`) Costs for using the parameter. This value is calculated from the recurring charge (`basePrice`) multiplied with the factors (`factor` and `valueFactor`). The value is rounded to two decimal places.
- **`valueFactor`** - (required, data type `float`) Factor to calculate the total costs for using the parameter depending on the parameter value. The recurring charge (`basePrice`) is multiplied with this factor to calculate the costs (`price`). This factor is set depending on the data type of the parameter. For numeric parameters it is set to the value of the parameter. For boolean parameters, the factor is set to `1` if the value is `true`. In all other cases, the factor is set to `0`.

UserAssignmentCosts

Specifies the costs for the parameter related to the user assignments of the subscription based on the price per user for the parameter as defined in the price model. If costs for service roles are

defined, a `RoleCosts` element is included in the `UserAssignmentCosts` element (see *RoleCosts* on page 58 for details).

A `UserAssignmentCosts` element has the following attributes:

- **`basePeriod`** - (required, data type `string`) Period on which the charges are based. Can be set to one of the following values: `MONTH`, `WEEK`, `DAY`, `HOUR`.
- **`basePrice`** - (required, data type `decimal`) Recurring charge for users per base period for the parameter according to the price model.
- **`factor`** - (required, data type `decimal`) Factor used to calculate the costs for using the parameter. The factor is calculated from the parameter usage period divided by the base period (`basePeriod`) multiplied with the number of users. The recurring charge (`basePrice`) is multiplied with this factor to calculate the costs (`price`).
- **`price`** - (required, data type `decimal`) Costs for using the parameter. This value is calculated from the recurring charge (`basePrice`) multiplied with the factors (`factor` and `valueFactor`). The value is rounded to two decimal places. If stepped prices are defined for user assignments, the costs are given in the `price` attribute.
- **`total`** - (data type `decimal`) Total costs for using the parameter including role-based costs. The value is rounded to two decimal places. For details on role-based costs, refer to *RoleCosts* on page 58.
- **`valueFactor`** - (required, data type `float`) Factor to calculate the total costs for using the parameter depending on the parameter value. The recurring charge (`basePrice`) is multiplied with this factor to calculate the costs (`price`). This factor is set depending on the data type of the parameter. For numeric parameters it is set to the value of the parameter. For boolean parameters, the factor is set to `1` if the value is `true`. In all other cases, the factor is set to `0`.

ParameterCosts

Specifies the total costs for using the parameter.

A `ParameterCosts` element has the following attribute:

`amount` - (required, data type `decimal`) Total costs for the parameter calculated by summing up the costs specified in the `PeriodFee` and the `UserAssignmentCosts` element for the parameter and its options. If role-based costs and/or stepped prices are specified for the parameter, these are added (see *RoleCosts* on page 58 and *SteppedPrices* on page 59). The value is rounded to two decimal places.

ParametersCosts

Specifies the total costs for all parameters.

A `ParametersCosts` element has the following attribute:

`amount` - (required, data type `decimal`) Total costs for the parameters calculated by summing up the costs of the individual parameters as specified in the `ParameterCosts` elements. The value is rounded to two decimal places.

Example:

```
<Parameters>
...
  <Parameter id="MAX_FOLDER_NUMBER2">
    <ParameterUsagePeriod endDate="1306879200000"
     endDateIsoFormat="2011-05-31T22:00:00.000Z"
     startDate="1304755088065"
     startDateIsoFormat="2011-05-07T07:58:08.065Z"/>
    <ParameterValue amount="200" type="INTEGER"/>
    <PeriodFee basePeriod="MONTH" basePrice="0.00"
     factor="0.5337789669205496" price="0.00" valueFactor="200.0"/>
```

```
    <UserAssignmentCosts basePeriod="MONTH" basePrice="0.00"
     factor="0.5337726052867383" total ="0.00"
     price="0.00" valueFactor="200.0"/>
    <ParameterCosts amount="0.00"/>
  </Parameter>

...
  <ParametersCosts amount="600.00"/>
</Parameters>
```

Options

Specifies the costs for parameter options.

An `Options` element is contained in a `Parameter` element.

For every option, an `Option` element is included in the `Options` element.

An `Option` element has the following attribute:

`id` - (required, data type `string`) Option ID.

An `Option` element contains the following subelements:

- `PeriodFee`
- `UserAssignmentCosts`
- `OptionCosts`

PeriodFee

Specifies the costs for using the parameter option in the given usage period.

A `PeriodFee` element has the following attributes:

- **`basePeriod`** - (required, data type `string`) Period on which the charges are based. Can be set to one of the following values: `MONTH`, `WEEK`, `DAY`, `HOUR`.
- **`basePrice`** - (required, data type `decimal`) Recurring charge per base period according to the price model.
- **`factor`** - (required, data type `decimal`) Factor used to calculate the costs for using the parameter option. The factor is calculated from the usage period divided by the base period (`basePeriod`). The recurring charge (`basePrice`) is multiplied with this factor to calculate the total costs (`price`).
- **`price`** - (required, data type `decimal`) Costs for the parameter option. This value is calculated from the recurring charge (`basePrice`) multiplied with the factor (`factor`), and is rounded to two decimal places.

UserAssignmentCosts

Specifies the costs for the parameter option related to the user assignments of the subscription based on the price per user for the option as defined in the price model. If costs for service roles are defined, a `RoleCosts` element is included in the `UserAssignmentCosts` element (see *RoleCosts* on page 58).

A `UserAssignmentCosts` element has the following attributes:

- **`basePeriod`** - (required, data type `string`) Period on which the charges are based. Can be set to one of the following values: `MONTH`, `WEEK`, `DAY`, `HOUR`.
- **`basePrice`** - (required, data type `decimal`) Recurring charge for users per base period for the parameter option according to the price model.
- **`factor`** - (required, data type `decimal`) Factor used to calculate the costs for using the parameter option. The factor is calculated from the usage period divided by the base period

(basePeriod). The recurring charge (basePrice) is multiplied with this factor to calculate the costs (price).

- **total** - (data type decimal) Total costs for using the parameter option including role-based costs. The value is rounded to two decimal places. For details on role-based costs, refer to *RoleCosts* on page 58.
- **price** - (required, data type decimal) Costs for using the parameter option. This value is calculated from the recurring charge (basePrice) multiplied with the factor (factor). The value is rounded to two decimal places. If stepped prices are defined for user assignments, the costs are given in the price attribute.

OptionCosts

Specifies the total costs for using the parameter option.

An OptionCosts element has the following attribute:

amount - (required, data type decimal) Total costs for the parameter option calculated by summing up the costs specified in the PeriodFee and the UserAssignmentCosts element. The value is rounded to two decimal places.

Example:

```
<Parameter id="MEMORY_STORAGE">
...
  <Options>
    <Option id="2">
      <PeriodFee basePeriod="MONTH" basePrice="100.00"
       factor="1.0" price="100.00"/>
      <UserAssignmentCosts basePeriod="MONTH" basePrice="0.00"
       factor="1.0" total ="0.00" price="0.00"/>
      <OptionCosts amount="100.00"/>
    </Option>
  </Options>
...
</Parameter>
```

RoleCosts

Specifies the costs for service roles.

If defined for the underlying application, roles can be used to grant specific privileges to different users. The roles are specified in the technical service definition as service roles. Service roles can be mapped to corresponding permissions in the application.

For each role, a price can be defined. This price is added to the base price per user in the cost calculation for a billing period.

The calculation of the charges for service roles depends on the cost calculation option which was chosen for the price model (see *PriceModel* on page 49 for details).

If the charges are calculated per time unit and the role assignment of a user is changed within a time unit, the affected time unit is charged pro rata. This means that the customer is charged exactly for the time each user role is assigned.

If the charges are calculated per time unit and a user with a specific role is removed from the subscription and assigned to it again with a different role in the same time unit, the customer is also charged for the time during which the user is not assigned to the subscription. This means that he is charged with the price for the first service role until the user is assigned to the subscription with the second service role.

A `RoleCosts` element is contained in a `UserAssignmentCosts` element (as subelement of the `PriceModel`, `Parameters`, or `Option` element).

A `RoleCosts` element has the following attribute:

`total` - (required, data type `decimal`) Total amount of costs for the service roles. The value is rounded to two decimal places.

For every service role, a `RoleCost` element is included in the `RoleCosts` element.

A `RoleCost` element has the following attributes:

- **`id`** - (required, data type `string`) ID of the service role.
- **`basePrice`** - (required, data type `decimal`) Recurring charge for the service role according to the price model.
- **`factor`** - (required, data type `decimal`) Factor used to calculate the costs for the service role. The factor is calculated as a fraction of the actual usage period. The recurring charge (`basePrice`) is multiplied with this factor to calculate the costs (`price`).
- **`price`** - (required, data type `decimal`) Costs for the service role. This value is calculated from the recurring charge (`basePrice`) multiplied with the factor (`factor`). The value is rounded to two decimal places.

Example:

```
<Parameter id="MEMORY_STORAGE">
...
  <RoleCosts total="0.00">
    <RoleCost basePrice="0.00" factor="0.4020087753882915"
     id="USER" price="0.00"/>
    <RoleCost basePrice="0.00" factor="0.8040175186678614"
     id="ADMIN" price="0.00"/>
  </RoleCosts>
...
</Parameter>
```

SteppedPrices

Specifies the stepped prices for a user assignment, event, or parameter.

Stepped prices allow for the definition of ranges for which different price factors apply. Step limits, i.e. the upper limits of ranges, can be set for:

- The **sum of the time units** users are assigned and work with a subscription in a billing period. For example, up to 10 hours one user is assigned to a subscription cost 10.00 € per hour, every additional hour the user is assigned costs 8.00 €.
- The **number of events** occurring in the usage of a subscription. For example, up to 10 file downloads cost 1.00 € per download, any additional download costs 0.50 €.
- **Values of numeric parameters**. For example, uploading up to 100 files costs 1.00 € per file, any additional upload costs 0.50 € per file.

Stepped prices are independent of any other price model elements.

A `SteppedPrices` element is contained in `UserAssignmentCosts` (as subelement of the `PriceModel` element), `Event`, and `PeriodFee` (as subelement of the `Parameters` element) elements.

A `SteppedPrices` element has the following attribute:

`amount` - (required, data type `decimal`) Summed up costs for all steps including the last one.

For every price step, a `SteppedPrice` element is included in a `SteppedPrices` element.

A `SteppedPrice` element has the following attributes:

- **`additionalPrice`** - (required, data type `decimal`) Summed up costs for the previous steps. The costs are calculated from the `limit`, `freeAmount` and `basePrice` attributes of the previous step (`(limit - freeAmount) * price`). The `additionalPrice` attribute of the first step always has a value of `0`. The value is rounded to two decimal places.
- **`basePrice`** - (required, data type `decimal`) Costs for the current step according to the price model.
- **`freeAmount`** - (required, data type `long`) Amount of units for the current step that are considered as a fixed discount, for example, the number of users that are free of charge. The value corresponds to the value of the `limit` attribute in the previous step. The `freeAmount` attribute of the first step always has a value of `0`.
- **`limit`** - (required, data type `string`) Step limit as defined in the price model.
- **`stepAmount`** - (optional, data type `decimal`) Summed up costs for the current step. These costs are calculated from the `basePrice` and `stepEntityCount` attributes of the current step. The value is rounded to two decimal places.
- **`stepEntityCount`** - (optional, data type `decimal`) Factor used to calculate the costs for the current step.

Example:

```
<PriceModel calculationMode="PRO_RATA" id="350001">
...
  <UserAssignmentCosts basePeriod="MONTH" factor="2.707940780619112"
 numberOfUsersTotal="4" price="1283.18">
    <SteppedPrices amount="1283.18">
      <SteppedPrice additionalPrice="0.00" basePrice="500.00"
       freeAmount="0" limit="2" stepAmount="1000.00"
 stepEntityCount="2"/>
      <SteppedPrice additionalPrice="1000.00" basePrice="400.00"
       freeAmount="2" limit="3" stepAmount="283.18"
       stepEntityCount="0.707940780619112"/>
      <SteppedPrice additionalPrice="1400.00" basePrice="300.00"
       freeAmount="3" limit="null" stepAmount="0.00"
 stepEntityCount="0"/>
    </SteppedPrices>
  </UserAssignmentCosts>
...
</PriceModel>
```

OverallCosts

Contains the total amount of the charges to be paid by a customer for all subscriptions in the current billing period. The costs are given in the currency specified in the price model.

If a discount was specified, the net amount of the costs is given in the `Discount` element (see *Discount* on page 62). If a VAT rate was defined, it is given in the `VAT` element (see *VAT* on page 61).

An `OverallCosts` element has the following attributes:

- **`netAmount`** - (required, data type `decimal`) Net costs after the net discount has been deducted from the original net costs (see *Discount* on page 62). The value is rounded to two decimal places.
- **`currency`** - (required, data type `string`) ISO code of the currency in which the costs are calculated.

- **grossAmount** - (required, data type `decimal`) Gross amount of the costs, calculated from the net costs (`netAmount`) plus VAT (see *VAT* on page 61). The value is rounded to two decimal places.

Example:

```
<BillingDetails key="10002" timezone="UTC+01:00">
...
<OverallCosts netAmount="900.00" currency="EUR" grossAmount="1053.00"/>

</OverallCosts>
</BillingDetails>
```

VAT

Specifies the VAT rate to be applied.

A supplier can define a basic VAT rate that applies by default to all prices for his customers. In addition to this basic VAT rate, country-specific or even customer-specific VAT rates can be defined. You can:

- Enable VAT rate support for your organization.
- Define a default VAT rate that applies to all prices for all customers.
- Define a country-specific VAT rate for every country where you want to sell your services.
- Define a customer-specific VAT rate, for example, in case a customer organization has a subsidiary located in another country than its parent organization.

The VAT rate settings have the following effects on the cost calculation for a customer:

- If VAT rate support is disabled, prices are calculated as net prices; no VAT is added to the overall costs.
- A customer-specific VAT rate takes priority over any default or country-specific VAT rate.
- The country-specific VAT rate for the country where the customer organization is located is applied to the cost calculation when no customer-specific VAT rate is defined.
- The default VAT rate is used in all other cases.

The VAT rate does not affect any price model elements. The calculated VAT amount is added to the overall costs and results in the gross price to be paid by a customer.

A `VAT` element is contained in the `OverallCosts` element.

A `VAT` element has the following attributes:

- **percent** - (required, data type `float`) VAT rate in percent, specified as a decimal number.
- **amount** - (required, data type `decimal`) Net amount of VAT to be added to the net costs (`netAmount` attribute of the `OverallCosts` element). The value is rounded to two decimal places.

Example:

```
<BillingDetails key="10002" timezone="UTC+01:00">
...
  <OverallCosts netAmount="900.00" currency="EUR" grossAmount="1053.00">
    <VAT percent="17.0" amount="153.00"/>
  </OverallCosts>
</BillingDetails>
```

Discount

Specifies the discount granted to the customer.

A discount can be defined for a customer which applies to all subscriptions of the customer to services. A discount may be valid as of the current or a future month. It can be restricted to a certain period of time. Before the time expires, the customer is notified by email so that he can react and contact the supplier.

The discount is defined as a percentage that is subtracted from the regular total price for a subscription. It is granted for all costs of a customer that incur in a billing period in which the discount is valid. It does not matter whether the discount is valid for the whole billing period or only a part of it.

A discount is completely independent of what a customer might purchase. If a discount is changed, the new discount is valid the next time the billing data is generated. Usually, a discount is only changed in agreement with the relevant customer.

A `Discount` element is contained in the `OverallCosts` element.

A `Discount` element has the following attributes:

- **`percent`** - (required, data type `float`) Percentage of costs to be deducted from the net costs, specified as a decimal number.
- **`discountNetAmount`** - (required, data type `decimal`) Net discount to be deducted from the original net costs (`netAmountBeforeDiscount`). The value is rounded to two decimal places.
- **`netAmountAfterDiscount`** - (required, data type `decimal`) Net costs after the net discount (`discountNetAmount`) has been deducted from the original net costs (`netAmountBeforeDiscount`). The value is rounded to two decimal places.
- **`netAmountBeforeDiscount`** - (required, data type `decimal`) Net costs before the net discount (`discountNetAmount`) has been deducted. The value is rounded to two decimal places.

Example:

```
<BillingDetails key="10002" timezone="UTC+01:00">
...
  <OverallCosts netAmount="900.00" currency="EUR" grossAmount="1053.00">
    <Discount percent="10.00" discountNetAmount="100.00"
     netAmountAfterDiscount="900.00"
     netAmountBeforeDiscount="1000.00" />
    <VAT percent="17.0" amount="153.00"/>
  </OverallCosts>
</BillingDetails>
```

Glossary

Administrator

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

Application

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

Billing System

A system responsible for calculating the charges for using a service.

Broker

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

Cloud

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

Cloud Computing

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

Customer

An organization which subscribes to one or more marketable services in ESCM 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 ESCM, 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 ESCM to provide their services to customers.

Marketplace Owner

An organization which holds a marketplace in ESCM, 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 ESCM.

Organization

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

Organizational Unit

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

OU Administrator

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

Payment Type

A specification of how a customer may pay for the usage of his subscriptions. The operator defines the payment types available in ESCM; 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. ESCM 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 ESCM. 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 ESCM for offering applications provisioned by technology providers to customers.

Technical Service

The representation of an application in ESCM. 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 ESCM.