A MITEL
PRODUCT
GUIDE

Unify OpenScape Fault Management

Unify OpenScape Fault Management V13, Service Workbench

Administrator Documentation
09/2023

A31003-F30D0-M100-01-76A9

Mitel

Notices

The information contained in this document is believed to be accurate in all respects but is not warranted by Mitel Europe Limited. The information is subject to change without notice and should not be construed in any way as a commitment by Mitel or any of its affiliates or subsidiaries. Mitel and its affiliates and subsidiaries assume no responsibility for any errors or omissions in this document. Revisions of this document or new editions of it may be issued to incorporate such changes. No part of this document can be reproduced or transmitted in any form or by any means - electronic or mechanical - for any purpose without written permission from Mitel Networks Corporation.

Trademarks

The trademarks, service marks, logos, and graphics (collectively “Trademarks”) appearing on Mitel’s Internet sites or in its publications are registered and unregistered trademarks of Mitel Networks Corporation (MNC) or its subsidiaries (collectively “Mitel), Unify Software and Solutions GmbH & Co. KG or its affiliates (collectively “Unify”) or others. Use of the Trademarks is prohibited without the express consent from Mitel and/or Unify. Please contact our legal department at iplegal@mitel.com for additional information. For a list of the worldwide Mitel and Unify registered trademarks, please refer to the website: http://www.mitel.com/trademarks.

Contents

1 Introduction

The Service Workbench is a compact helpdesk application. It offers support with the planning, coordination and monitoring of support activities.

The Service Workbench consists of three layers:

The bottom layer is a relational database where all information is stored.

The middle layer consists of an application server, business rules and the Service Workbench Web. The business rules are stored on the application server, which is also used as the web server for the Service Portal.

Ultimately, the client components of the Service Workbench and the Service Workbench Service Portal enable the users and customers to access the system.

Thanks to the use of a central database, each user can access the solutions database, which significantly reduces the administrative work required.

Furthermore, the Service Workbench is characterized by its platform independence since it is implemented in Java.

The following chapter contains instructions for installation, configuration and administration of the Service Workbench.

1.1 Target Groups

This manual is intended for all system administrators who are planning and executing a Service Workbench installation or who want to configure a Service Workbench.

The system administrator must have a basic knowledge of working with Windows and/or Unix. The system administrator should also be familiar with the concepts of a heterogeneous network environment.

1.2 Writing Conventions and Icons

The writing conventions used in the manual are explained below.

Style	Meaning
`Text`	Directory and file names as well as user entries in fields or via the command line are given in the `Courier` font. *Example:* Enter `/opt/application/bin`
Ctrl+C	Keys on the keyboard *Example:* Press Alt+F4.

Copy …	Operating steps are given in the Arial font.
File->Open ...	Successive selections of menu items and buttons are separated by an arrow. Example: Tab **Equipment=>Partitions**
Cancel	Buttons, menu names and menu items are displayed in **bold**. Example: Click **OK** to confirm changes.
Help	Cross-references to other chapters and emphases are given in *italics*. Example: see *Desktop User Guide*.

2 Installation

2.1 Requirements

The following components are required in order to successfully install the Service Workbench:

- Wildfly 8.2
- Oracle Java 1.8 or Oracle/OpenJDK Java 11
- Supported databases:
  - MySQL, Version 5.5.x or 5.6.x

The used database has to be configured as follows:

- The database tables must be in UTF-8 format.
- The database must not be case-sensitive.
- InnoDB must be used as the database engine.
- The TCP-port of the database has to be consistent. The port needs to match the database port configured in the standalone-full.xml, configuration described in *Section 2.2*.

If the Fault Management Extension is used, the following software components have to be installed and configured:

- OpenScape FM including the Navigation plugin.
- OpenScape Event Correlation Engine (ECE).

For the installation and configuration of the listed components, reference is made to the corresponding product documentation.

2.2 Installation of the Service Workbench

The installation of the Service Workbench is bundled with the installation of OpenScape FM. By following the installation steps in the *OpenScape FM Desktop* manual the Service Workbench will be installed automatically.

2.3 Updating an Existing Installation of the Service Workbench

Besides the installation of a new Service Workbench, an existing installation can be updated. The installation routine will automatically recognise if a Service Workbench is already installed. In this case, the values saved in the files in the Wildfly directory and the values in the database will be retained.

2.4 Switching the Database of the Service Workbench

The Service Workbench will initially use the Derby database provided by the OpenScape FM. To change the underlying database the following steps have to be performed:

- The new database needs to be configured according to the listed requirements in *Section 2.1*
- The appropriate jdbc driver for the database has to be included into the deployments directory of the Wildfly (`<install dir>/Wildfly/standalone/deployments/`).
- The database connection parameters have to be edited accordingly.

The database connection parameters can be configured in the file standalone-full.xml (located in `<install dir>/Wildfly/standalone/configuration/`). The determining entry for the database connection can be found under:

```
<server xmlns="urn:wildfly:domain:2.2">

<extensions>

<management>

<profile>

<subsystem xmlns="urn:wildfly:domain:datasources:2.0">

<datasources>
```

By default the configuration file provides exemplary configurations for MySQL databases which can be edited and commented in. If one of the example configurations is used, the active Derby configuration should be commented out.

2.5 First Login

Once the installation is complete a password has to be assigned to the administrator account **admin**. Subsequently a login to the Service Workbench is performed with this account.

3 Configuring the Application

When administrators login at the Service Workbench user interface, they have access to additional menu entries. One is the **Administration** menu, whose menu entries will be explained in more detail in the following chapters. Furthermore, only administrators have access to the **Delete** subitem in the **Dataset** menu and the **System Actions** tab of the **External Actions** configuration dialog in the menu **Configuration**.

Besides the ability to configure the application via the GUI, there are also extended options for configuring some items via the central server configuration. The configuration options available are described in *Section 3.5*.

3.1 Administration

All menu items described in this chapter are found under the **Administration** menu. All configuration dialogs, that can be opened via the different menu items, posses the following controls:

 **Add** - creates a new dataset.

 **Delete** - removes selected datasets.

 **Edit** - allows to change the values of a selected dataset. (Alternatively a dataset can be double-clicked.)

3.1.1 Categorization

The menu item **Categorization** enables the configuration of relevant Categories, which are needed for Tickets and Solutions. The menu item opens an overview, as a tree diagram, of the categories already available.

New Categories are created as subcategories of the Category which is currently selected.

Editing a Category allows to change the name of the Category or to set responsibilities.

Responsibilities define which Users or Groups are responsible for objects of the respective category. There is no limit on the number of Groups or Users that can be responsible for a particular category. The selection field **Default Group** allows to define a default group for the category. If a Ticket, which is assigned to the respective Category, is saved without being assigned a particular Group or User, the Default Group will automatically be set.

Note:
A previously used Category cannot be deleted, if the Category is subject of interdependencies within the system (e.g. between the Category and a Ticket).

If a previously used Category is changed, the altered category name will also appear in the corresponding Tickets and Solutions.

Note:
When a ticket is created in the Service Workbench Service Portal, only the top-level Categories are displayed to the Customer.

3.1.2 Users

In the Service Workbench Users represent the members of the support team working with the Service Workbench. The menu item Users allows to manage the Users of the Service Workbench. The respective dialog window displays an overview of all existing Users with login, surname, forename, role and status. Tickets, Workorders and Components can be assigned to a User. The column **Logged In** of the overview displays which Users are currently logged in to the Service Workbench.

When creating a new User, an input mask with the following configuration fields (* mandatory) will be opened:

Login: * Login name for logging into the system.

Password: * Password for logging into the system.

Surname: * Surname of the user.

Forename: * Forename of the user.

Role: Access rights of the user.

- **Administrator**: A User with this role has full access to the application.
  This role is excluded from data filtering should it be switched on. Users with this role can always see all data.
- **Staff**: A User with this role cannot administer the application or delete datasets.
- **Staff [R/O]**: This role is derived from the role *Staff*. A User with this role is restricted to read-only access to the datasets.
- **Fault Mgmt**: This special role is intended for Users who have to connect to the fault management system. A User with this role has no rights in the user interface. This role is only visible, if the fault management connector is licensed.
  This role is excluded from data filtering should it be switched on. Users with this role can always see all data
- **Manager**: The manager (superior) of the User, if there is one. The manager of a User is used in the escalation process and has the corresponding rights, provided these are configured.

Phone: Phone number of the User.

Mobile: Mobile phone number of the User.

Status: The current status of the employee: *active*, *inactive* and *discarded* for discarded employees (in which case, they can no longer login to the system).

E-mail: * E-mail address, as required for system notifications.
The e-mail address of predefined User should be changed to a valid e-mail address.

Create VCalender Entries: Can be used to switch on the receipt of calendar entries by e-mail. Calendar entries are currently used in the tickets for Followup Dates.

Access customer‘s personal data: If this is set, the User can access the page **Private** within Customer windows (see *User Guide*).

Preferred Language: * Defines the language to be used in system messages, e.g. e-mails, sent to the user.

Maximum Number of entries in the Overview*:* The number specified here is the maximum number of datasets to be shown in the Service Workbench Overview per region.

Group Membership*:* This table is used to assign a User to one or more Groups.

Location Membership*:* This table is used to display all Locations the current User is assigned to. A User can be assigned to a Location via the menu item **Administration->Locations** (see *Section 3.1.4).*

Rights*:* This table is used to restrict the rights of Users with the Role Staff. For Users with other Roles this table is disabled. In this table the access rights for the different object types, like Workorder or Company, can be set to read-only.

Note:
Deleting Users is not enabled because Users, as processors, are subjects of interdependencies within the system. Therefore it is advisable to set the Status of Users, which are no longer used, to *Discarded*. Such users will then be shown in red in the list of available Users.

3.1.3 Groups

The menu item **Groups** allows to manage the Groups of the Service Workbench. Groups consist of a number of Users. This has the advantage that, e.g. at the creation of a Ticket, a whole group of Users can be informed about a problem. For example the Group *IT-Berlin* could be used for all service employees who attend to the location *Berlin*.

In combination with active data filtering for Groups it is possible to restrict the access to certain data for whole groups of Users with the role *Staff* or *Staff R/O*. The data which is available for Users in the Service Workbench is then depending on Group. Data sets which are assigned to a Group, which the User is not part of, are not displayed for the User. This also applies for data sets which are not assigned to any Group. Data sets that are assigned to a certain User are always displayed for this User, regardless of group assignments. The group-filter is only available for Tickets, Workorders and Components. More about data filtering can be found in *Section 3.5.1*. Further information on data filtering can be found in the *User Guide*.

When creating a new Group a unique name for that Group has to be entered in the dialog. When editing the Group the name of the Group can be changed, but has to stay unique. Additionally a list of all members (Users) of the Group will be displayed when editing.

Note:
Groups can only be deleted, if they are not assigned to any Tickets or Workorders.

Edited Group names will be transferred to the client on restart.

3.1.4 Locations

Location Management is found under the item **Locations**. Here Locations can be created to which Users, Companies, Customers and Components can be assigned. If data filtering by location is activated, the data that can be seen by Users with the role *Staff* or *Staff [R/O]* can be limited. In this case, the data displayed for a User in the Service Workbench would be location dependent. Hence, data assigned to a location for which the User is not responsible would not be visible to that User. Datasets which are not assigned to a location are visible for all

Users. *Section 3.5.1* describes how to activate data filtering. Hence, which data is visible for a user depends on the configuration in Location Management. Further information on the subject of data filtering is provided in the user documentation.

Selecting the button **New** creates a new location.

Selecting an entry and clicking on the **Edit** button edits a location and makes an assignment.

The name of the location can be entered in the field **Location**. In the fields **Phone Prefix** and **Address** further information for the Location can be registered. In the **Localisation** section, various language versions for the location can be defined. If the user selects a language in a client for which no location localization has been defined, then the versions from the **Location** field will be offered to the user.

- To define another localization, the **New** button in the **Localisation** section has to be clicked.
- A new line will be added. Clicking on the **Language** column provides a selection list from which a language can be chosen.
- Then a localization can be entered in the **Name** column.

In the Assignments section, numerous Users, Companies, Customers and Components can be assigned to the current Location. Assignments can also be made in the respective dataset via the field **Location** or in the **Locations** section.

- To make an assignment, the relevant tab in the **Assignments** section should be used and the button **Edit Assignments** has to be clicked.
- A window opens with a list of all datasets of the respective type that have been created.
- The last column, **Assignment**, shows whether or not the dataset is assigned to the current location. By clicking on an **Assignment** checkbox, the dataset to the location can be assigned or the assignment can be undone.
- Clicking **OK** saves the changes and closes the window.

3.1.5 Holidays

The menu item **Holidays** opens the *Holiday Management* window. Holidays are not regarded as working days which will be considered by the calculation of escalation points. When creating a new **Holiday** entry, a name and date for the holiday have to be entered. The holiday will then be shown in an overview of all the holidays already entered in the system.

In Holiday Management, holidays can be defined and the year to which it applies can be specified in the overview **Holidays (unique)**. This holiday will then be valid only once – in the year specified.

In the overview **Holidays (recurrent)**, holidays can be defined without specifying a particular year. These are considered holidays in every year
(for example *New Year, 01.01.*).

In Holiday Management, a selected entry can be deleted via the **Delete** button or edited via the **Edit** button.

3.1.6 Working Hours

Escalation times are always calculated for defined working hours. The menu item **Working Hours** displays a configuration window where the default opening and closing times for each day of the week can be entered.

3.1.7 Status Transitions

Status transition refers to a change in the status of a ticket, for example from the status *New* to the status *Working*.

To prevent undesired ticket status changes but still provide the flexibility to select which status changes are allowed, individual status changes can be permitted/prohibited by ticking/unticking the boxes in the status transition matrix as appropriate.

Note:
The status transition *Solved* to *Forwarded* should only be removed, if the Service Workbench Customer Web component is not used. In the Customer Web, the customer has the opportunity to reject solved tickets and hence transfer them to the status *Forwarded*.

3.1.8 SLA (Service Level Agreement)

To each ticket an SLA with one or more service agreements can be assigned. These agreements define the individual reminder stages for achieving a particular status condition for the ticket in accordance with the priority level indicated. For example, it may be that a ticket with high priority has to be addressed within one hour and must be solved within three hours. In this case, the time (min.) can be set to *60* for the priority *High* and the status *Working* to *180* (min.) for the status *Solved*. These time periods are always calculated within the working hours specified.

SLAs can be configured using menu entry **SLA** in the **Administration** menu. The system will display an overview of all available SLAs and in this overview SLAs can be deleted using the **Delete** button or added using the **Add** button. To add an SLA, the name of the SLA has to be enter first, then the new SLA can be selected and edited via the **Edit** button. Another overview of the selected SLA opens at this point.

In this form the name of the SLA can be changed again, whereupon the system will display all agreements for that SLA. Each of these agreements comprises of a priority specification, status and time and these are all displayed in the overview. For a given priority and a particular status, a time (period) within which this status must be achieved can be specified. If the status value required is not achieved within the time specified, this will trigger an escalation (e-mails and diary entry).

In a further detailed view, the individual thresholds at which an escalation should be triggered can be set. This threshold value can be added via the **Add** button, viewed and edited via the **Edit** button and deleted via the **Delete** button.

If a threshold is set, it has to be set as a percentage of the time already specified. Furthermore, it can also be selected who should be notified when this threshold is reached.

For each agreement, there must be one threshold with the **Threshold Value [%]** "*100*". This entry is automatically created when an agreement is added and cannot be deleted. This is the only way to ensure an escalation is triggered if the defined status value is not achieved within the time specified. By default, the user will be notified when the 100% threshold is reached, but this can be edited.

Note:
An SLA can only be deleted if no ticket has been assigned to it yet.

The name of the SLA called *Standard*, can be changed, but not deleted. Unless otherwise specified, this Standard SLA is always used for new tickets.

The SLA called *FMDefault* is used as the default SLA for tickets automatically created by the fault management system. This item is only visible if the Fault Management feature has been activated.

If changes are made to an SLA that is being used in a ticket, the changes will not take effect until the escalation time has been re-calculated. This always happens when

- a different SLA or another priority is selected and saved in a ticket,
- an escalation time (threshold value) has been reached,
- the ticket passes from the status *Deferred* to the status *Working* and the monitored status is not *Solved* or *Closed*.

3.1.9 Itemized Lists

Itemized lists are the preset entries which, in many of the forms, appear as *drop-down lists*. Some of these itemized lists can be customized here. The overview list shows all configurable itemized lists. The **Edit** button can be used to view and edit an entry.

The detailed view of the selection list shows the current language setting, item value and order. A new selection element, together with language setting, can be added via the **Add** button. A selection value and an order number have to be entered for the new element; specifying a language setting is optional.

With respect to the order, the element with the lowest value appears at the top of the list and the element with the highest value appears at the bottom. If two elements have the same order number, the element that appears first in the database will be placed above the other.

In addition, a language setting can be specified. Using the **Add** button creates a new language entry. For this new entry, a language from the selection list has to be selected under the **Language** column and the localized text has to be entered in the column **Name** by double-clicking the *Name* column. If no localization is defined, the selection value itself will be shown for each language.

The **Delete** button can be used to delete a language setting.

The **Delete** button can be used to delete an element and the **Edit** button to edit an existing element.

Note:
Any changes made will not take effect until the client is restarted.

3.1.10 Component Catalog Types

Component catalog types allow the assignment of components to groups. The menu entry **Component Types** opens the management of the types. New types can be created here. Old types can be duplicated, edited or deleted.

A double click on a component type opens a configuration window for the selected type. In this window data can be added, edited or deleted for the type.

The selected icon is attached to all components assigned to the type. The fields **Has Software** and **Has Network Interfaces** define, whether the pages **Software** and **Network Interfaces** will be displayed for components of the selected type. The defined fields will be displayed as type fields for the respective components and can be filled with data.

The fields will be displayed for components on the page that has the name of the component catalog type.

3.1.11 Service Templates

The section **Services** of the Service Portal provides service functionalities in the form of **Service Tiles** to the Customers. Using the predefined Service Tile *Report Problem* for example enables Customers to create a new Ticket. The menu item **Service Templates** opens an overview window in which the Service Templates can be managed.

All currently configured Service Templates are displayed in the overview window. The column **Icon** displays the icon identifying the respective Service Template in the Service Workbench Client. The column **Name** contains the function description of the Service Template which is displayed in the Service Portal. The column **Status** determines whether the template will be displayed in the Service Portal. Only Service Templates of state *Active* are displayed in the Service Portal as Service Tiles.

When creating a new Service Template the Service Template Wizard will guide the administrator through the following steps:

Service Type: The *Standard Service* is used to interact with the workflow engine. Service Templates of type *Hyperlink* are used to create Service Tiles with any desired links.

Status: Templates of state *Active* are displayed in the Service Portal. The status *Deactivated* allows to create templates which initially are not to be displayed in the Service Portal.

Icon: This icon represents the Service Template in the client. However this is not the icon representing the template in the Service Portal. The icon to represent the template in the Service Portal can be configured by using the menu item **Load Picture...** in the context menu of the image area on the right hand side of the dialog.

Name: The description of the template will be displayed as the label of the corresponding Service Tile in the Service Portal.

Category: If different Service Tiles have the same category, a new tile for the category will be displayed in the Service Portal. This tile will open a window containing all Service Tiles of the respective category. The icon for this category tile has to be of added, as an image file with the same name, to the directory `<install dir>\Wildfly\standalone\configuration\jh-config.jar\resources\serviceportal`.

Description: The function description of the template will be displayed as the tooltip of the corresponding Service Tile in the Service Portal.

Hyperlink (only templates of type Hyperlink): The hyperlink to be invoked when the Service Tile is used.

The buttons **< Back** and **Next >** are used to navigate through the pages of the wizard. By using the button **Cancel** the wizard can be closed without creating a new Service Template. The button **Save and Close**, on the last page of the wizard, saves the configured template and closes the configuration wizard.

When editing an existing template, selected in the overview window, the configuration wizard will be opened pre-filled with the configured values of the template. Existing Service Templates can be deleted using the **Delete** button in the overview window.

By using the buttons **Move up** and **Move down** in the overview window the order in which the Service Tiles are displayed in the Service Portal can be determined.

Note:
The provided default templates can not be edited or deleted. If these Service Templates are not be displayed in the Service Portal, they can be set to the status *Deactivated*.

3.1.12 Cost Centers

Different datasets can be assigned to one cost center. Cost centers can be created using the *Cost Center* form.

The form is opened by selecting the menu item **Administration->Cost Centers**.

Using the button **New** creates a new Cost Center.

In addition to the **Name**, **Division** and **Account Number**, a default **Company** can be assigned. The company selected here, together with all customers assigned to this company, will then be assigned to this Cost Centre as the default Cost Centre unless otherwise specified.

If a Cost Center is assigned to a dataset, e.g. in the field **Cost Center** of a Customer form, the name of the Cost Center will be displayed. If the flag **Show full preview** is set for a Cost Center, the division and account number of the Cost Center will also be displayed.

The individual index cards contain an overview of the datasets to which this cost centre is assigned.

3.1.13 Fault Management Connector

If the Service Workbench has been licensed for fault management, it will have an interface to the fault management system. In this case, the menu item **Fault Management Connector** will be available. This menu item offers a dialogue where the default settings for this interface can be defined. Further information about the fault management interface and this dialogue are provided in *Chapter 4*.

3.1.14 Mail Monitor

Selecting the menu item **Administration->Mail Monitor** opens a window showing all the e-mails *not* sent by the system. This provides an overview of the latest emails that have not been sent and a chance to intervene, if necessary.

There are two status values for emails that have not been sent: *Unsent* and *Unsendable*

Unsent means that the email has been added to the queue for sending, but has not yet been processed.

Unsendable means that the email could not be sent due to a fault or error. The error messages can be viewed in the server log file (*Section 3.1.16*). Entries with the status *Unsendable* are displayed in red.

The context menu can be used on a selected entry to view detailed information about the email or to delete the entry.

3.1.15 Changelog

Selecting the menu item **Administration->Changelog** opens a window where changes that have been made in the system can be found. Changes in this respect are newly created datasets, edited datasets and deleted datasets.

The results list is displayed when the **Search** button is clicked. There are several filter criteria for such searches:

Object type: For selecting the object type, e.g. Ticket or Workorder

User: For selecting the User who made the change in the system

from: For selecting the from date. Only change entries made on or after this date will be displayed.

to: For selecting the to date. Only change entries made up to and on this date will be displayed.

Detailed information about an entry can be viewed in the results table by double-clicking it or opening its context menu. With respect to changes, the values before and after saving are of particular interest. Additionally the corresponding form can be opened using the context menu.

Note:
Change entries are only retained for a certain number of days depending on the configuration. The default configuration is 30 days. The variable *last edit days* needs to be added to the configuration file `<install dir>\Wildfly\standalone\configuration\jh-config.jar\resources\jhconfig.properties` to change the configuration. After editing the configuration file a restart of the Wildfly server is necessary.

3.1.16 Viewing the Server Log File

If any problems arise while working with the Service Workbench, the current server log file can be viewed. This can be done by selecting the menu entry **Administration->Show Server Log File**. A window will open displaying the current server log file (Wildfly). The **Search** field can be used to search for a particular term in this window. If the term is found, it will be highlighted in grey. To search for the next occurrence of the term, either the **Search Next** button or *F3* can be pressed. Pressing *Shift-F3* will start a backwards search.

3.2 Creating Templates for the Quick Ticket Function

Frequently recurring tickets can be saved as templates using the quick ticket function and re-used as templates for new tickets as and when required. The quick ticket functions are located in the menu **Extras** of the ticket form. New templates can be created using the menu item **Save as Ticket Draft**. The entered values of the current Ticket form are used for the new template. Process specific data, like the Follow Up Date or the diary entries, are exempted from this.

If templates have already been created, they can be can be applied to the currently open Ticket using the menu item **Use a Ticket Draft**. The displayed templates are ordered by their categories. Detailed information about the entries of the quick ticket is shown by tooltips. Using the administrator login, entries can also be deleted.

Note:
Only users with the role *Administrator* can create or delete templates.

An existing template can no longer be modified. Instead a new template has to be created. This saves the administrator a lot of time because some fields are already filled out with data and only a few fields need to be explicitly edited.

3.3 Creating Templates for the Quick Workorder Function

Similar to the quick ticket function (see *Section 3.2*), Workorders can also be saved and used as templates. Only administrators are permitted to create and delete such templates.

3.4 Creating Templates for the Quick Component Function

Similar to the quick ticket function (see *Section 3.2*), components can also be saved and used as templates. Only administrators are permitted to create and delete such templates.

3.5 Server Configuration

The central server configuration is used to configure general server properties. It is divided into the pages **Server/ Common**, **Outgoing mails**, **Service Portal**, **Synchronization**, **Workflow**, **Internal Tickets**, **Incoming mails**, **Relations**, **Logbook Rules** and **Server Messages**.

3.5.1 Server/Common

The page **Server/Common** contains several configuration areas.

The configuration area **Server** includes the following fields:

- **File date format and Preview**: This field allows the configuration of the date format displayed in automatically created Reports

- **Client callback port**: The callback port used for automatic updates of clients in case of data changes (e.g. by the Workflow).

The area **Client-SingleSignOn** provides the following configuration fields concerning the client login:

- **Client SSO active**: If this flag is set, a domain account can be used to login to the client. The account has to exist as a User in the Service Workbench. The password of the internal user is not used for the authentication.
- **Direct Login**: If this flag is set in addition to the flag **Client SSO active**, at the start of the client a login will automatically be performed using the system account
- **Domains**: The authentication is performed against the entered (comma separated) domains. The authentication is attempted successively once against each domain until an authentication is valid.
- **KDC**: Key Distribution Center server.

The area **OSFM Integration** is used to enter the connection data to the OpenScape FM (OSFM) Server:

- **Integration active**: If this flag is set, the license information is directly read from the OSFM and the Service Workbench client can be started per SingleSignOn directly from the OSFM.
- **Server**: Address of the OpenScape FM (OSFM) server.
- **Port**: Port of the OpenScape FM (OSFM) server.
- **Login**: The OSFM user account which is used by the Service Workbench for communication purposes with the OSFM.
- **Password**: The corresponding password for the *Login*.
- **Web-Port**: Port number for the web access.

The area **Intervals** provides settings for the interval length in seconds in which the server will perform searches. If certain searches should not be performed, their field entries need to be set to *-1*. This will automatically be performed, if a field is deactivated.:

- **Escalation**: The interval in which the server checks whether an escalation point for Tickets or Workorders has been reached.
- **Reescalation Tickets**: If the escalation of a Ticket is complete and no revision was made since the last escalation, the entered interval determines when the last escalation message will be send again.
- **Reescalation Workorders**: If the escalation of a Workorder is complete and no revision was made since the last escalation, the entered interval determines when the last escalation message will be send again.
- **DB-Cleaning***:* The interval in which the server checks whether old database entries for Mails and Tickets are present.
- **Component Check**: The interval in which the server checks for existing components with expiration warnings.
- **Mailing**: The interval in which the server will deliver all outgoing mails.

By selecting a data filter in the area **Data filter implementation** the information visible for a user can be restricted. This makes it is possible to restrict Users to only see datasets which are assigned to a respective Location or Group of which the User is a member. Users with the role Administrator always have access to all datasets. Datasets which are directly assigned to a User are always visible for this User. The possible data filtering values are:

- **None**: No data filtering.
- **Location**: A data filtering based on the assigned Locations of datasets will be activated.
- **Group**: A data filtering based on the assigned Groups of datasets will be activated.

The area **Tickets** contains the following selection fields:

- **Close automatically**: A solved ticket will be closed by the system after a certain time, even without a confirmation by the user. The time interval (in hours) can be configured if the checkbox is marked.
- **Close when workorders completed**: When the last Workorder of a Ticket is completed, it will be asked whether the Ticket should also be closed.
- **Delete automatically**: Closed tickets will be deleted from the database after a defined time. The time interval (in days) can be configured if the checkbox is marked.
- **„Send mails to customer“ initially set***:* If this flag is set, the checkbox, determining whether Customers should be informed about status changes of a Ticket, is initially set in all Tickets.
- **Increase worktime automatically**: If this flag is set, the worktime of a Ticket will automatically be updated every minute.
- **Allow worktime edit**: If this flag is set, the worktime of a Ticket can be edited by a User.
- **Auto component prefix**: If this flag is checked, the names of the Components assigned to a Ticket will automatically be set as (comma separated) prefixes in the short description of the Ticket.

The area **Workorders** contains the following selection fields:

- **Send escalation mail to manager**: If this flag is checked, an additional escalation message is send to the manager of the User or Group that is responsible for the escalated dataset,
- **Auto component prefix**: If this flag is checked, the names of the Components assigned to a Workorder will automatically be set as (comma separated) prefixes in the short description of the Workorder.
- **Increase worktime automatically**: If this flag is set, the worktime of a Workorder will automatically be updated every minute.
- **Allow worktime edit**: If this flag is set, the worktime of a Workorder can be edited by a User.

The area **Changelogs** contains the following selection fields:

- **Changelog storage**: This field determines the time period (in days) for which changes made to the datasets will be stored in the changelog.
- **Ticket-Workingtime storage**: This field determines the time period (in days) for which the workingtimes of datasets will be stored.

The are **Attachments** contains the following fields:

- **Maximum size**: This field determines the maximum permitted size of attachments (in MB per attachment).
- **Maximum number**: This field determines the maximum permitted amount of attachments.

The area **CTI** contains the following fields:

- **CTI active**: This field activates the Computer Telephony Integration.

The area **Other** contains the following fields:

- **Remember Table Sorting**: If this flag is set, the order by which the tables are sorted will be saved.
- **Report „Security Software“ enabled**: If this flag is set, the integrated report Security Software will be activated.
- **Mail notification to old user**: If this flag is set, a message will be send to the previously assigned User of a dataset if the dataset is assigned to another User.
- **Check open forms**: If this flag is set, the server will check if a dataset is already opened in another client when the dataset is opened.
- **Search - Show full result**: If this flag is set, all datasets will initially be displayed when a search is opened.
- **Create solutions with ticket details**: If this flag is set the short and detailed descriptions of Tickets will be adopted for newly created Solutions.

The area **Table Model** contains the following fields:

- **Chunk size**: in this field the size of the table model chunks can be configured. It is advisable not to change the default value.

3.5.2 Outgoing Mails

The page **Outgoing mails** is used to define settings for the automatic delivery of emails. A sender email address can be entered which can be used by the recipient to answer the emails. The field **Signature** is used to define an additional text that will be attached to all emails that are automatically created. The **Hold-Back Time** defines how many days a sent email should be stored.

In addition the area **Tickets** can be used to configure settings for Tickets that are created automatically (e.g. if a recipient is unknown). Default values for the priority can be assigned.

3.5.3 Service Portal

The page **Service Portal** is used to configure the web interface which is used by the customers to connect to the ticket system.

The area **ServicePortal general** is used to configure the Service Portal Interface.

The header of the Service Portal can be modified. The text, text color and background color of the header can be changed. If no header should be displayed the checkbox *Show Header* has to be unchecked.

The other selection fields define whether Customers have the option to:

- **Show „Change password“**: If this flag is set, the Customers are able to change their passwords.
- **Show „Forgot password“**: If this flag is set, the Customers are able to request a new password on the login page of the Service Portal.
- **Show Comments**: If this flag is set, the Customers have access to the tab Comments when viewing a Ticket.
- **Select category for new tickets**: If this flag is set, the Customers are able to select a Category when creating a Ticket.
- **Default Category**: If Categories are not selectable during Ticket creation, the defined Category will be assigned to the Tickets. Subcategories can be entered using „ | “ as a delimiter.
- **Select component for new tickets**: If this flag is set, the Customers are able to select a Component when creating Ticket.
- **Component is required**: If this flag is set, Customers (for whom at least one Component is available) are required to select a Component when creating a Ticket.
- **Select urgency for new tickets**: If this flag is set, the Customers are able to select an urgency when creating a Ticket.
- **Default Urgency**: If urgencies are not selectable during Ticket creation, the defined urgency will be assigned to the Tickets.
- **Select group for new tickets**: If this flag is set, the Customers are able to select a Group when creating a Ticket.
- **Select attachment for new tickets**: If this flag is set, the Customers are able to add attachments when creating a Ticket.

In addition the 10 extension fields for Tickets can be displayed during Ticket creation.

In the area **Object Fields** the presentation of different object types in the Service Portal can be configured:

- **Tickets**
  - **Show empty ticket fields in table preview**: If this flag is set, the tooltip for a Ticket will also display fields for which no value is defined in the Ticket.
  - **Table Ticket table tooltip fields**: This table defines which fields are to be displayed in the table tooltip for a Ticket.
  - **Show empty ticket fields**: If this flag is set, fields for which no value is defined will also be displayed when a Ticket is opened.
  - **Table Ticket fields**: This table defines which fields are to be displayed when a Ticket is opened.
- **Workorder**

- **Show empty workorder fields in table preview**: If this flag is set, the tooltip for a Workorder will also display fields for which no value is defined in the Workorder.
- **Table Workorder table tooltip fields**: This table defines which fields are to be displayed in the table tooltip for a Workorder.
- **Show empty workorder fields**: If this flag is set, fields for which no value is defined will also be displayed when a Workorder is opened.
- **Table Workorder fields**: This table defines which fields are to be displayed when a Workorder is opened.

- **Customer**
  - **Show empty customer quick info fields**: If this flag is set, the customer information of the logged in Customer will also display fields for which no value is defined in the Customer.
  - **Table Customer info fields**: This table defines which fields are to be displayed in the customer information of the logged in Customer.
  - **Show empty customer fields**: If this flag is set, fields for which no value is defined will also be displayed when a Customer info is opened.
  - **Table Customer search fields**: In the Service Portal it is possible to search for persons (Customers). This table defines which Customer fields are taken into account for the search and are to be displayed in the result list. All matching Customers will be displayed. The only exception being Customers added to the black list located in `<install dir>\Wildfly\standalone\configuration\jh-config.jar\resources\blacklist_customer_web.txt`. Modifications to this file only take effect after a server restart.
- **Components**
  - **Show empty component quick info fields**: If this flag is set, fields for which no value is defined will also be displayed when the Component quick info for the logged in Customer is opened.
  - **Table Component info fields**: This table defines which fields are to be displayed when the Component quick info for the logged in Customer is opened.

The area **Authentication** holds the configuration settings for the authentication method. The following alternatives are available:

- Loginname and Password from the customer data (Default)
- Login with the Windows user data (NTLM). With NTLM authentication it is possible to activate SingleSignOn (needs the Microsoft Internet Explorer).
- Login using an LDAP server as e.g. Microsoft Active Directory.

3.5.4 Synchronization

The page **Synchronization** is used to configure the synchronization of the Wildfly database with other databases.

If the **OSFM-Sync** is activated, the foreign keys (to identify the respective object in the Fault Management) are synchronized into the Fault Management to allow back navigation.

3.5.5 Workflow

The page **Workflow** is used to activate the connection to the Workflow Manager and to define the connection parameters for the Workflow Engine.

3.5.6 Internal Tickets

The page **Internal Tickets** can be used to define the settings for internal tickets in the scopes *Licenses*, *Contracts* and *Components*.

For example if an expiring Contract exists, a Ticket with the configured settings will automatically be created to draw attention to this problem.

The expiration warning tickets react to entries in the field **Expiration Warning**, not to entries in the field **Expiration Date**.

3.5.7 Incoming Mails

The page **Incoming Mail** is used to define the parameters for the reception of emails (Mail2Ticket-Interface). Besides the Server, Port and Request Protocol the information to connect to the mail server and the mail folder can be entered here.

Within the area **Tickets/Workorders** regular expressions can be entered to define the assignment of incoming mails to already existing tickets or workorders.

Within the area **Templates** automatic settings can be attached to email addresses. This enables the automatic assignment to categories, groups and priorities for emails of certain customers or from certain email addresses. The field **Response Mail** defines, whether a receipt should be send.

3.5.8 Relations

The page **Relations** allows the creation of new pages for specific object types, like e.g. Tickets. The page lists the assigned entries. To create a new relation, the button **Add** must be pressed. After this the **Source** and a **Target** have to be assigned for the relation.

If the **Source** is set to *Ticket* and the **Target** to *Component* then for each ticket an area with the effected components will be displayed.

The **Connectivity** defines whether only *One* or *Many* components can be listed per Ticket.

3.5.9 Logbook Rules

The page **Logbook Rules** allows the definition of rules for the creation of logbook entries for the different object types, like Tickets or Customers. These rules specify at which points Users have to create new logbook entries.

The rules always refer to specific fields of the corresponding datasets. For example a rule can be created which ensures that every time a Ticket is assigned to another User a new logbook entry has to be written. It is not possible to create rules which are valid for different object types. For example, a rule stating that specific Users need to write logbook entries for every change, regardless of whether the change was made in a Ticket or a Workorder is not possible.

Different rule sets concerning a specific object type (e.g. all rule sets created for Tickets) are linked with OR-operations. The individual rules of a rule set are always linked with AND-operations.

New rule sets can be created for a selected object type using the **Add** button. To add rules to the rule set the rule set has to be selected when using the **Add** button.

Deleting a rule set will also delete the individual rules of the rule set. Selecting an object type and using the **Delete** button will delete all rule sets for that object type.

A rule is configured by the following values:

- **Fields**: The field to which the rule will refer to. A selection list provides all available fields of the specific object type.
- **Operations**: The operation determines for which action, performed for the field, the rule will apply. The following operations are available:
  - **new**: The rule will take effect, when the specified field was empty and a value is stored in it.
  - **change**: The rule will take effect, when the value of the specified field is changed.
  - **delete**: The rule will take effect, when the value of the specified field is deleted.
  - **value**: The rule will take effect, when the value of the specified field is changed from the configured *Old value* to *New value*.
- **Old value** and **New value** (only for operation *value*): If the operation is set to *value*, the old value and new value that need to be set, for the rule to take effect, can be configured. Configuring a value as * implies that the specified field can contain any non-empty value for the rule to take effect.

3.5.10 Server Messages

In the tab **Server Messages** the messages which are send or displayed by the Service Workbench server can be configured. For example the templates for mails send by the server can be modified. Therefore the key for the respective action has to be selected in the list on the left. The corresponding template can then be configured on the right hand side.

3.6 Configuring Reports

3.6.1 General Information

Service Workbench reports use the Jasper Reports reporting framework, which is available free of charge (`http://jasperforge.org/projects/jasperreports`). The reports themselves can be created using a graphical report editor and then integrated into the Service Workbench.

One free graphical report editor is *iReport*, which can be downloaded free of charge from: `http://jasperforge.org/projects/ireport`

Alternatively the *Jasper Assistant* can be used, which however is not free of charge. This graphical report editor can be downloaded from the following website:

`http://www.jasperassistant.com`

The Service Workbench uses version 4.0.2 of Jasper Reports.

Reports created by the editors generally have the file extension `.jasper` and can be integrated into the Service Workbench report directory simply by copying. New reports can be integrated during system operation without the need to reboot.

The relevant manufacturer's documentation should be referred for instructions on how to use the report editors.

3.6.2 Integrating Reports

The Service Workbench provides different directories for reports on the server. There are directories for licence reports, ticket reports, automatically generated reports and reports for the Service portal. These directories are where the reports supplied as standard are installed. New report files can be easily copied to these directories and will then appear in the respective report menu of the Service Workbench. The administrator defines the parent directory for these directories during installation.

The reports stored in the directory `CustomerReports` are not available in the Service Workbench client, but can be viewed in the Service Portal. The reports are only available for explicitly selected Customers. Reports can be released for Customers using the menu item **Extras->Report Assignment**.

To ensure the Service Workbench recognizes the reports, an XML file has to be created in this directory for each report. The XML file should have the same name as the report file, but with the file extension `.xml`.

The XML file is structured as per the following example:

```
<?xml version="1.0" encoding="ISO-8859-1"?>

<!-- Unterstützte Datentypen: java.lang.String, java.lang.Integer, java.lang.Float,
java.util.List, java.util.Date, java.lang.Boolean-->
```

```
<Report>
   <ReportName file="test.jasper">
   <Label lang="en" label="Test Report in English"/>
   <Label lang="de" label="Test Report in Deutsch"/>
   </ReportName>

   <ReportParameter default="%">
      <Name>param1</Name>
      <Type>java.lang.String</Type>
      <Label lang="en" label="Label name of the parameter" />
      <Label lang="de" label="Anzeigename des Parameters" />
   </ReportParameter>

   <ReportParameter>
      <Name>param2</Name>
      <Type>java.lang.String</Type>
      <Label lang="en" label="Label name as string" />
      <Label lang="de" label="Anzeigename als String" />
   </ReportParameter>

   <ReportParameter default="345">
      <Name>Integer</Name>
      <Type>java.lang.Integer</Type>
      <Label lang="en" label="Parameter as Integer" />
      <Label lang="de" label="Parameter als Integer" />
   </ReportParameter>

   <ReportParameter default="123.456">
      <Name>Float</Name>
      <Type>java.lang.Float</Type>
      <Label lang="en" label="Testparameter as Float" />
      <Label lang="de" label="Testparameter als Float" />
   </ReportParameter>

   <ReportParameter default="01.02.2008">
      <Name>Date</Name>
      <Type>java.util.Date</Type>`
      <Label lang="en" label="Testparameter as Date" />
      <Label lang="de" label="Testparameter als Datum" />
   </ReportParameter>

   <ReportParameter default="31.03.2008">
      <Name>Date</Name>
      <Type endofday="true">java.util.Date</Type>
      <Label lang="en" label="Testparameter as Date EOD" />
      <Label lang="de" label="Testparameter als Datum EDT" />
   </ReportParameter>
```

```
   <ReportParameter default="three">
      <Name>ListAsString</Name>
      <Type>java.util.List</Type>
      <Label lang="en" label="Testparameter as List" />
      <Label lang="de" label="Testparameter als Liste" />
      <List>
         <Item>one</Item>
         <Item>two</Item>
         <Item>three</Item>
         <Item>four</Item>
         <Item>five</Item>
      </List>
   </ReportParameter>

</Report>
```

The line `<ReportName file="test.jasper">Test Report</ReportName>` is used for storing the name of the report file and the associated display name for report selection in the Service Workbench report menu.

Next, report parameters can be defined, if required. The number of report parameters is unlimited; but they can also be omitted.

The element `<ReportParameter default="%">` defines a new report parameter with "`%`" as the default setting. This setting is displayed as the default value when the report is opened in the Service Workbench report menu.

In this respect, the following three sub-elements are always required:

- `<Name>param1</Name>`
  Definition of the report parameter name (as per the definition in the Jasper Report).
- `<Type>java.lang.String</Type>`
  Definition of the report parameter data type. The following data types are currently supported:
  - `java.lang.String`
  - `java.lang.Integer`
  - `java.lang.Float`
  - `java.util.List`
  - `java.util.Date`
  - `java.lang.Boolean`
- `<Label lang="en">Displayed name of the parameters in english</Label>`
  Localized display name of the report parameter. The language ("`de`" or "`en`") is specified with `lang`.

If the data type should be `java.util.List`, then the corresponding list elements can also be defined. To this end, another `<List>` element can be attached, which in turn contains the individual list elements in `<Item>` sub-elements.

If the data type should be `java.util.Date`, then an additional attribute `endofday="true"` can be assigned. The date is then provided with an end-of-day time upon parameter transfer; otherwise, the time is set to *00:00*.

Every report created for the Service Workbench should have the report parameter **ReportPath**. The Service Workbench always fills out this parameter with the respective path to the report files directory. The parameter is then transferred on start-up. The **ReportPath** parameter must have the data type `java.lang.String`.

The logo at the top of the sample report is currently loaded from the graphics file `ReportHeader.jpg`. This graphics file can be changed to suit the requirements, but it should not exceed a size of 1240 x 308 pixels.

If an XML file is corrupt, the associated report will not appear in the Service Workbench report menu and a corresponding error message will be sent to the Wildfly server log file.

3.6.3 Automatically Generated Reports

The integrated reports can also be automatically created in the background according to a given time frame. These reports are stored in a shared directory on the (Wildfly) server and can also be sent as emails.

The settings for the reports to be automatically generated are applied in the file `autoreports.xml` in the Wildfly installation directory. This file is located in the following subdirectory: `<Wildfly-Install-Directory>\server\default\deploy\jh-config.jar\resources\`

The XML file is structured as per the following example:

```
<?xml version="1.0" encoding="ISO-8859-1"?>

<autoreports>
   <report>
      <name>testname1</name>
      <file>path-to-testreport.jasper</file>
      <format>PDF</format>
      <email>mail-receiver@myhost.com</email>
      <lang>de</lang>
      <timer unit="hour">24</timer>

      <parameter>
         <type>java.lang.String</type>
         <name>param1</name>
         <value>test1</value>
      </parameter>

      <parameter>
         <type>java.langInteger</type>
         <name>param2</name>
         <value>999</value>
      </parameter>
```

```
      <parameter>
         <type endofday="false">java.util.Date</type>
         <name>param3</name>
         <value>01.01.2008</value>
      </parameter>

   </report>
</autoreports>
```

Any number of reports to be automatically generated can be added under the parent element `<autoreports>`. In this respect, each report is defined by a separate sub-element `<report>` (to safeguard the XML format, the associated schema file `autoreports.xsd` is also stored in the same directory).

The following settings/elements must be added underneath:

- `<name>testname</name>`
  Descriptive name of the report; this must be unique.
- `<file>path-to-testreport.jasper</file>`
  Path to the report file.
- `<format>PDF</format>`
  Output format of the report (currently PDF only).
- `<email>mail-receiver@myhost.com</email>`
  The email address, to which this report should be sent.
- `<email-subject>Test-Report</email-subject>`
  The subject line of the email.
- `<email-body>Test-Report</email-body>`
  The body of the email.
- `<lang>en</lang>`
  The language version of the report to be sent.
- `<timer unit="hour">24</timer>`
  Indication of when the report should be automatically triggered. The following values are currently supported in the unit attribute:
  - `hour`
    The report is launched at intervals of however many hours are indicated by the value. Reports are always generated on the hour.
  - `day`
    The report is launched at intervals of however many days are indicated by the value.
  - `weekday`
    The report is always launched on the day of the week specified in the value. The values available are “`mon`”, “`tue`”, “`wed`”, “`thu`”, “`fri`”, “`sat`” and “`sun`”.
  - `monthday`
    The report is launched on the day of the month specified in the value.

- `<parameter>`
  The parameters for the report with the following sub-elements:
- `type`
  The Java data type, e.g. `java.lang.String`.
- `name`
  The name of the parameter in the report.
- `value`
  The value of the parameter. For the data type `java.util.Date`, the following variables can also be used: `CURRENT_DATE, CURRENT_MONTH_START, CURRENT_MONTH_END, CURRENT_YEAR_START, CURRENT_YEAR_END`

3.7 Deleting Datasets

Datasets can only be deleted by Users with the role *Administrator*. When a dataset is deleted different checks are performed for the different object types.

3.7.1 Ticket

A Ticket can only be deleted, if no Workorders are assigned to it. When a Ticket is deleted, all associated diary entries will also be deleted from the system.

3.7.2 Customer

A Customer can only be deleted from the database, if no Ticket is assigned to that Customer.

If the Customer that should be deleted is a key contact at a Company, the link with this Customer will also be lost. This will also happen if Components are still assigned to this Customer, whether it be a User or a Contact.

3.7.3 Company

A Company can only be deleted, if no Customers or Subcompanies are assigned to it and there are no links inComponents/Component Catalogs as Manufacturer or Supplier.

3.7.4 Component Catalog

A Component Catalog can only be deleted, if no Components are assigned to it.

3.7.5 Component

A Component can only be deleted, if no Subcomponents, Workorders or Tickets are assigned to it.

3.7.6 Solution

A Solution can only be deleted, if it is not being used in a Ticket.

3.7.7 Workorders

There are no conditions with respect to deleting a Workorder.

3.8 General Information

3.8.1 Setting the Clock on the Server

It has to be kept in mind that the Wildfly server needs to be restarted, if the time of the server, on which the Service Workbench server is installed, is changed.

3.8.2 Changing the Labels for the Client

The labels of the client can be changed to suit the requirements. Generally, this refers to the 10 extension fields of the individual object types, such as Tickets or Workorders. The labels for the 10 extension fields are pre-configured, but can be customized.

The following files contain the relevant defaults:

```
<Wildfly>\server\default\deploy\jh-config.jar\resources\clientlabels.properties
<Wildfly>\server\default\deploy\jh-config.jar\resources\clientlabels_en.properties
```

The first file contains the labels that are used if the client is set to German. The second file is used if the client is set to English.

Each line contains an entry for a label.
`Format: Key = Value`

Lines with a `#` at the start are comments and are ignored by the system.

If these files have been edited, the Wildfly server need to be restarted.

3.8.3 Editing Files Using the Client

Files which are included in the configuration file `<install dir>\Wildfly\standalone\configuration\jh-config.jar\resources\configfiles4client.conf` can be edited using the submenu **Server Configuration** of the **Administration** menu.

Modifications of the file `configfiles4client.conf` only take effect after a restart of the Wildfly server.

The file is structured in the following way:

```
# Files to display as editor entry in admin menu
exampleFile.properties
```

4 Connection to the Fault Management System

In addition to the manual assessment of problems, it is possible to automatically create Tickets with the OpenScape FM.

For the automatic creation of Tickets one of the following plugins is needed for the Service Workbench:

- Fault Management Connector
- Event Correlation Engine (ECE)

With these plugins, events detected by the OSFM can be directly provided to service technicians as Tickets. In addition to the problem description these Tickets may contain further information.

This chapter deals with extensions to the Service Workbench that relate to the automatic creation of Tickets by the fault management (FM) system (OpenScape FM).

4.1 Fault Management Connector

Extending the Service Workbench with the Fault Management Connector allows Tickets to be automatically generated by the FM.

Via the Service Workbench client, a User, a Workgroup, a Customer, an SLA and a Category can be defined. Unless otherwise specified, these are then automatically assigned upon Ticket creation.

In addition to the Tickets themselves, it is also possible to automatically generate Component Catalogs and Components, to which the Tickets created are assigned. Furthermore, an associated Workorder can also be generated for a Ticket.

Various elements of the graphical interface associated with the connection of the Fault Management Connector are described below in more detail.

Information how to configure the data that will be generated is provided in the documentation for the Fault Management Connector.

4.1.1 Access Rights

Just like manually created messages, Tickets that were created automatically by the fault management system can be viewed and edited by Users with the role of *Staff* or *Administrator*.

However, automatic creation of Tickets by the fault management system is only permissible for Users with the role *Fault Mgmt*. Only this role has the right to create Tickets automatically. This role has no other rights whatsoever, not even the right to open the graphical interface of the Service Workbench.

During installation of the Service Workbench, a User is automatically set up with the login *faultMgmt* and the role *Fault Mgmt*. Like any other user, this can be modified by a User administrator.

4.1.2 Defaults

Clicking the menu entry **Administration->Fault Management Connector** opens a configuration window where a range of default values for the automatic creation of Tickets can be defined.

The individual input fields have the following meanings:

- **SLA**:
  In this field, the selection menu can be used to define a default SLA for automatically created Tickets.
  For the Tickets to be created successfully, agreements for this SLA have to be defined (see *Section 3.1.8*)
- **User**:
  In this field, the selection menu can be used to define a default User.
  All automatically generated Tickets are assigned to this User, if the ticket is not assigned with a Component for which a User has already been defined or which is a part of another Component for which a User has already been defined.
- **Group**:
  In this field, the selection menu can be used to define a default Workgroup.
  All automatically generated Tickets are assigned to this group, if the Ticket is not assigned with a Component for which a Group has already been defined or which is a part of another Component for which a Group has already been defined.
- **Customer**:
  The *Search* entry of the menu located next to the input field can be used to define a default Customer for automatically created Tickets.
  This Customer is assigned to all automatically generated Tickets that are assigned with a Component for which no Customer has already been defined or which is a part of another Component for which no Customer has already been defined.
- **Category**:
  In this field, the Category Tree can used to define a default Category.
  This Category is assigned to all automatically generated Tickets.

4.2 Event Correlation Engine (ECE)

Using the ECE, new Tickets can automatically be created in the Service Workbench for events which occurred in the OSFM. The Tickets are created directly, without the need for a special User.

Besides the creation of Tickets, already existing Tickets can be modified or read and displayed in the OSFM. Components, Workorders and new logbook entries can be created automatically.

More information about the range of functions and the configuration of the ECE plugin for the Service Workbench can be found in the documentation of the Event Correlation Engine plugin.

Index