A MITEL
PRODUCT
GUIDE

Mitel OpenScape Fault Management

Mitel OpenScape Fault Management V13 Service Workbench

User Guide
10/2025

A31003-F30D0-U119-01-7619

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 Preface

This chapter discusses the following aspects:

- Goal and audience for this User Guide
- Terminology
- Structure of this User Guide
- The conventions used in this User Guide

1.1 Purpose

This User Guide describes the functions for standard users within the **Service Workbench**, an extension component for the **OpenScape Fault Management**.

1.2 Audience

This guide addresses users who want to learn how to use the **Service Workbench** for the OpenScape Fault Management. The reader should know how to use the OpenScape Fault Management. A detailed description of this program can be found in the *OpenScape Desktop User Guide*.

How Administrator functions are used is described in the separate *Service Workbench Administrator Documentation*.

1.3 Organization of this Guide

This guide is organized as follows:

- *Chapter 2, "Introduction"* offers a functional overview of the Service Workbench.
- *Chapter 3, "Overview"* gives an overview about the terms used in this user guide and shows examples for standard use cases.
- *Chapter 4, "First Steps"* describes how the Service Workbench is started and explains the content of the initial page.
- *Chapter 5, "Tickets"* describes the functions and data structures of Tickets.
- *Chapter 6, "Workorders"* describes the functions and data structures of Workorders.
- *Chapter 7, "Solutions"* describes the functions and data structures of Solutions.
- *Chapter 8, "Inventory"* describes the inventory based functions and data structures for Component Catalogs, Components, Software, Rooms and Ports.

- *Chapter 9, “Contacts”* describes the contact based functions and data structures for Customers and Companies.
- *Chapter 10, “Actions”* describes the handling of server and client actions that can be started to perform regular tasks.
- *Chapter 11, “Relation Viewer”* describes the Relation Viewer that is a part of most datasets.
- *Chapter 12, “General Functions”* describes some general functions like the handling of datasets, uplink viewers or export and printing functions.
- *Chapter 13, “Administration”* describes some administrative functions.
- *Chapter 14, “Fault Management”* describes functions related to the automatic creation of Tickets and Components by the OpenScape Fault Management.
- *Chapter 14, “Service Portal”* describes the handling and the functions of the Service Portal. A web interface that enables Customers to submit and review their Tickets.

1.4 Used Conventions

The following font conventions are used in this document:

Bold Font: indicates that a word is a new or important term. Bold is also used for buttons, menu names and item names:

Example: **JavaDB**. The **Cancel** button.

Bold Computer Font: indicates data to be entered by the user:

Example: **`java`**.

Computer Font: indicates computer output, including UNIX prompts, an explicit directory or a file name:

Example: `prompt%`.

Italics: indicates a reference to another manual or to a different section within the current manual:

Example: *OpenScape Desktop User Guide.*

Italics type is also used for emphasis:

Example: *All* users will be affected.

1.5 Terminology

- **OpenScape FM** stands for OpenScape Fault Management.

2 Introduction

The handling of incidents is usually performed in a number of worksteps that are separated in time: problem analysis, various actions to remove the problem and final tests. These are often performed by different staff members. The Service Workbench allows a coordinated error removal. Especially when several staff members are involved in the solution process.

The Service Workbench is an incident management system with the goal to optimize the problem resolving efficiency and documentation within the service team of an organization. Problems and requests can be automatically or manually collected in a central database, prioritized, categorized and assigned to certain teams or team members. The handling of the problems/incidents stored in the *Service Workbench* is monitored by escalation mechanisms. This ensures a timely solution for serious incidents. In addition solutions for problems can be documented and collected in a knowledge database to use them for similar future problems.

With the help of the module Service Workbench Ticket Creator tickets based on events within the OpenScape FM can be automatically created in the Service Workbench. For this the Service Workbench connects itself to the event management of the OpenScape FM by using the Ticket Creator. It collects, controlled by definable selection criteria (filters), relevant HiPath/OpenScape-events. The Ticket Creator uses these events to generate tickets and information about the affected components within the Service Workbench.

For objects that are related to events within the OpenScape FM, the Service Workbench Ticket Creator can automatically create entries within the Service Workbench that represent these objects. Problem tickets will then reference these objects.

Besides the automated creation, tickets can also be created manually by users of the OpenScape FM infrastructure. The Service Workbench provides a web client that can be used by the end user to create tickets for new requests, orders or complaints. It can also be used to monitor the status of this tickets. This prevents support requests for problems that have already been committed. In addition the Mail2Ticket-Interface can be used to create tickets and workorders by sending emails to the Service Workbench.

A problem ticket can be assigned to a specific staff group or dedicated to an individual responsible staff member. A member of the group that starts with the handling of the problem accepts the ticket. It also documents its actions to analyze and solve the incident within the ticket. During the handling of the ticket the ticket handler can create a number of workorders to delegate various required actions to different staff members or groups. Escalation mechanisms ensure that notification messages are send when predefined thresholds in relation to handling times are reached. This ensures that, e.g. when the ticket handler gets ill, an open problem will not be forgotten.

The Service Workbench supports an efficient analysis and handling of HiPath/OpenScape incidents. It provides a back navigation from a ticket to the respective OpenScape object within the OpenScape FM. For further work steps, the handler can also use functions directly that are provided by the context menu of the respective object.

If a problem has been solved, the staff member can document the solution of the problem within the solution database. Here the solution will be available to support the handling of similar future problems. When a ticket is closed the respective customer or problem reporter can be automatically informed by email.

By using the Service Workbench the service process is organized in a comprehensible fashion. The generation of trouble tickets is optimized, service cases are handled more efficient and the workflow is controlled (escalation) and documented. The whole service process is managed by clear and easy to handle methods.

3 Overview

This chapter provides an overview about the terms used within the Service Workbench.

Section 3.2 contains a ticket handling example.

3.1 Terms

As described in *Chapter 2*, the Service Workbench is used to collect information about problems that are reported by customers, service technicians or the OpenScape FM.

Reported problems are stored in **Tickets** that are the central element of the Service Workbench (see *Chapter 5*). Tickets represent a problem and are the collection point for the information that helps the service technicians to solve the respective problem. This information can e.g. be added by descriptive text elements, attachments or messages. Tickets also document and monitor the review and troubleshooting activities that are performed during the problem solving process.

A **Priority** is assigned to each Ticket to emphasize the importance of the respective problem.

Tickets can be created manually by Customers using the Web Interface (see *Section 14.4.1*), by service technicians using the Service Workbench Client (see *Section 5.2*) or automatically by an OpenScape FM that directly connects to the Service Workbench.

In the context of the Service Workbench, **Customers** (see *Section 9.1*) report problems and have a restricted access to their problem tickets. **Companies** (see *Section 9.2*) are used to aggregate associated Customers.

A **User** (see *Section 13.2*) represents a service technician that is responsible for the handling of Tickets and the respective problems. Users can be aggregated to **Groups** of service technicians (see *Section 13.3*). Tickets can be assigned to a single User or to a Group of Users.

Often a problem affects a specific object (e.g. a specific PC). Specific objects are represented within the Service Workbench as **Components** (see *Section 8.2*). These Components may contain **Subcomponents** (like a printer or hard disk assigned to the specific PC). Subcomponents are represented as Components that are **Part Of** another Component (in the example the specific PC) which allows the creation of a hierarchical object tree.

Software Instances are handled as a special case of Components and stored separately (see *Section 8.4)*. A list of the installed **Software Instances** can be added to a Component.

Components can be assigned to Tickets to indicate the affected object.

General object types can be defined as so called **Component Catalogs** (see *Section 8.1*) which contain basic information about the object type. Components can be defined as an instance of a Component Catalog. For example: a Component Catalog entry may describe a generic printer type, while individual Component entries represent actual physical printers of this type.

Software is the equivalent of Component Catalogs (see *Section 8.3*). For example: Software may represent a specific operating system, while Software Instances represent the actual installations of this operating system.

Rooms can be used to describe physical locations (see *Section 8.5*). Customers, Components and **Ports** (representing e.g. telephone sockets) can be assigned to individual Rooms to provide a quick overview.

The current state of the Ticket and the respective problem is displayed by the **Status** of the Ticket. This enables the Users to identify unsolved problems at a glance. The Status can e.g. be a value like *New*, *Working*, *Solved* or *Closed*.

A defined **Workflow** ensures, that Status changes have to be performed in a predefined order (see *Appendix A*). For example, by default only Tickets with the Status *New* or *Solved* can be set to the final Status *Closed*.

Changes to the Status of a Ticket or general changes to a Ticket are listed in the **Status Transition** page and the **Diary** page of the Ticket.

Service Level Agreements (**SLA**) can be assigned to Tickets (see *Section 13.8*). Depending on the assigned SLA and the Priority, messages will be created, if Tickets are not resolved in the time frame allowed by the SLA.

The time calculation for SLAs considers **Working Time** and **Holidays**.

Usually solving a problem involves a number of tasks that have to be performed. These tasks can be represented within the Service Workbench in the form of **Workorders** (see *Chapter 6*). Workorders can be attached to a ticket, and are treated similar to tickets. This includes the assignment to Users or Groups and an individual Status.

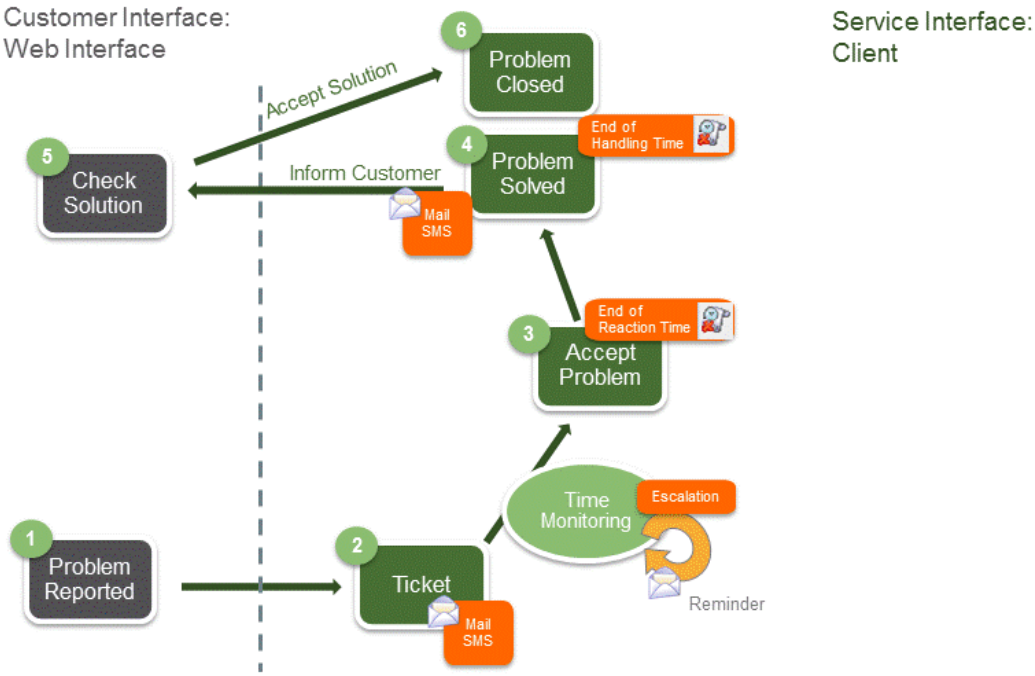
When a problem is solved, a **Solution** (see *Chapter 7*) has to be attached to the respective Ticket. Solutions keep track of the steps that were performed to solve the problem. Already known Solutions can be attached to new problems.

To assist the problem solving process, **Actions** that are often used for objects in the Service Workbench, can be defined and started for individual entries (see *Chapter 10*). Actions can be started manually by service technicians or automatically by the Workflow. They can be provided for any type of data (e.g. Components, Customers, Tickets, Rooms). Examples for Component Actions are the execution of a Ping or the opening of a remote desktop for the respective Component object.

3.2 Example: Ticket Handling

Starting with the reporting of a problem and ending with its closure, the following examples describe the possible steps that are passed within the Service Workbench.

Example 1: Simple Service Case



Step 1: In the represented case, a Customer uses the web based customer interface to **Report** his problem. The direct reporting of the case by the Customer avoids the loss of information and prevents misunderstandings during the Ticket creation.

Step 2: The appropriate technician will be informed about the new **Ticket** (problem case), e.g with an email. The **Time Monitoring** of open Tickets helps to prevent that problems are forgotten and SLAs are violated.

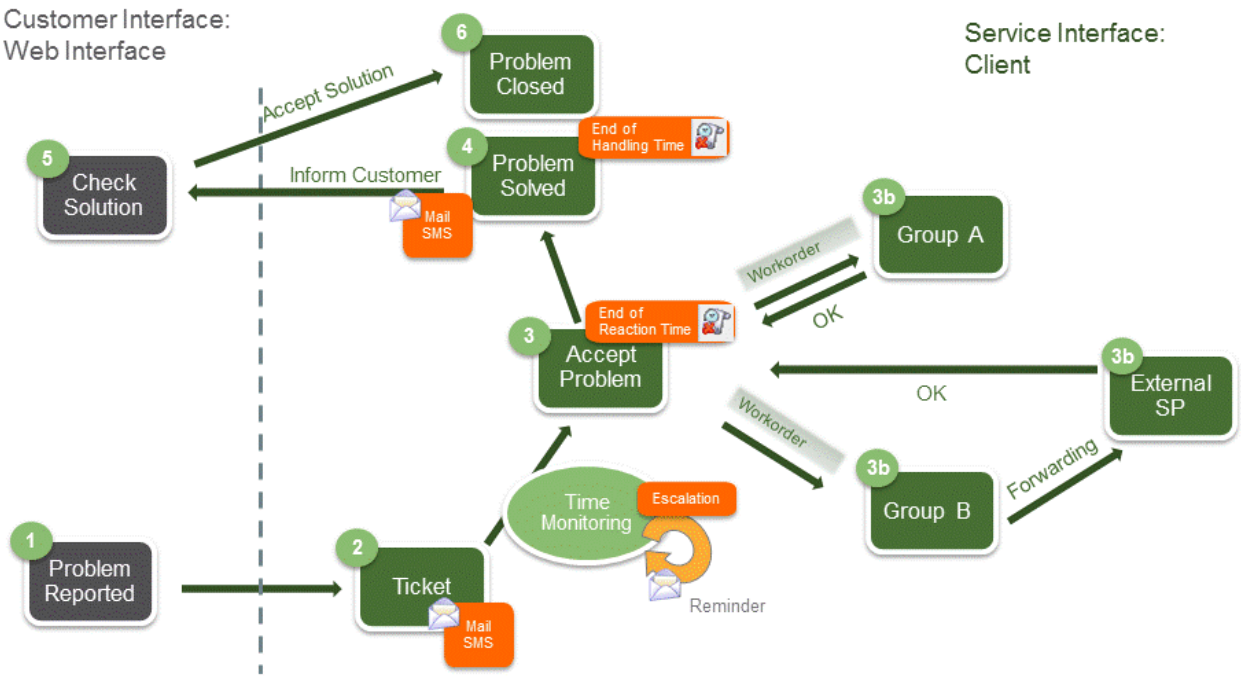
Step 3: The problem will be set to **Working** and the responsible service technician tries to solve the problem.

Step 4: If the service technician can solve the problem without the help of another service technician or a service team, the Ticket can be set to **solved**. The Customer will be automatically informed about the change of the status.

Step 5: Within the Web-Interface the Customer gets informed about the **Solution** of his problems and he can **check**, if the problem is solved from his point of view.

Step 6: If the Customer agrees with the Solution for the problem, he can accept it by using the Web-Interface. This will **close** the respective Ticket. The service technician can also close a Ticket after it is handled.

Example 2: Extended Service Case



The work steps described in the previous example 1 are mostly sufficient for simple service cases. For more complex cases, the Service Workbench supports problem solving by teams. The handling of a Ticket for a more complex problem can look like this:

Step 3b: If the service technician needs further technical support to solve the problem, he can forward the Ticket to other technicians or a technician group. If he needs help from different service teams during the solution process, he can create Workorders and assign them to the respective teams. This also includes the involvement of external service providers. The centralistic documentation of the problem solution process provides an overview about the current state of the solution process at any time.

Step 4: If all involved support teams report a successful handling and the problem is resolved, the Ticket can be set to **solved**. The Customer will be automatically informed about the change of the status.

4 First Steps

This chapter describes how a Service Workbench Client can be started. It also describes the initial view of the Client.

4.1 Starting the Service Workbench Client

The Service Workbench Client can be started using one of the following three methods:

- **From a Web Browser**:
  By entering the URL
  `https://server:3080/serviceworkbench/` or
  `http://server:3080/serviceworkbench/`
  within a Web Browser.
  Within the URL `server` is the hostname or IP of the machine on which the Service Workbench Server is running, and `3080` is the (Default) Connection Port. Both values have to be replaced accordingly.
- **From the OpenScape FM**:
  By using the menu entry **Service Workbench->Service Workbench Client** from the main menu **Add-Ons** of the OpenScape FM. Prior to starting the client for the first time, the interface has to be configured by using the entry **Service Workbench->Configure Service Workbench Server** from within the same main menu.

All three methods open the Service Workbench Client displaying the Initial View (see *Section 4.2*).

4.2 Initial View

When a Service Workbench Client is initially opened, it displays the **Navigation Tree** on the left side and an **Overview** on the right side. These elements are described in the following subsections.

4.2.1 Navigation Tree

The **Navigation Tree** is always located on the left side of the Service Workbench Clients. It is used to choose the information that should be displayed on the right side of the Client.

Most entries in the Navigation Tree represent the various dataset types (see *Section 12.2*) or predefined searches (represented as child objects for dataset types) for them. If a dataset or a predefined search is selected in the Navigation Tree, the matching data will be displayed on the right.

The context menu provided for most entries can be used to activate various functions. These are described in the sections addressing the respective datasets.

When the topmost entry of the Navigation Tree is selected (**Service Workbench**), the initial view of the Service Workbench Client will be displayed.

4.2.2 Dashboard Overview

The page **Dashboard** relates to Tickets and Workorders and displays a number of tiles that contain the following information:

- **New Tickets of the last 14 days**: A history diagram based on the number of Tickets that were created each day (based on their Priority) during the last 14 days.
- **New Workorders of the last 14 days**: A history diagram based on the number of Workorders that were created each day during the last 14 days.
- **Last 10 Tickets**: The 10 Tickets that were created last are listed. Individual forms can be opened by double clicking the respective entry.
- **Last 10 Workorders**: The 10 Workorders that were created last are listed. Individual forms can be opened by double clicking the respective entry.
- **My open Tickets - Priority**: A statistic about the number of Tickets (based on their Priority) that are assigned to the User and are not in Status *Closed*.
- **Components with most open Tickets**: A statistic about the Components to which the most still open Tickets are assigned.
- **Next 10 escalating Tickets**: The 10 Tickets that will escalate next if they are not handled are listed. Individual forms can be opened by double clicking the respective entry.
- **Ticket-Overview**: A statistic that displays the total number of Tickets that are not assigned to anyone, the number of Tickets assigned to the current User that are still open and the number of Tickets for the current User that are currently escalated.

The arrangements of the individual tiles can be changed by using Drag&Drop.

5 Tickets

Within the Service Workbench environment, Tickets describe individual problems or requests that should be handled by service technicians.

To ensure that the handling of the Tickets follows certain standards, a predefined Workflow (see *Section 5.3*) has to be followed during the handling process.

Individual tasks within the process can be separately defined as Workorders (see *Chapter 6*).

If a Ticket should be closed to which open Workorders are assigned, a respective notification will be displayed for the Ticket handler.

To finish the handling process, a Solution (see *Chapter 7*) has to be assigned to the Ticket, and the Customers that reported the problem or request get a chance to evaluate this Solution (see *Section 14.3.2*).

Ticket datasets contain information about the respective problems or requests.

5.1 Listing Tickets

The Ticket data is collected within individual datasets (see *Section 12.2*) of type *Ticket*.

An overview about all Tickets, that might be interesting for the current User, is displayed by clicking the Navigation Tree object **Ticket**. The overview contains three lists of Tickets that are sorted in descending order by Ticket Number. Tickets with the Status *Closed* are not shown in this list. Tickets that have already exceeded their processing deadline (Escalation Date) are shown in red. The other Tickets are shown in blue. The lists contain:

- All Tickets that are currently not assigned to a User or Group.
- All Tickets that are assigned to the current User and are still open.
- All Tickets that are assigned to a Group of which the current User is a member that are still open.

In addition five predefined Extended Searches (see *Section 12.2.3.3*) are provided for Tickets. Their results are displayed when the respective entries, located below the Navigation Tree object **Tickets**, are clicked. This corresponds to selecting the entry **Open Table View** within the context menu of the respective object.

The predefined Extended Searches for Tickets are:

- **My Tickets**: All open Tickets where the current User is the User of the Ticket.
- **Open**: All Tickets that are not in Status *Closed*.
- **Closed**: All Tickets that are in Status *Closed*.
- **All**: All Tickets known by the Service Workbench.
- **Last Edited**: All Tickets that were changed by the current User within the last 30 days.

5.2 Creating Tickets

Tickets can be created as usual by using the **New** menu or a respective **Uplink Viewer** (see *Section 12.2.1*). But there are other methods to create Tickets without using the Service Workbench Client:

- Problems or requests can be stated by Customers using the Service Workbench Service Portal. These problems or requests will be represented by automatically created Tickets (see *Chapter 14*).
- If the Service Workbench is connected to an OpenScape FM and the connection is configured accordingly, then Tickets can be automatically created for events that are detected by the OpenScape FM (see *Chapter 14*).
- If the Mail2Ticket Interface is activated and configured accordingly, incoming emails may trigger the creation of Tickets (see *Section 5.9*).
- Tickets can be automatically created by the Service Workbench itself if internal problems are detected by the system (e.g. when the sending of notification emails fails).

5.3 Workflow for Tickets

The Workflow for Tickets helps to ensure that the handling of Tickets has to abide a predefined guideline.

The processing follows certain steps that have to be performed in a defined order. Within the Service Workbench these steps are represented by the **Status** of a Ticket (see *Section 5.3.1*).

Besides the order of the steps, the **Priority** and the **SLA** that is assigned to a Ticket defines a guideline for the time in which a Ticket has to be handled (see *Section 5.3.2*).

5.3.1 Status

The **Status** of a Ticket describes the handling state of the Ticket. The six possible Status values have the following intent:

- **New**: Every Ticket is created with this Status. Only Tickets that have not been handled at all should have this Status.
- **Forwarded**: The Ticket has been assigned to another User. The new User is notified, but currently no one is working on the Ticket.
- **Working**: The Ticket is currently handled by the Ticket‘s User or Group. This Status should be used when the support is actively working on the problem.
- **Deferred**: The handling of the Ticket has been put on hold. Currently no one is actively working on the Ticket, and escalation times (see *Section 5.3.2*) are also paused. This Status should only be used when the support is currently waiting for an action that is beyond their control.

Examples:

- The service team is waiting for a response (e.g. a logfile) from the Customer that reported the problem, and can do nothing useful for the Ticket as long as the response is not received.
- A request for a new printer has been handled by the support (the printer has been ordered) but the printer yet has not been received. The Ticket can only be closed when the printer has been delivered to the Customer, but there is no reason to escalate the Ticket.

- **Solved**: This Status should be used when a Ticket is solved but the Solution is not yet confirmed by the Customer that reported the problem.
- **Closed**: The handling process for the Ticket is finished. This Status is usually reached when the Ticket‘s Solution is confirmed by the Customer that reported the respective problem, or when a request is denied by the service team. If further actions should be performed, a new Ticket has to be created by duplicating the closed Ticket (see *Section 12.2.4*).

The handling of every Ticket starts with the Status *New* and ends with the Status *Closed*.

To which Status a Ticket can be changed is depending on the current Status. While there is a useful predefined definition for the allowed changes, it can be modified by an Administrator (see *Section 13.7*).

Status changes have to be commented by the User.

5.3.2 SLA

An **SLA** or Service Level Agreement can be assigned to a Ticket. Roughly speaking, it defines the time that might be used to solve the problem.

Within the SLA individual times can be defined that determine the maximum time allowed, dependent on the **Priority** of the Ticket. Within these times a certain or higher Status for the Ticket has to be reached. The order of the Status values in this context is: *New*, *Forwarded*, *Working*, *Deferred*, *Solved*, *Closed*.

Thresholds can be assigned that trigger escalations when the allowed time or a percentage of it, has been exceeded. Only working hours are used to calculate used time and the handling time is frozen while a Ticket is in Status *Deferred*.

SLAs can only be configured by an Administrator (see separate *Administrator Guide*).

By default, for each escalation one warning will be send to the defined Users or Groups. To prevent that the handling of the Ticket is ‚forgotten‘ after the warning, an *Administrator* can configure the general escalation mechanism to repeat the warning in defined time intervals until the Status of the Ticket changes.

Example for a possible configuration:

The allowed time to reach the Status *Working* for Tickets with the Priority *Medium* has been set to 8 hours.

Actions will then be performed when the Status of the Ticket is not *Working*, *Deferred*, *Solved* or *Closed* and the Thresholds are exceeded.

Thresholds in this case might be:

- First escalation at 50% (4 hours): Send an email to the User of the Ticket.

- Second escalation at 75% (6 hours) send an email to the User and the Group.
- Third escalation at 100% (8 hours) send an email to the User and the Manager.

These escalations will only trigger, if the **SLA** is assigned to the Ticket, the **Priority** of the Ticket is *Medium*, the Ticket has not yet reached at least the **Status** *Working*, and the trigger times are exceeded.

5.4 Editing Tickets

Individual Tickets can be edited by using the form *Ticket*. It can be opened by e.g. double clicking an individual Ticket from any list of Tickets (see *Section 5.1*). It can also be opened by using the entry **Open** from the context menu of a Ticket object.

When a new Ticket is created, the same form is used.

The main page **Activity** contains a list of the manual changes made to the ticket (see *Section 5.5*) and, if a connection to the Event Correlation Engine Workflow has been defined, the page **Workflow** can be used to perform the next workflow steps.

The respective **Ticket-Dataset** itself is displayed on the main page with this name which contains the following information:

Base Data:

- **No.**: The unique ID of the Ticket. The number is automatically created when a Ticket is created.
- **Source**: A selection menu that indicates the media that was used to register the Ticket. Predefined are the Sources *Phone*, *Mail*, *Web* and *Support*. These values can be modified by an Administrator in the Itemized List *Ticket:Source*.
- **Type**: A selection menu that indicates the type of the Ticket. Predefined are the Types *Problem Report*, *Query* and *Extension Request*. These values can be modified by an Administrator in the Itemized List *Ticket:Type*.
- **Customer**: The Customer that reported the problem described in the Ticket. The checkboxes below can be marked if an email to the Customer should be automatically created when the Ticket changes to the Status *Working* or *Solved*.

  An Uplink Viewer connection to an existing or newly created dataset of type *Customer* (see *Section 9.1*).
- **Cost Center**: The Cost Center that will be used for the Ticket. When a Customer is assigned to the Ticket, its Cost Center will be automatically used for the Ticket when this field has not been set before. When a Customer has no assigned Cost Center the Cost Center of the Customer‘s Company will be used instead.

  An Uplink Viewer connection to an existing or newly created dataset of type *Cost Center* (see *Section 13.12*).
- **Affected Component**: The Component associated with the Ticket. When a Component search is opened, it is initially restricted to Components assigned to the Customer of the Ticket.

  An Uplink Viewer connection to an existing or newly created dataset of type *Component* (see *Section 8.2*).
- **Other Component**: A custom text field in which components can be named for which no separate record exists.

- **Short Description**: A text field containing a few keywords that describe the problem or request associated with the Ticket. This field is mandatory and the description may not exceed 250 characters.
- **Ticket Description**: A field with the detailed description of the problem. By default, the description may not exceed 4000 characters.

Workflow and Escalation Data:

- **Status**: A selection menu for the Status of the Ticket (*New*, *Forwarded*, *Working*, *Deferred*, *Solved*, *Closed*). The Status defines a handling state of the Ticket in relation to the Ticket‘s Workflow (see *Section 5.3*). A Ticket may only be set to a Status that is allowed as a potential successor Status for the current Status of the Ticket. Permissible Status values are indicated by a green tick in front of the Status. A new Ticket always starts with the Status *New*.
- **Priority**: A selection menu for the Priority of the Ticket in relation to the selected SLA. Possible values are *Low*, *Medium* and *High*. The selected value in combination with the SLA assigned to the Ticket define the reaction times for the Ticket (see *Section 5.3*).
- **Urgency**: A selection menu that indicates the priority of the Ticket as it is seen by the Customer that created the Ticket. While this selection has no direct effect on the escalation times for the Ticket, it can be used as an indicator for the Ticket‘s importance. Possible values are *Low*, *Medium* and *High*.
- **SLA**: This selection menu indicates the Service Level Agreement that will be used for the Ticket. Reminder times are set according to the SLA and the Status and Priority of the Ticket. SLAs can be created or modified by an Administrator (see separate *Administrator Guide*).

  If no SLA is selected, the system will automatically apply the default SLA *Standard*.

  If a Ticket is automatically created by the Fault Management, the SLA *FMDefault* will be assigned.

  The SLA *Workflow* provides no escalation times. It should be used when the escalation is handled by the Workflow Engine.
- **Classification**: Here, a Ticket Property can be selected that has been set via the Itemized List `Ticket:Classification` (see Service Workbench Administrator Documentation). By default, the confidentiality level of the Ticket can be set here.
- **Follow-Up Date**: A date and time field that can be used to define a follow-up date. When this date is reached, an email containing an event invitation (VCalendar) will be send to the current User of the Ticket. When no User has been specified, the email will be send to all members of the assigned Group. If neither a User nor a Group are assigned, an entry to the Diary will be created.

  The Follow-Up can be used to manually define a reminder date, independent of the SLA escalation mechanisms. This can e.g. be used if the service technician waits for a response by the Customer to get a reminder when the response is delayed.

  When the main menu entry **Release->Release to Web (Follow-Up)** has been checked in the Ticket‘s form, the Follow-Up Date and Code will be visible for a Customer using the Service Workbench Service Portal.
- **Follow-Up Code**: This selection menu contains the reason for the Follow-Up. A reason can be typed or selected from the attached list. The list values can be modified by an Administrator in the Itemized List *Ticket:Follow-Up Cause*.

- **Escalation**/**Next Escalation**: These fields display the time when the next escalation is scheduled and the current escalation level. Escalation times are recalculated when the Status, the SLA or the Priority of the Ticket changes or when an escalation time is reached.
- **Solution**: The Solution that has been found for the Ticket. A Ticket cannot get into Status *Solved* or *Closed* when no Solution has been assigned. More about the selection of Solutions can be found in *Section 5.7*.

  An Uplink Viewer connection to an existing or newly created dataset of type *Solution* (see *Chapter 7*).

  Generally Customers using the Service Workbench Service Portal will only see the Solution when the Ticket changes to the Status *Solved*. If the Solution should be visible at an earlier time, the main menu entry **Release->Release Solution to Web** can be checked in the Ticket‘s form.

Relation Viewer:

- The Relation Viewer at the bottom of the form contains references to other objects that were assigned to the Ticket (e.g. Workorders and Attachments).
- The page **Workorders** displays a list of Workorders (see *Chapter 6*) that have to be completed to solve the current Ticket. The buttons on the right side can be used to create or delete Workorders for the Ticket or to find a specific Workorder using an Uplink-Viewer. Any number of Workorders can be assigned to a Ticket.
- The pages **Diary** and **Status Transitions** keep track of comments and status changes for the Ticket. Entries that were manually entered or value changes for configured fields are displayed in blue. The page **New Diary Entry** can be used to assign worksteps as new entries for the Ticket‘s Diary and assign execution times to them.
- The page **Info** contains information about the **Time Consumed** for the Ticket, when the ticket was **Created** and when it was **Last Changed**. The Time Consumed is obtained by adding the times entered in the field **Time Consumed (min)** on the page **New Diary Entry**.
- The page **Comments** can be used to create Comments for the Customer that is assigned to the Ticket, or to read Comments generated by the Customer for the Ticket (see *Section 14.3.1*).
- The pages **Extension Fields** (see *Section 11.6*) and **Attachments** (see *Section 11.7*) can be used to provide additional information for the Ticket.
- The page **E-Mails** lists the emails that modified the Ticket automatically (see *Section 5.9*) or that were sent to the customer by the ticket (see *Section 5.10*).
- The page **Costs** can be used to view or modify the costs connected to the Ticket (see *Section 11.4*)
- The page **Categorisation/Forwarding** can be used for three major tasks:

  1. A specific **Category** (see *Section 13.1*) has to be assigned to the Ticket by selecting an entry within the Category tree.

     If a Ticket is not assigned to a User or a Group, the Ticket will be assigned to the default Group of the selected Category. This will be applied when the Ticket is saved. If a Default Group has been configured for a Category, it will be displayed as a tooltip within the Category tree.
  2. A Solution can be found for the Ticket based on the Category by pressing the button **Solution** (see *Section 5.7.2*).

3. The currently assigned **User** or **Group** is displayed. These entries define the service technician or technicians that are responsible for the handling of the Ticket. The Ticket can be forwarded to a new **User** (see *Section 13.2*) or **Working Group** (see *Section 13.3*) by using the respective selection menus.

   The Users and Groups listed in the menus are marked based on the selected Category:
   Users/Groups marked in green are recommended for the Category,
   Users currently not active are marked in red.

   If a Ticket is reassigned to a different User or Group, that User or all members of the Group are notified by email to inform about the new responsibility.

5.5 Changelogs for Tickets

The changes that were performed during the editing of a Ticket (see *Section 5.4*), can be displayed in a protocol window.

This can be displayed by using the entry **Changelog** from the main menu **Extras** within the Ticket‘s form.

The entry opens a window that displays the original values and the current values in table form.

In addition, manual changes made within the Ticket can be watched on the main page **Activity**.

5.6 Ticket Templates

Tickets for common problems are often very similar to each other. To avoid repetitive work, mainly for the first-level support staff that is usually responsible for the manual creation of Tickets, the Service Workbench provides the storage of Ticket Templates. These templates are sorted based on the Category of a Ticket and can be loaded into a Ticket to fill most of the fields with the stored data.

Create Ticket Template: (only possible with Administrator access)

Any Ticket can be stored as a Template by using the entry **Save as Ticket Draft** from the main menu **Extras** of the Ticket‘s form. When this entry is selected, the current content of most fields of the Ticket will be stored to the new template (e.g. *Priority*, *Short Description*, *Ticket Description*, *Category*). Not included to the stored data are links to other objects that are displayed in the Relation Viewer of the Ticket.

Use Ticket Template:

A stored Template can be loaded into a displayed Ticket by using the entry **Use a Ticket Draft** from the main menu **Extras** of the Ticket‘s form. This will display a Category Tree in which the stored Templates are sorted based on their Category. If a Template is selected, its data will overwrite the corresponding data of the current Ticket.

5.7 Assigning Solutions

Tickets can only be set to the Status *Solved* or *Closed* when a Solution has been assigned to the Ticket (see *Chapter 7*). There are two fundamental methods to assign a Solution to a Ticket:

- Using the Uplink Viewer **Solution** to create a new Solution or to find an existing Solution by using a keyword search (see *Section 5.7.1*).
- Performing a search based on the **Category** of the problem (see *Section 5.7.2*).

Both methods allow a quick search for Solutions during the creation of a Ticket.

5.7.1 Searching Solutions by Keyword

The keyword search for a Solution can be started by using the field **Solution**.

A text, for which a Solution should be found, can be entered into this field and the option **Search** in the respective Uplink Viewer will start the search.

Only Solutions that contain the search word in their *Component Catalog*, *Category*, *Solution Description*, *Solution Short Description*, *Number* or *Status* will be displayed. If the field **Solution** is left empty, all Solutions will be displayed.

Since the whole expression and not individual words will be searched, only catchwords or keywords should be used instead of full sentences.

By using the Uplink Viewer, new Solutions can also be created.

5.7.2 Searching Solutions by Category

If the Category of the problem should be considered when a Solution is searched, this can be done by using the page **Categorisation/Forwarding**.

By clicking the button **Solutions** located besides the Category Tree, the currently selected Category will automatically be used for the search of a Solution. The result list will only display Solutions with the respective Category.

By checking **Also check parent categories** or **Also check sub categories** the search will also find Solutions that have been defined for Parent Categories or Sub Categories.

5.8 Releasing Tickets

The Service Workbench Service Portal can be used by Customers to create and monitor Tickets (see *Chapter 14*).

The data displayed in the Service Portal can be restricted for individual Tickets by using the menu entries of the main menu **Release** within the Ticket‘s form.

The following restrictions can be set by removing the marks in front of the respective entries:

- **Release to Web (Ticket)**: If this entry is not marked, the Ticket cannot be seen in the portal at all.
- **Release to Web (Follow-Up)**: If this entry is marked, then the Follow-Up information for the Ticket is visible in the portal.

Follow-Ups are used to manually define a reminder date, independent of the SLA escalation mechanisms.

- **Release Solution to Web**: If this entry is not marked, the Solution for the Ticket can only be seen in the portal when the Status of the Ticket changes to *Solved*. If the Solution should be visible at an earlier time, the entry has to be marked.

5.9 Mail-Interface for Tickets

The Service Workbench provides the option to modify or create Tickets through received email.

For this feature the **Mail2Ticket Interface** has to be activated and configured accordingly (see separate *Administrator Guide*).

When an email is received by the Service Workbench, the system checks whether the **Subject** of the email contains a known **Ticket ID** or **Workorder ID**.

If this is the case, the email will be assigned to the respective Ticket or Workorder and will be displayed within the Relation Viewer on the page **E-Mail**. In addition a notification will be send to the User of the Ticket or Workorder. to inform about the new E-Mail assignment.

If this is not the case, a new Ticket will be created. The subject of the email will be used as the **Short Description** of the Ticket, and the body of the email will be used as the **Ticket Description**. Email attachments will be added to the Ticket as **Attachments**. The Ticket will be assigned to the User or Group defined by the Mail2Ticket Interface, and notifications are send to the respective User or Group members. The email itself will also be attached to the Ticket and can be seen in the page **E-Mail** within the Relation Viewer of the Ticket.

5.10 Sending Emails to Ticket Customers

By using the entry **Email to Customer** from the main menu **Extras** from a Ticket form, emails can be directly send to the Customer of the current Ticket.

Emails sent by this method will be added to the Ticket as an email attachment and can be viewed within the Relation Viewer on the page **E-Mails**.

Additional recipients can be added as a **CC**. They have to be separated by a comma.

5.11 Reports for Tickets

By using the entry **Ticket Report** from the main menu **Extras** from a Ticket form, reports can be generated for the currently displayed Ticket.

The created reports contain the complete information of the Ticket. By using the respective marks, it can be selected whether the report should also contain the *Workorders*, *Diary entries*, *Attachments* and *Emails*.

The result can be a printout or a file (e.g. in PDF format).

5.12 Converting Tickets to Workorders

Administrators can convert an individual Ticket into a Workorder by selecting the entry **Convert to Workorder** from its context menu within a displayed list of Tickets (see *Section 5.1*).

This action creates a new Workorder which will be pre-filled with the matching data from the Ticket. The Number of the new Workorder is displayed, and the Ticket will be deleted.

6 Workorders

Within the Service Workbench environment, Workorders define individual tasks that are to be handled by service technicians.

Workorder datasets contain information about the respective task.

Service Orders are a special case among the Workorders. Service Orders are orders that were created automatically by the Service Portal.

In the following, unless explicitly stated otherwise, Service Orders are handled exactly like 'normal' Workorders.

6.1 Listing Workorders

The Workorder data is collected within individual datasets (see *Section 12.2*) of type *Workorder*.

An overview about all Workorders, that might be interesting for the current User, is displayed by clicking the Navigation Tree object **Workorder**. The overview contains four lists of Workorders that are sorted by the Workorder number in descending order. Workorders with the Status *Closed* are not shown in these lists. Workorders that have no assigned User and Group are shown in red. The other Workorders are shown in blue. The lists contain:

- All Workorders that are currently not assigned to a User or Group.
- All Workorders that are assigned to the current User and are still open.
- All Workorders that are assigned to a Group of which the current User is a member that are still open.
- All Workorders that have been initiated by the current User.

In addition five predefined Extended Searches (see *Section 12.2.3.3*) are provided for Workorders. Their results are displayed when the respective entries located below the Navigation Tree object Workorders are selected. This corresponds to selecting the entry **Open Table View** within the context menu of the respective object.

The predefined Extended Searches for Workorders are:

- **My Workorders**: All Workorders where the current User is the User of the Workorder and that are not in Status *Closed*.
- **Open**: All Workorders that are not in Status *Closed*.
- **Closed**: All Workorders that are in Status *Closed*.
- **Last Edited**: All Workorders that were changed by the current User within the last 30 days.
- The searches **My Service Orders**, **Open Service Orders**, **Closed Service Orders** and **Last edited Service Orders** correspond the searches listed above. They differ only by the fact that they are restricted to Workorders of the type *Service Orders*.
- **All**: All Workorders known by the Service Workbench.

6.2 Workflow for Workorders

In contrast to Tickets (see *Section 5.3*), there are no rules for Workorders that enforce a strict order of Status changes.

Nevertheless there are some underlying consequences based on the *Status* and *Escalation Date* assigned to a Workorder. These are described in the following subsections.

6.2.1 Workorder Status

The **Status** of a Workorder describes the handling state of the Workorder. While the eight possible Status values have a mostly informative purpose, there is an underlying intent:

- **New**: Every Workorder is created with this Status. Only Workorders that have not been handled at all should have this Status.
- **Submitted**: The task described by the Workorder has to be confirmed by a superior or expert. The Workorder is on hold until it is approved.
- **Approved**: The task described by the Workorder is approved by a superior or expert, but no one is currently working on the task.
- **Working**: The task described by the Workorder is currently handled by the Workorder‘s User or Group. This Status should be used when the support is actively working on the task.
- **Forwarded**: The Workorder has been assigned to another User. The new User is notified, but currently no one is working on the Workorder.
- **Deferred**: The Workorder will not be processed further for the time being. A *Followup Date* and a *Followup Code* must be specified for this status.
- **Rejected**: The Workorder was not processed but finished nevertheless.
- **Closed**: The handling process for the Workorder is finished. This Status is considered final and can only be removed by an Administrator. The removal of the Status should be avoided, because it might generate an inconsistent state if the assigned Ticket is already in Status *Closed*, or if the Costs for the Workorder are already calculated. If further actions are planned, a new Workorder should to be created by duplicating the closed Workorder (see *Section 12.2.4*).
- **Error**: The Workorder has been handled by Workflow Actions and the handling failed (see *Section 10.1*).
- **Deactivated**: The automated handling of Workflow Actions have been paused.

The handling of every Workorder starts with the Status *New* and ends with the Status *Closed*.

To which Status a Workorder can be set is not depending on the current Status. An exception is the Status *Closed* that can only be removed by an Administrator.

Status changes have to be commented by the User.

6.2.2 Workorder Escalation

Unlike for Tickets (see *Section 5.3*) there is no automatic escalation mechanism for Workorders.

The time/date fields **End Date** and **Escalation Date** within the Workorder form can be used to define notification times for the service technician or technicians that are responsible for the task described by the Workorder.

These escalations are meant to inform the technicians about the planned or upcoming end date.

The escalation is stopped when the Workorder has the Status *Closed*, *Error* or *Deactivated*.

By default, for each escalation one warning will be send to the defined Users or Groups. To prevent that the handling of the Workorder is ‚forgotten' after the warning, an *Administrator* can configure the general escalation mechanism to repeat the warning in defined time intervals until the Status of the Workorder changes.

6.3 Editing Workorders

Individual Workorders can be edited by using the form *Workorder*. It can be opened by e.g. double clicking an individual Workorder from any list of Workorders (see *Section 6.1*). It can also be opened by using the entry **Open** from the context menu of a Workorder object.

When a new Workorder is created, the same form is used.

The main page **Activity** contains a list of the manual changes made to the Workorder (see *Section 6.4*).

If a connection to the Event Correlation Engine Workflow has been defined, the page **Workflow** can be used to perform the next workflow steps.

If the Workorder is a *Service Order*, the main page **Service Details** provides further information about the Service.

If the Workorder was created by the Web Shop, the main page **Order** contains additional information about the order.

The respective **Workorder-Dataset** itself is displayed on the main page with this name which contains the following information:

Base Data:

- **No.**: The unique ID of the Workorder. The number is automatically created when a Workorder is created.
- **Type**: This field shows, whether the Workorder is a ‚normal' Workorder or a Service Order.
- **Initiator**: The User that created the Workorder. This value is set on the creation of the Workorder and cannot be changed afterwards. The checkboxes below can be marked if an email to the Initiator should be automatically created when the Workorder changes to the Status *Working* or *Closed*.
- **Working Group** and **User**: This selection menus define the service technician or technicians that are responsible for the handling of the Workorder. The Workorder can be forwarded to a new **User** (see *Section 13.2*) or **Working Group** (see *Section 13.3*) by using the respective selection menus.
- **Short Description**: A text field containing keywords that describe the task associated with the Workorder. This field is mandatory and the description may not exceed 250 characters.

- **Description**: A field with the detailed description of the task. The description may not exceed 4000 characters.
- **Ticket**: The Ticket that describes the problem for which the task in this Workorder has been created. When a Workorder is created from a Ticket‘s form, this field and the fields *Component* and *Customer* are automatically filled based on the Ticket data.

  An Uplink Viewer connection to an existing or newly created dataset of type *Ticket* (see *Chapter 5*).
- **Component**: The Component associated with the Workorder. Usually this is the Component that is also assigned to the respective Ticket.

  An Uplink Viewer connection to an existing or newly created dataset of type *Component* (see *Section 8.2*).
- **Customer**: The Customer that reported the problem (Ticket) for which the Workorder was created. The Customer selected in this field can see the released Workorder (see *Section 6.6*) in the Service Portal (see *Chapter 14*).

  An Uplink Viewer connection to an existing or newly created dataset of type *Customer* (see *Section 9.1*).
- **Cost Center**: The Cost Center that will be used for the Workorder. When a Customer is assigned to the Workorder before the Cost Center, the Customer‘s Cost Center will be automatically used for the Workorder. When a Customer has no assigned Cost Center the Cost Center of the Customer‘s Company will be used instead.

  An Uplink Viewer connection to an existing or newly created dataset of type *Cost Center* (see *Section 13.12*).
- **Diary Entry**, **Time Consumed**: Fields that can be used to assign worksteps as new entries for the Workorders‘s Diary (see below) and to assign execution times to these steps. The entry will be added to the Diary when the dataset is saved.

Workflow and Escalation Data:

- **Priority**: A selection menu for the Priority of the Workorder. Possible values range from *1* to *5*. where *1* is the highest Priority (most important).
- **Status**: A selection menu for the Status of the Workorder (*New*, *Submitted*, *Approved*, *Forwarded*, *Working*, *Closed*, *Error*, *Deactivated*). The Status defines a handling state of the Workflow. A new Workorder always starts with the Status *New*.
- **Start Date**, **End Data**, **Escalation Date**: These are time/date fields that define the planned start and end date for the handling of the Workorder. The Escalation Date is used as a reminder date for the responsible technicians.

Relation Viewer:

- The Relation Viewer at the bottom of the form contains references to other objects that were assigned to the Workorder (e.g. Emails and Attachments).
- The pages **Diary** and **Status Transitions** keep track of comments and status changes for the Workorder.
- The pages **Extension Fields** (see *Section 11.6*) and **Attachments** (see *Section 11.7*) can be used to provide additional information for the Workorder.

6.4 Changelogs for Workorders

The changes that were performed during the editing of a Workorder (see *Section 6.3*), can be displayed in a protocol window.

This can be displayed by using the entry **Changelog** from the main menu **Extras** within the Workorder‘s form.

The entry opens a window that displays the original values and the current values in table form.

In addition, manual changes made within the Workorder can be watched on the main page **Activity**.

6.5 Workorder Templates

Similar to the Ticket Templates, Workorders can also be saved as templates and re-used. This is handled respective to the methods described in *Section 5.6* for Ticket Templates.

6.6 Releasing Workorders

The Service Workbench Service Portal can be used by Customers to create and monitor Tickets and to view the related Workorders (see *Chapter 14*).

By using the menu entry **Release to Web** of the main menu **Release** within the Workorder‘s form, this function can be restricted.

If the entry **Release to Web** is unmarked, the Workorder cannot be seen in the portal.

6.7 Mail Interface for Workorders

The Service Workbench provides the option to automatically attach emails to Workorders.

For this feature the **Mail2Ticket Interface** has to be activated and configured accordingly (see separate *Administrator Guide*).

When an email is received by the Service Workbench, the system checks whether the **Subject** of the email contains a known **Workorder ID**.

If this is the case, the email will be assigned to the respective Workorder and will be displayed within the Relation Viewer on the page **E-Mail**. In addition a notification will be send to the User of the Workorder, to inform about the new E-Mail assignment.

6.8 Sending Emails to Workorder Customers

By using the entry **Email to Customer** from the main menu **Extras** from a Workorder form, emails can be directly send to the Customer of the current Workorder.

Emails sent by this method will be added to the Workorder as an email attachment and can be viewed within the Relation Viewer on the page **E-Mails**.

Additional recipients can be added as a **CC**. They have to be separated by a comma.

6.9 Reports for Workorders

By using the entry **Workorder Report** from the main menu **Extras** from a Workorder form, reports can be generated for the currently displayed Workorder.

The created reports contain the complete information of the Workorder. By using the respective marks, it can be selected whether the report should also contain the *Diary entries*, *Attachments* and *Emails*.

The result can be a printout or a file (e.g. in PDF format).

7 Solutions

Within the Service Workbench environment **Solutions** describe the procedures that led to the removal of the problem described by a Ticket.

Solutions fulfill two essential tasks:

- They provide known procedures. This allows the fast assignment of a Solution to a new Ticket when the Ticket concerns a known problem.
- With their assignment to a Ticket, they document the conducted procedures.

Therefore a Ticket cannot be closed when no Solution has been assigned to it

Strictly speaking, Solutions do not necessarily describe an actual solution for a problem. They provide the explanation why a Ticket has been closed. *No problem detected* or *Will not be treated* are also possible Solutions within the Service Workbench environment.

Released Solutions can be seen by Customers using the Web Interface (see *Chapter 14*). This can shorten the problem solving process since Customers may suggest Solutions or they may perform them by themselves, preventing a problem message at all.

7.1 Listing Solutions

The Solutions are collected within individual Datasets (see *Section 12.2*) of type *Solution*.

A list of all datasets can be displayed by clicking the Navigation Tree object **Solution**. This corresponds to selecting the entry **Open Table View** within the context menu of the object **Solution**.

7.2 Assigning Solutions

Within the respective Ticket, Solutions can be assigned to a problem. This can be done with a Keyword search or a Category search (see *Section 5.7*).

7.3 Editing Solutions

Individual Solutions can be edited by using the form *Solution*. It can be opened by e.g. double clicking an individual Solution from the list of all Solutions (see *Section 7.1*). It can also be opened by using the entry **Open** from the context menu of a Solution object.

When a new Solution is created, the same form is used.

A Solution dataset contains the following information:

Base Data:

- **No**: The running number of the Solution.
- **Solution Description**, **Solution Short Description**: The description and short description of the Solution. These are the vital search fields for finding a Solution. They should be filled with care.
- **Ticket Description**, **Ticket Short Description**: The description and short description of the Ticket for which the Solution was created. When a Solution is newly created from out of a Ticket, the respective Ticket entries will be copied into these fields (see *Section 5.7*).
- **Status**: A selection list for the Status of the Solution. When a Solution is newly created, this value will be set to *New*. Only Solutions with the Status *Reviewed*, are visible to Customers using the Web Interface (see below). In addition the Status *Invalid* can be selected.
- **Release to Web**: This checkbox defines whether the Solution is unlocked for display in the Web Interface. Within the Web Interface only those Solutions will be displayed, that are released to web and that are in the Status *Reviewed* (see *Section 14.3.2*).
- **Solution Type**: This field is used to distinguish the Solution type. Predefined are the types *Solution*, *Workaround* and *Known Bug*. These values can be modified by an Administrator in the Itemized List *Solution:Type*.
- **Categories**: The problem Category for which the Solution should be available (see *Section 13.1*). The Category can be considered during Solution searches (see *Section 5.7.2*).

Statistical Data:

- **Usages**: A counter that displays how often the Solution was used to solve a problem. This value will be increased when a Ticket, to which this Solution is assigned, changes to the Status *Closed* (see *Section 5.3*).
- **Helped**: A counter that displays how often the Solution has been considered helpful by a Customer using the Web Interface (see *Section 14.3.2*).

Diary Data:

- **New Diary Entry**: A field that can be used to assign comments to the Solution. They will be added to the Diary of the Solution when saved.

Relation Viewer:

- The Relation Viewer at the bottom of the form contains references to other objects that were assigned to the Solution (e.g. Tickets).
- The page **Diary** contains information and comments for the Solution.
- The page **Comments** can be used to view and create Customer comments. Using the Web Interface, these can be answered by the Customer (see *Section 14.3.1*).

8 Inventory

This chapter describes the datasets of the Service Workbench that are responsible for the representation of object structures. These datasets represent the devices that are affected by problems. This includes the description of the objects themselves as well as their location.

When a problem is reported, the inventory gives service technicians an overview about the devices that might be involved with the problem. This includes their properties and location.

The inventory uses the following datasets:

- **Component Catalogs** describe device models (e.g. the printer model `xyz` from the producer `abc`). Component Catalogs provide general information about the model, like producer or version. The respective user documentation can also be stored within a Component Catalog dataset. More about Component Catalogs can be found in *Section 8.1*.
- **Components** describe actual existing individual objects (e.g. the printers `myPrinter1` and `myPrinter2` of Component Catalog `xyz`). They collect information about the respective object, like Name, Location and Serial Number. If a problem occurs that can be assigned to a device, the respective Ticket can be connected with the affected Component. More about Components can be found in *Section 8.2*
- **Detected Software** stores all entries that are identified as Software. If a Software with the same name is detected more than once, it will still only be added once. The datasets contain general information about the Software (e.g. wether the Software is an update or not). More about Detected Software can be found in *Section 8.3*
- **Rooms** and **Ports** describe the locations of Components or Customers (Rooms) and how Components are connected to the in-house network (Ports). More about Rooms and Ports can be found in *Section 8.5* and *Section 8.6.*

Section 8.7 describes the option to automatically collect data for the data types *Component Catalog*, *Component* and *Detected Software* and add it to the Service Workbench.

8.1 Component Catalog

Within the Service Workbench environment, Component Catalogs define model types of devices for which problems should be solved. Component Catalog datasets contain general information about the respective model type. They also contain information about the devices of this model that are in use.

8.1.1 Listing Component Catalogs

The Component Catalog data is collected within individual datasets (see *Section 12.2*) of type *Component Catalog*.

A list of all datasets can be displayed by clicking the Navigation Tree object **Component Catalog**. This corresponds to selecting the entry **Open Table View** within the context menu of the object **Component Catalog**.

Besides the Table View a Tree View is available for Component Catalogs. This can be opened by using the entry **Open Tree View** from the context menu of the object **Component Catalog**.

The Tree View provides an overview about the assignment between Component Catalogs and the known objects (Components) of the respective Component Catalog.

The list of Component Catalogs can be exported into a comma separated list (see *Section 12.5*).

8.1.2 Editing Component Catalogs

Individual Component Catalogs can be edited by using the form *Component Catalog*. It can be opened by e.g. double clicking an individual Component Catalog from the list of all Component Catalogs (see *Section 8.1.1*). It can also be opened by using the entry **Open** from the context menu of a Component Catalog object. This can be done in the Table View and the Tree View.

When a new Component Catalog is created, the same form is used.

A Component Catalog dataset contains the following information:

Base Data:

- **Name**, **Description**, **Version**: Text fields that describe the Component Catalog.
- **Type**: The types are used to classify the basic function of a Component Catalogs (like Monitors, Workstations or Switches). The individual types are freely definable and are configured as Component Catalog Types (see *Section 13.10*). Additional properties and values can be assigned to Component Catalogs Types.

Contact Data:

- **Manufacturer**, **Supplier**: The default manufacturer or default supplier of Components of this Component Catalog.

  Both are Uplink Viewer connections to an existing or newly created dataset of type *Company* (see *Section 9.2*).

Fault Management Connection:

- **FM Foreign Key**: This field is used in relation with Tickets that were automatically created by the Fault Management. It is the Fault Management object ID that is used for the back navigation (see *Section 14.5*).

Relation Viewer:

- The Relation Viewer at the bottom of the form contains references to other objects that were assigned to the Component Catalog. Here e.g. the user documentation for the Component Catalog can be provided as an Attachment.

  The page **Components** contains a listing of all Components of the current Component Catalog that are known by the Service Workbench.

8.2 Components

Within the Service Workbench environment, Components define individual devices for which problems should be solved. If e.g. three printers of a Component Type exist, three individual Components should be created to identify the individual printers.

Components may contain **Subcomponents** (like a printer or hard disk assigned to a specific PC). Subcomponents are represented as Components that are **Part Of** another Component (in the example the specific PC) which allows the creation of a hierarchical object tree.

Component datasets contain information about the respective device.

8.2.1 Listing Components

The Component data is collected within individual datasets (see *Section 12.2*) of type *Component*.

A list of all datasets can be displayed by clicking the Navigation Tree object **Component**. This corresponds to selecting the entry **Open Table View** within the context menu of the object **Component**.

Besides the Table View a Tree View is available for Components. This can be opened by using the entry **Open Tree View** from the context menu of the object **Component**.

The Tree View provides an overview about the Component hierarchy. Subcomponents are represented as child objects of the Component to which they belong.

The list of Components can be exported into a comma separated list (see *Section 12.5*).

8.2.2 Editing Components

Individual Components can be edited by using the form *Component*. It can be opened by e.g. double clicking an individual Component from the list of all Components (see *Section 8.2.1*). It can also be opened by using the entry **Open** from the context menu of a Component object. This can be done in the Table View and the Tree View.

When a new Component is created, the same form is used.

A Component dataset contains the following information:

Base Data:

- **Status**: The status of the Component. A fixed value list with the entries *New*, *Operating*, *In Stock*, *In Repair*, *Maintenance* and *Inactive*.
- **Component Catalog**: The Component Catalog that defines the object type of which the current Component is one instance.

  An Uplink Viewer connection to an existing or newly created dataset of type *Component Catalog* (see *Section 8.1*).
- **Purpose**: A selection menu with the entries from the Itemized List *Component:Purpose***.**

- **Name**, **Description**, **Serial No**, **Inventory No**, **Order No**: Text fields that describe and identify the Component.
- **Installation Date**, **Acquisition Date**, **Guaranty Till**: Date fields related to the Component.
- **Last Inventory**: A date and a time field that show the last time when the Component was inventoried. This fields will be updated during the Automated Inventory process (see *Section 8.7*).
- **Part of**: When the current Component is a Subcomponent of another Component, this can be entered here. Subcomponents of the current Component can be displayed in the Relation Viewer (see below).

  An Uplink Viewer connection to an existing or newly created dataset of type *Component* (see *Section 8.2*).

Contact and Location Data:

- **Room**: The room where the Component can be found.

  An Uplink Viewer connection to an existing or newly created dataset of type *Room* (see *Section 8.5*).
- **Location**: A selection menu with the defined Locations (see *Section 13.4*).
- **Contact Person**, **Customer**: A contact for the Component and the Customer that is assigned to be responsible for the Component.

  Both are Uplink Viewer connections to an existing or newly created dataset of type *Customer* (see *Section 9.1*).
- **Manufacturer**, **Supplier**: The manufacturer or supplier of Component.

  Both are Uplink Viewer connections to an existing or newly created dataset of type *Company* (see *Section 9.2*).

Handling Data:

- **Group**: The default service technician group that is responsible for Tickets concerning this Component.

  A selection menu with the defined Groups (see *Section 13.3*).
- **User**: The default service technician that is responsible for Tickets concerning this Component.

  A selection menu with the defined Users (see *Section 13.2*).
- **Cost Center**: The Cost Center that will be automatically used for datasets that are created for the Component (for example Tickets).

  An Uplink Viewer connection to an existing or newly created dataset of type *Cost Center* (see *Section 13.12*).

Fault Management Back Navigation

(Only available when the Service Workbench is connected to an OpenScape FM)

- **FM-Foreign-Key**: This page located in the Relation Viewer and the corresponding text field contain an automatically created reference to the object that represents the Component within the OpenScape FM (see *Section 14.4*).
- **FM-Navigation**: This main menu contains entries to navigate to the Component‘s representation within the OpenScape FM and to use the context menu of this representation (see *Section 14.5*).

Diary Data:

- **New Diary Entry**: A field that can be used to assign comments to the Component. They will be added to the Diary of the Component when saved.

Relation Viewer:

- The Relation Viewer at the bottom of the form contains references to other objects that were assigned to the Component (e.g. Attachments and Subcomponents).
- The pages **Tickets** and **Workorders** display a list of Tickets or Workorders that reference the Component as a possible source of the problem or task.
- The page **Software** displays a list of all software installations manually recorded for this Component. When **Show Updates** is checked, the installed updates are also listed.
- The pages **Diary** and **Status Transitions** keep track of comments and status changes for the Component.

8.2.3 Replacing Components

Components can be replaced by new Components without the need to recreate references and underlying Component hierarchies manually.

This can be done by selecting the Component that should be replaced within the list of Components (see *Section 8.2.1*). Then the entry **Replace** has to be selected from the main menu **Dataset**.

This will open a wizard which guides through the various replacement options. E.g. it can be specified whether Subcomponents should be transferred or whether the old Component should be deleted.

8.2.4 Component Templates

Similar to the Ticket Templates, Components can also be saved as templates and re-used. This is handled respective to the methods described in *Section 5.6* for Ticket Templates.

8.2.5 Creating Multiple Components

Multiple Components can be created using a wizard. This wizard can be started by using the entry **Create many Components** from the context menu of the Navigation Tree object **Component**.

The wizard provides the following steps to configure the new Components:

- **Template**:
  A Template can be selected for the Components that should be created. The available Templates are listed and detailed information about the individual Templates can be displayed in tooltips.
- **Supplier / Manufacturer**:
  Further information about the new Components can be entered. This includes e.g. the Manufacturer, the Supplier and the Acquisition Date.

- **Attachments**:
  Attachments that should be added to the new Components can be selected here.
- **CSV-Import**:
  This page can be used to select an input file and to configure the format to generate Components based on a comma separated value (CSV) list. The left column displays the headers detected in the CSV-File. The right column can be used to select the Component elements that should be filled with the content of the respective column.
- **Amount and Name of Components**:
  The new Components unique names can be assigned here. The names are generated based on a **Prefix** and a **Start Number**. The field **Format Preview** shows an exemplary name based on the entered values.
- Pressing the button **Finish** will create the new Components.

8.3 Software

Within the Service Workbench environment, Software defines software versions for which problems should be solved. Software datasets contain general information about the respective Software. They also contain information about the devices on which the Software has been installed.

8.3.1 Listing Software

The Software data is collected within individual datasets (see *Section 12.2*) of type *Software*.

In addition Extended Searches for Software (see *Section 12.2.3.3*) are provided. The results of the searches will be displayed if the respective entry located below the entry **Software** within the Navigation Tree is clicked. This corresponds to selecting the entry **Open Table View** within the context menu of the respective object.

The predefined Extended Search for Software is:

- **All**: All Software datasets.

The list of Software can be exported into a comma separated list (see *Section 12.5*).

8.3.2 Editing Software

Individual Software can be edited by using the form *Software*. It can be opened by e.g. double clicking an individual Software from the list of all Software (see *Section 8.3.1*). It can also be opened by using the entry **Open** from the context menu of a Software object.

When a new Software is created, the same form is used.

A Software dataset contains the following information:

Base Data:

- **Name of Software**: Text field that defines the name of the Software. This name is used in lists and selection fields and should be intuitive and unique.
- **Description**: Text field that should contain a short description of the Software.
- **Update**: A Checkbox, that defines whether the Software is an update of a Software. Updates can be handled differently or might be hidden in some tables or reports.
- **Source**: This text field displays whether the Software has been created manually (*Manual*) or by an SQL data source (*SQL-Sync*). More about different import methods can be found in *Section 8.7*. A Software can only be modified, if it has the Source *Manual*.

Relation Viewer:

- The page **Components with this Software** displays a list of Components for which an instance of this Software has been found. If e.g a problem for a specific Software is known, this list will display the Components that are affected.
-

8.4 Software Instances

Within the Service Workbench environment, Software Instances define individual Software installations for which problems should be solved. If e.g. three installations of a specific Software exist, three individual Software Instances should be created to identify the individual installations.

Software Instance datasets contain information about the respective software installation.

8.4.1 Listing Software Instances

In relation to other types of datasets, a large number of Software Instances may exist. For this reason no overview about all instances will be offered within the Navigation Tree. This list would usually contain a large number of repetitions (e.g one entry per installation of an operating system).

Lists of Software Instances are only displayed for individual Components. The page **Software** within the Relation Viewer shows a list of all installations that have been performed for the current Component.

8.4.2 Creating Software Instances

New Software Instances are usually created by automatic inventory functions (see *Section 8.7*).

Manually, Software Instances can only be created for Components. This can be done by using the menu entry **New->Software** from the main menu **File** within the form of the respective Component.

8.4.3 Editing Software Instances

Individual Software Instances can be edited by using the form *Software Instance*. It can be opened by double clicking an individual Software Instance from a list of Software Instances (see *Section 8.4.1*).

When a new Software Instance is created, the same form is used.

A Software Instance dataset contains the following information:

Base Data:

- **Product Name**: The name of the installed Software.

  An Uplink Viewer connection to an existing or newly created dataset of type *Software* (see *Section 8.3*).
- **Manufacturer**, **Version**: Text field for the name of the Manufacturer and the Version of the Software, that have been installed on the Component.
- **GUID**: Text field for the global unique ID of the Software.
- **Name of Executable File**, **Installation Path**: Text fields for the location of the Software (file name and storage location) within the file system of the Component.
- **Installation Date**: A date field for the data of the installation.

8.5 Rooms

Within the Service Workbench environment Rooms represent places to which Customers and Components can be assigned. They are used to define associations in respect of locations. Rooms provide an overview about the relations between Customers and devices (Components) and how they are connected by the in-house network.

Often Rooms will be actual physical rooms. But a Room might also represent a staff car or a building unit.

Room datasets provide a quick overview about the topological structure of the Company.

8.5.1 Listing Rooms

The Room data is collected within individual datasets (see *Section 12.2*) of the type *Room*.

A list of all datasets can be displayed by clicking the Navigation Tree object **Room**. This corresponds to selecting the entry **Open Table View** within the context menu of the object **Room**.

The list of Rooms can be exported into a comma separated list (see *Section 12.5*).

8.5.2 Editing Rooms

Individual Rooms can be edited by using the form *Room*. It can be opened by e.g. double clicking an individual Room from the list of all Rooms (see *Section 8.5.1*). It can also be opened by using the entry **Open** from the context menu of a Room object.

When a new Room is created, the same form is used.

A Room dataset contains the following information:

Base Data:

- **Number**: The number or unique id of the Room.
- **Name of Room**, **Building**, **Floor**: Text fields that describe the Room.
- **Purpose**: The purpose of the Room (e.g. *Office*, *Conference Room* oder *Printer Room*).

  A selection menu with the entries of the Itemized List *Room:Purpose*.
- **Location**: A selection menu with the defined Locations (see *Section 13.4*).

Network Data:

- **Network Switch**, **Telephone System**: Both selection menus define the Room‘s connections to the in-house network.

  Network Switch is a selection menu with the entries of the Itemized List *Room:Switch*.

  Telephone System is a selection menu with the entries of the Itemized List *Room:Pbx*.
- **Ports**: A page within the Relation Viewer. It contains a list of the Ports that are located within the Room (see *Section 8.6*).

Work Data:

- **Cost Center**: The Cost Center that is assigned to the Room. This will e.g. be used for janitor work orders.

  An Uplink Viewer connection to an existing or newly created dataset of type *Cost Center* (see *Section 13.12*).
- **Max. Workplaces**: The maximum number of persons (Customers) that can be assigned to the Room without getting the Room overstaffed (see the following field).
- **Overstaff**: If this is checked, the number of Customers assigned to the Room may get larger then the defined number of workplaces. Else a warning will be displayed when the assignment of a Customer to the Room makes the Room overstaffed.

Relation Viewer:

- The Relation Viewer at the bottom of the form contains references to other objects that were assigned to the Room (e.g. Customers and Components).

8.6 Ports

Within Service Workbench environment **Ports** represent network and phone connectors.

These can be assigned to a Room and a Component. They show the service technicians how the individual Components are connected by the in-house network.

8.6.1 Listing Ports

Ports can only be listed for one Room at a time (see *Section 8.5.2*).

The listing for a single Room is found on the page **Ports** within the Relation Viewer of the form Room. On this page Ports can also be created or deleted.

8.6.2 Editing Ports

Ports can be edited using the form *Port*. This can be opened e.g. from within the listing of Ports for a Room (see *Section 8.6.1*) by double clicking an individual Port.

When a new Port is created, the same form is used.

A Port dataset contains the following information:

Base Data:

- **Name**: The Name of the Port.
- **Type**: The Port type (e.g. *Ethernet [100MBit]*).

  A selection menu with the entries of the Itemized List *Port:Type*.
- **Room**: The Room in which the Port can be found.

  An Uplink Viewer connection to an existing or newly created dataset of type *Room* (see *Section 8.5*).

Network Data:

- **Backplane**: The backplane of the Port. For example, this can be the subnet to which the port is connected.

  A selection menu with the entries of the Itemized List *Port:Backplane*.
- **Component**: The Component that is connected to the Port.

  An Uplink Viewer connection to an existing or newly created dataset of type *Component* (see *Section 8.2*).

8.7 Automated Inventory

Die Service Workbench allows the automatic and manual registration of Components with additional hardware and software information. For this the dataset types *Component Catalog*, *Component* and *Software* are used.

With the interface (Event Gateway) to the Fault Management, objects managed by the Fault Management can be automatically transferred to the Service Workbench as Component Catalogs and Components.

With the interface SQLSync it is possible to automatically collect Components with hardware and software from database content. The interface can be invoked by using the entries from the menu **System Management Sync** from the main menu **Synchronisation**.

More about this can be found in the separate *Administrator User Guide*.

9 Contacts

Within the Service Workbench contact data will be stored as Customer datasets (see Section 9.1). These can be assigned to individual Companies (see Section 9.2).

9.1 Customers

Within the Service Workbench environment, a **Customer** is a person that submits problems with the goal that they are solved by service technicians.

The problem messages sent by the Customers are stored within the Service Workbench database as Tickets (see *Chapter 5*). Using the Web Interface (see *Chapter 14*), these can be created directly by the Customer. Alternatively a service technician can create them for the Customer (e.g. after a phone call).

The current state of their problem tickets can be seen by Customers using the Web-Interface (see *Chapter 14*).

9.1.1 Listing Customers

The Customer data is collected within individual datasets (see *Section 12.2*) of the type *Customer*.

A list of all datasets can be displayed by clicking the Navigation Tree object **Customer**. This corresponds to selecting the entry **Open Table View** within the context menu of the object **Customer**.

The list of Customers can be exported into a comma separated list (see *Section 12.5*).

9.1.2 Editing Customers

Individual Customers can be edited by using the form *Customer*. It can be opened by e.g. double clicking an individual Customer from the list off all Customers (see *Section 9.1.1*). It can also be opened by using the entry **Open** from the context menu of a Customer object.

When a new Customer is created, the same form is used.

A Customer dataset contains the following information:

Base Data:

- **Status**: A selection menu for the Status of the Customer (*Active*, *Inactive*, *Discarded*, *External*). Currently the Status has only informative value.
- **Gender**: A selection menu (*male* or *female*).
- **Title**: A selection menu with the entries of the Itemized List *Customer:Academic Title*.
- **Personnel No.**: An identification number assigned to the Customer.
- **Short Name**: A letter code assigned to the Customer.

- **Academic Title**: A selection menu with the entries from the Itemized List *Customer:Title*.
- **Last Name**, **First Name**, **Street**, **ZIP Code**, **City**: Text fields that describe the Customer.
- **Country**: A selection menu with the entries from the Itemized List *Company:Country*.

Contact Data:

- **Phone**, **Mobile**: Selection lists with the landline and mobile numbers. These lists can be expanded. If configured, the respective button can be used to initiate a call to the selected number.
- **Facsimile** and **Pager**: Text fields with the respective data.
- **Mail**: The email address of the Customer. By using the button, an email can be generated for the defined address.
- **Receive automatic generated mails**: If this is checked, the Customer will get informative email notifications from the Service Workbench. This is for example the case, when the Status of a Ticket changes.

Web Interface Access Data:

- **Use Service Portal**: If this is checked, the Customer can use the Web Interface to create and monitor Tickets (see *Chapter 14*).
- **Login** and **Password**. Text fields with the access data for the Web Interface.

  If the Web Access has been configured to use the Windows Domain Server, the Password field is deactivated.

Company Data:

- **Company**: The Company to which the Customer is assigned.

  An Uplink Viewer connection to an existing or newly created dataset of type *Company* (see *Section 9.2*).
- **Department**: A selection menu with the entries from the Itemized List *Customer:Department*.
- **Position**: A selection menu with the entries from the Itemized List *Customer:Position*.
- **Special Field** and **Profession**: Two unbound text fields to describe the Customer.
- **Location**: A selection menu with the defined Locations (see *Section 13.4*).
- **Preferred Language**: The language that will be used by the system to communicate with the Customer (e.g. within automatic emails or reports). Currently supported are English and German.
- **Cost Center**: The Cost Center that will be automatically used for datasets that are created for the Customer (for example Tickets).

  An Uplink Viewer connection to an existing or newly created dataset of type *Cost Center* (see *Section 13.12*).
- **Room**: It will be checked automatically, if *Overstaff* is allowed for the Room. If Overstaff is not allowed and the assignment exceeds the maximum number of work spaces, a warning message will be displayed (see *Section 8.5*).

  An Uplink Viewer connection to an existing or newly created dataset of type *Room* (see *Section 8.5*).
- **1st Superior**, **2nd Superior** and **Secretary**: Uplink Viewer connections to existing or newly created datasets of type *Customer* (see *Section 9.1*).

- **1st Superior Group** and **2nd Superior Group**: Uplink Viewer connections to existing or newly created datasets of type *Group* (see *Section 13.3*).
- **SLA**: The standard service agreement that will always be used when a ticket, for which no individual SLA has been assigned, is assigned for the Customer.

  A selection list with the defined SLAs (see *Section 13.8*).

Diary Data:

- **New Diary Entry**: A field that can be used to assign comments to the Customer. They will be added to the Diary of the Customer when saved.

Relation Viewer:

- The Relation Viewer at the bottom of the form contains references to other objects that were assigned to the Customer (e.g. Tickets and Components).
- The pages **Diary** and **Status Transitions** keep track of comments and status changes for the Customer.
- On the page **Private**, further confidential information about the Customer can be accessed. For example, this data includes the *Birthday* or the *Private Address*.

  Hint:
  This page will only be displayed if the property **Access customer's personal data** has been enabled for the current User (see *Administrator Documentation*).
- The pages **Tickets** and **Workorders** contain a list of the Tickets and Workorders that are assigned to the Customer.

9.1.3 Creating a User from a Customer

In some cases it may be desired to let an already existing Customer work as a User.

In this case an *Administrator* can use the entry **Create User from Customer** from the menu **Extras** within the form of the respective Customer.

This entry generates a new User based on the data of the current Customer. E.g. the Name, Location and Email Address will be copied into the new dataset.

9.2 Companies

Within the Service Workbench environment Companies are units to which Customers (the problem reporter) can be assigned. These can be actual companies or any other organizational unit to structure the Customer base.

Company datasets provide a quick overview about the datasets that have been assigned to the Company or to one of its staff (Customers).

9.2.1 Listing Companies

The Company data is collected within individual datasets (see *Section 12.2*) of type *Company*.

A list of all datasets can be displayed by clicking the Navigation Tree object **Company**. This corresponds to selecting the entry **Open Table View** within the context menu of the object **Company**.

Besides the Table View a Tree View is available for Companies. This can be opened by using the entry **Open Tree View** from the context menu of the object **Company**. The Tree View displays the structure of the Companies regarding the Subcompany relations.

The list of Companies can be exported into a comma separated list (see *Section 12.5*).

9.2.2 Editing Companies

Individual Companies can be edited by using the form *Company*. It can be opened by e.g. double clicking an individual Company from the list off all Companies (see *Section 9.2.1*). It can also be opened by using the entry **Open** from the context menu of a Company object. This can be done in the Table View and the Tree View.

When a new Company is created, the same form is used.

A Company dataset contains the following information:

Base Data:

- **Name**, **Add On 1**, **Add On 2**, **Add On 3**, **Street**, **ZIP Code**, **City**: Text fields that describe the Company.
- **Country**: A selection menu with the entries from the Itemized List *Company:Country*.
- **Location**: A selection menu with the defined Locations (see *Section 13.4*).
- **Role**: A selection field that describes the function of the Company. By default the values *Manufacturer* and *Supplier* are available.

  A selection menu with the entries from the Itemized List *Company:Role*.
- **Subcompany from**: This field makes it possible to assign the current Company as a Subcompany of another defined Company. This relation will be displayed in the Tree View of the Company overview (see above).

  An Uplink Viewer connection to an existing or newly created dataset of type *Company* (see *Section 9.2*).
- **Central Contact Person**: The central contact person within the Company.

  An Uplink Viewer connection to an existing or newly created dataset of type *Customer* (see *Section 9.1*).
- **Default Cost Center**: Defines the Cost Center for new Customers of the Company and for Customers of the Company to which no individual Cost Center has been assigned.

  An Uplink Viewer connection to an existing or newly created dataset of type *Cost Center* (see *Section 13.12*).

Contact Data:

- **Phone**: Selection list with the landline numbers. This list can be expanded. If configured, the respective button can be used to initiate a call to the selected number.

- **Mail**: The email address of the Company‘s Central Contact Person. By using the button, an email can be generated for the defined address.
- The text fields: **Fax** and **URL**.

Diary Data:

- **New Diary Entry**: A field that can be used to add comments to the Company. They will be copied to the Diary of the Company when the dataset is saved.

Relation Viewer:

- The Relation Viewer at the bottom of the form contains references to other objects that were assigned to the Company (e.g. Customer and Tickets).
- The page **Diary** keeps track of comments for the Company.

10 Actions

Within the Service Workbench, Actions are manually or automatically started processes that perform certain predefined process steps. These can be used to support service technicians by e.g. automatically collecting needed data, or by providing often needed functions.

There are two types of Actions within the Service Workbench:

- **Workflow Actions** are performed by the Service Workbench Server for a specific Service Workbench object (see *Section 10.1*). They are triggered manually by a User that handles the object or automatically when the object is created, modified or deleted.
- **Client Actions** are performed on the system on which the Service Workbench client is running (see *Section 10.2*). Each User can configure and define Client Actions for himself. Administrators can define Client Actions that are also available for other Users.

Important Note:
If an Action should affect an OpenScape FM object, the object ID of the respective OpenScape FM object has to be entered as a *Foreign Key* within the Service Workbench object for which the action should be performed (siehe *Section 14.4*).

10.1 Workflow Actions

Workflow Actions are actions that can be manually or automatically started on the Service Workbench server to perform tasks for individual Service Workbench objects.

Workflow Actions are defined and configured within the OpenScape FM Event Correlation Plugin and, using the ECE workflow, can perform virtually any possible task.

Three types of Workflow Actions are provided within the Service Workbench:

- **Run-Through Workflow Actions** are manually started for individual Service Workbench objects and need no further input by the User (see *Section 10.1.1*).
- **Step-by-Step Workflow Actions** are manually started for individual Service Workbench objects and can be manipulated by further input of the User (see *Section 10.1.2*).
- **Automatic Workflow Actions** are automatically started for individual Service Workbench objects if the respective object is created, deleted or modified (see *Section 10.1.3*).

Important Note:
The OpenScape FM Event Correlation Plugin is needed to perform Workflow Actions.

Necessary Access Rights:

Workflow Actions are performed within the OpenScape FM for the current user of the Service Workbench. To perform a Workflow Actions the current user of the Service Workbench needs to have at least OpenScape FM *Operator* rights for the *ECE Workflow* nodes that are involved by the respective Action. If the rights are insufficient, the Action cannot be performed and the entry to start the Action is not displayed within the Service Workbench.

If for example the Action to manage a specific IP node should be allowed, *Operator* rights for the *ECE Workflow* node
`ECE => Workflow => SWB Menu entries & ticket handling => IP Node => IP Node Actions => Manage IP Node`
are needed for the respective User.

By reducing or granting rights for individual *ECE Workflow* nodes and individual Users, the access to specific Actions can be individually configured within the OpenScape FM.

10.1.1 Run-through Workflow Actions

Run-through Workflow Actions are Workflow Actions that, once they are started, require no additional input by the User. They can be manually started by a User for an individual Service Workbench object and are started on the Service Workbench server.

If at least one Run-through Workflow Action is available for an object, the main menu **Workflow Actions** will be shown in the object‘s form. This menu contains one entry for each Run-through Workflow Action that is available for the current object.

When a menu entry that represents a Run-Through Workflow Action is selected, the respective action will be performed for the current object.

Run-through Workflow Actions may be available for objects of the type: *Ticket*, *Workorder*, *Component Catalog* or *Component*.

10.1.2 Step-by-Step Workflow Actions

Step-by-Step Workflow Actions are Workflow Actions that require additional input by the User. They are performed on a step-by-step basis while the User decides by multiple choice selection which step should be performed as the next step. Each step performs actions that have been defined within the OpenScape FM Event Correlation Plugin.

Step-by-Step Workflow Actions can be manually started by a User for an individual Service Workbench object.

If at least one Step-by-Step Workflow Action is available for an object, the page **Workflow Action** will be shown within the object‘s form. This tab contains elements to select the desired Step-by-Step Workflow Action and to select the steps that should be performed next for the current object.

Step-by-Step Workflow Actions may be available for objects of the types: *Ticket* or *Workorder*.

10.1.3 Automatic Workflow Actions

Automatic Workflow Actions are actions that are automatically performed if an OpenScape FM object of a specific type is created, modified or deleted.

The Automatic Workflow Actions have to be build within the Event Correlation Engine. For each object type and for each event type (creation, modification or deletion of an object) individual Automatic Workflow Actions can be defined within the ECE Workflow. If a more detailed handling is needed, specific filters can be integrated into the ECE Workflow.

The combinations of object types and event types that should actually be active and should initiate a handling within the ECE, can be defined by an Administrator within the Service Workbench.

More about the definition of Automatic Workflow Actions and the configuration of the active object types and events can be found in the *Service Workbench Administrator Guide* and the *OpenScape FM Event Correlation Engine Plugin User Guide*.

10.1.4 Predefined Workflow Actions

By installing the OpenScape FM a number of predefined Workflow Actions for objects of the types *Tickets* and *Components* will be created.

These provide general functions and functions for the HiPath/OpenScape 4000 Plugin, IP Manager Plugin and System Management Plugin.

As usual, the predefined Actions can be found within the menu *Actions* (Run-through Workflow Actions) or on the page *Workflow Action* (Step-by-Step Workflow Actions), if a form for the respective type is opened that contains a matching object of the respective plugin.

Using the Event Correlation Engine Plugin, additional actions can be designed by an Administrator.

10.2 Client Actions

Client Actions are actions that can be manually started on the Service Workbench client to perform a predefined task (like performing a ping to the IP node assigned to a current ticket, or the opening of a system shell window).

Client Actions are commands that are started on the local client system. They are started from within the menu **Actions** from the main menu of the form that represents the object for which they should be started. The menu *Actions* will only be displayed when at least one Client Action has been defined for the current object type. Possible object types are: *Ticket*, *Workorder*, *Solution*, *Component Catalog*, *Component*, *Company*, *Customer* and *Room*.

Each Client Action that can be performed for an object that is currently displayed by a form, can be started by a respective entry within the menu *Actions* of this form.

Client Actions can be defined and configured by the User of the client and will be stored for future use. Administrators can also define Client Actions that can be started by other Users.

The definition and configuration of Client Actions is started with the entry **External Actions** from the main menu **Configuration** within the Service Workbench central window.

This entry opens the configuration window in which the tab **My Actions** can be used to define Client Actions for the current User. Administrators can use the tab **System Actions** to define Client Actions to all Users. Both definitions are identically handled.

Within the configuration window the selection menu **Object Type** is used to select the object type for which Client Actions should be configured. The list then displays all actions that are currently configured for this type and the buttons **Add**, **Edit** and **Remove** can be used to create a new action for this object type, or to edit or delete an entry that is selected within the list.

If *Add* or *Edit* is selected the configuration window for the specific Client Action opens.

Within this window the following parameters are available:

- **Object Type**: A read only field that displays the object type for which the current Client Action should be provided.
- **Name**: The label that will be displayed for this action within the menu *Actions*.
- **Operating System**: The operating system for which this action will be available. If the client is running on another operating system, the action will not be shown in the menu *Actions*.
- **Command**: The command that will be started when the respective entry is selected from the menu *Actions*.
- **Parameter**: The parameters for the Command that will be executed for the action. Each parameter has to be defined within an individual row.

  Like for Emails (see *Section 12.8.2*), variables and relation templates can be selected from the context menu to add variable elements to the parameters.
- **Execute in Shell**: If this is checked, the command will be performed in its own shell.
- **Kill on Exit**: If this is checked, the command will be terminated when the OpenScape FM client is closed.
- **Use for all Objects**: If this is checked, the action will be offered for all objects of the correct type. If unchecked, individual objects for which the action should be provided can be defined using an object search.

11 Relation Viewer

The Relation Viewer contains data associated/linked with the dataset. It is part of the form and is always located at the bottom of the form.

For example, for Tickets, amongst other data the associated Workorders are displayed as a list. Diary entries and file attachments are also displayed in the Relation Viewer.

If a Relation Viewer page offers a tabular view of linked datasets, they can be opened by double-clicking the respective entry.

The number of currently linked datasets is always indicated by a tooltip when hovering the mouse cursor over the page name.

11.1 Relations

Relations are logical links between two dataset elements. Usually but not necessarily between different types of datasets.

Exemplary relations are:

- A Component to Component Catalog relation defines that a Component is an instance of a specific Component Catalog.
- A Ticket to Component relation defines that a Ticket describes a problem for a specific Component.
- A Component to Component relation defines that a Component is a Subcomponent of another Component.

Relations within the Service Workbench are One-To-Many relations.

Using the second example from above, this means that while each Ticket only references one specific Component (the one that has the specific problem), this Component itself can be referenced by any number of Tickets (from all Tickets that describe problems for this Component).

Using the third example from above, this means that while each Component may be a Subcomponent of at most one other Component, in contrast a Component may have as many Subcomponents as desired.

Independent of the types of datasets involved in a relation, the handling of the relations is always identical:

- **Uplink-Viewers** (see *Section 12.3*) are used to display *To-One* links on the dataset form of the object that relates to one object. The Uplink-Viewer can also be used to change the link or to create a new object which will then be linked immediately.

  Example: The Uplink-Viewer *Affected Component* in the Ticket form is used to display, create or change a link to the affected Component.
- Respectively named **Relation Viewer tabs** are used to display the *To-Many* links for objects that may relate to many objects. These tabs display the opposite direction of the links defined by the Uplink-Viewers described above.

Each page contains the list of all related objects. Double-clicking entries of the list will open a form for the respective object.

Example: For Components, the Relation Viewer page *Ticket* will list all Tickets assigned for the Component and the Relation Viewer page *Subcomponents* will list all its Subcomponents.

- Usually Uplink-Viewers are used to define links between objects of the Service Workbench. An exception is the creation of new objects. In this case the dataset to which a *To-One* link should be created can be opened and the respective entry of the menu **New** within the main menu **File** can be used. This will open a form to create a new dataset of the selected type in which the Uplink-Viewer is already linked to the dataset from which the menu entry has been activated.

  Example: A new Component should be created for an already existing Component Catalog. In this case, the appropriate Component Catalog can be opened. If the entry **New->Component** is selected from its form, a form to create a new Component is opened. In this form the Uplink-Viewer *Component Catalog* is already linked appropriately.

11.2 Diary

For most dataset types (Ticket, Workorder, Customer, Company, Component, Solution) the page **Diary** within the Relation Viewer can be used to get a historic listing of comments made by the System (automatically created for major events, like creating a Ticket or omitting a Workstep) or by Users.

For most dataset types listed above, User comments can be added to a dataset by filling the field **Diary Entry** or **New Diary Entry** and saving the dataset. This will add the content of the field to the list of diary entries and empty the field itself.

For Tickets the Relation Viewer page **New Diary Entry** can be used to enter a User comment.

11.3 Status Transitions

For dataset types to which a **Status** can be assigned (Ticket, Workorder, Customer, Component), the page **Status Transition** within the Relation Viewer can be used to see the historic changes of the Status.

11.4 Costs

The page **Costs** within the Relation Viewer can be used to add costs to the respective dataset of most dataset types. The buttons on the right side of the page can be used to add, delete of modify the cost data.

11.5 E-Mails

Emails can be attached to most types of datasets (e.g. Tickets, Components). The attached emails can be viewed in the page **E-Mail** within the Relation Viewer.

11.6 Extension Fields

Forms that contain a Relation Viewer also have the page **Extension Fields** within the viewer. On the Extension Fields page, there are text fields into which up to 250 characters can be entered each.

These fields are used to add information that does not match to the predefined fields. An *Administrator* can change the names of these fields.

For example, one field may be called *Costs* and the field below could be called *Currency*.

11.7 Attachments

An attachment is any URL, file path or file assigned to a dataset. An attachment file can be uploaded directly from the Service Workbench client and it will be stored centrally in the database. Therefore the uploaded files can also be opened from other places. Attachments are useful to provide additional data delivered by a Customer.

For example, it is possible to add log-files or screenshots that better illustrate a problem.

Important Note:
Attachments can only be added to datasets that have been saved at least once.

As many attachments as desired can be added to a dataset. This can be done by using page **Attachments** in the Relation Viewer of the respective form. The maximum size of the file that can be uploaded depends on the configuration of the Service Workbench system (default setting: 5 MB).

The attachment can be saved directly to the Service Workbench, or just the path to the file can be saved, which is useful for very large files. Web-based URLs can be saved by entering the URL.

The field **Label** can be used to define a name for a path or an URL. This name is used within the table for the attachments instead of a long path or a long URL. The actual path or URL is displayed as a tooltip.

When opening an attachment represented by a file path, the system's file explorer is used to open the file path. With URLs, the system's web browser is used to open the corresponding URL. Paths that contain embedded Windows network drives are always replaced with the UNC notation.

Example:
`X:\Temp\screenshot.jpg`
is replaced by
`\\remote-pc\share\Temp\screenshot.jpg`

12 General Functions

This chapter covers basic functions that are used for a number of different dataset types.

12.1 Sorting Tables and Selecting Columns

At various places within the Service Workbench, tables are used to list entries.

To sort the entries in ascending order by the desired column, the appropriate column header has to be clicked once. To sort the entries in descending order, the column header has to be clicked again.

For the main data tables, which are displayed in the right-hand section of the main window, it can be chosen which columns are to be shown. Right-clicking on the table header opens a context menu in which all the relevant data fields for the selection can be found.

The arrangement of the table columns and their width can be changed using Drag&Drop. All changes are stored individually for the User and will be displayed when the table is opened again.

12.2 Datasets

Datasets represent the various problems, tasks and objects stored in the database of the Service Workbench. For example: each Ticket is represented by an individual dataset. Since different types of information are stored for different object types, each type has its own dataset type (e.g. a dataset type for Tickets, another for Customers, and so on). The different dataset types are described in the respective sections of the previous chapters.

This section describes the functions that are common among the different dataset types.

12.2.1 Creating Datasets

There are three basic methods to create datasets for the various dataset types. Generally the methods for the different datasets are identical or at least very similar:

Independent dataset:

The main window contains the button **New** located below the main menu. If the button **New** is pressed while a dataset type is selected within the Navigation Tree (e.g. the entry *Ticket* or *Customer*), an empty form for the respective dataset type will be opened.

Alternatively, the context menu of the button **New** can be used to open an empty form of the selected dataset type.

Pressing the buttons **Save**, **Save and Close** or **Save and New** or selecting the respective entries within the form‘s menu **File** will create a new dataset with the current data.

Save and New also opens a new empty form to create an additional dataset.

Dataset for current Form:

Forms displaying datasets provide the menu **New** within their main menu **File**. The entries from this menu each open a form for the respective dataset type and relate the new dataset to the dataset from which the form was opened.

The opened form is the same form as the one opened for an independent dataset of the same type (see above), but some fields are pre-filled to correlate to the dataset from which the form was opened.

Dataset for a Link:

Most dataset forms provide Uplink Viewers (see *Section 12.3*) to define links to other datasets. The entry **New** from the button of the Uplink Viewer opens a blank form of the referenced dataset type to create a new dataset. When this dataset is saved, it is linked with the current dataset.

12.2.2 Displaying Datasets

The datasets are generally listed in tables (e.g. when a dataset type is selected in the Navigation Tree). Double-clicking an entry of the table opens the respective dataset in a separate form.

Alternatively, the context menu entry **Open** can be used. The same menu entry is also displayed in the main menu **Dataset**.

The Service Workbench can be configured by an *Administrator* to generate a warning message, if a dataset gets opened that is already open for another User. This feature can be used to avoid the editing of one dataset by two Users at the same time.

The menu entry **Open** provides the option to open multiple datasets simultaneously. The desired datasets have to be selected in the table and then the menu entry **Open**, from either the main menu or from the context menu, has to be selected. A separate form will then be opened for each dataset.

12.2.3 Searching Datasets

Searches can be performed to restrict the displayed datasets to the ones that match the search criteria. There are three search types that are described in the following subsections:

12.2.3.1 Simple Search

When a dataset table is displayed, the toggle button **Search** determines whether the search area will be displayed. The area can also be opened or closed by using the entries **Open Search** and **Close Search** from the main menu Edit. The search area contains the field **Search for** and a respective button.

When the button is pressed, a search is performed that displays all datasets of the current type that contain the entered text as a substring in one of the fields listed in the top left section of the area. The search is *not* case sensitive.

12.2.3.2 Fulltext Search

The Fulltext Search performs the Simple Search described above simultaneously for all dataset types. For each type the same fields are considered that are used for a Simple Search of this type.

The Fulltext Seach is performed when a search text is entered into the field left of the button **Fulltextsearch** and the button is pressed.

The result list may contain datasets of various dataset types.

12.2.3.3 Extended Search

If the button **Extended Search** is pressed within the area for Simple Searches (see *Section 12.2.3.1*), a window opens in which an individual search can be configured. The window can also be opened by using the entry **Extended Search** from the main menu **Edit**.

Within the window, search conditions for various fields of the dataset can be defined and combined with each other.

The button **Search** within the configuration window performs the search.

The button **Save search** stores the search for the current User. A name can be assigned to the search and it will be displayed in italic and below the matching dataset type within the Navigation Tree. If this entry is selected, the search will be performed again.

Users with *Administrator* rights may define Searches for all Users.

12.2.4 Duplicating Datasets

Datasets can be replicated by selecting the respective dataset in a dataset table and using the entry **Duplicate** from its context menu or by using the entry **Duplicate** from the main menu **Dataset**.

For open dataset forms, the entry **Duplicate** of the main menu **File** can be used to replicate the current dataset.

Both actions will create a copy of the selected or current dataset and display it in a new form. If the new dataset includes a Status field, the Status will be set to *New*.

12.2.5 Deleting Datasets

If one or more datasets are selected in a dataset table, they can be removed from the system by using the context menu entry **Delete**.

Alternatively, the respective entry can be used from the main menu **Dataset**.

Only users who have *Administrator* rights are allowed to delete datasets.

12.2.6 Editing Multiple Datasets

Multiple datasets of the same type can be edited simultaneously by selecting them in a dataset table and using the entry **Edit Multiple** from their context menu.

This opens a window that contains one row per dataset field.

Only those fields will be edited for the selected datasets for which the respective row is checked.

For every selected field, a value can be entered behind the field name. This will set the respective field to the entered value for all selected datasets.

If the checkbox **Reset** is checked, the respective field will be emptied for all selected datasets.

The button **Next** provides an overview about the changes based on the current selection.

The button **Finish** performs the changes.

12.3 Uplink Viewer

The Uplink Viewer is used to link datasets with one another, to view links or to remove links. Uplink Viewer are always found behind text fields whose content represents data of a referenced dataset.

The Uplink Viewer consists of a button with a selection function. Therefore a number of actions can be performed with this button:

The entry **New** opens a blank form of the referenced dataset type to create a new dataset. When this dataset is saved, it is linked with the current dataset.

The entry **View** allows to view the details of the currently linked dataset. The form for the relevant dataset is opened.

The entry **Search** opens a mask that can be used to search existing datasets according to the associated text field.

The entry **Remove Link** can be used to delete the link between the current dataset and the dataset shown in the field.

View and **Remove Link** are only active if a dataset is currently linked.

12.4 Printing

The Service Workbench offers a print function to print out information shown in the main window or in tables.

Printing Previews:

Previews displayed in the main window (including overviews for Tickets and Workorders) can be printed by using the entry **Print Preview** from the main menu **File**.

Tickets and Workorders can also be printed as Reports (see *Section 5.11* and *Section 6.9*).

Printing Tables:

If a dataset table is displayed in the right-hand section of the main window, this table can be printed by using the entry **Print Table** from the main menu **File**. The table will be printed exactly as it appears in the application.

12.5 Exporting Tables

If a table is displayed in the right-hand section of the main window, then for most data types (e.g. Customers, Components) the content of this table can be exported as a CSV file. This can be started by selecting the entry **Export Table** from the main menu **File**.

Prior to the export, the settings like the storage location, the coding or additional CSV configurations can be assigned. The columns and datasets that are currently displayed in the application will be exported.

12.6 Datasets Filtering

Generally, all datasets can be seen by all Users. The Service Workbench provides the option to restrict the viewing of datasets based on Locations or Group membership.

This function has to be activated by an Administrator (see *Administrator User Guide*). The Info dialogue (see *Section 12.11.1*) can be used to check, if this function is active.

Location Filters:

If the Location Filter is active, Users with the role *Staff* or *Staff [R/O]* can only view datasets that are assigned to a Location to which they are assigned. The dataset may be assigned directly or indirectly to a Location.

For example: a User can only see Components that are assigned to one of his locations. In contrast, the visibility of a Ticket depends on the Location of the Customer for which the Ticket was opened.

The following datasets are only visible if a Location is directly assigned to the dataset:

- Customer
- Component
- Company

The following datasets are visible depending on the Location of another dataset:

- Ticket: depending on the Location of the Customer
- Workorder: depending on the Location of the Customer for the assigned Ticket

If no information about the Location is available or if the dataset type is not included in either of these two lists, then a dataset is visible to all Users. For example, a Workorder without an assigned Ticket is visible to all Users.

A special case are datasets that are directly assigned to a User. In this case, the User will be able to view the dataset independent of the Location.

Group Filters:

If the Group Filter is active, Users in the role *Staff* or *Staff [R/O]* will only be able to view datasets assigned to themselves or to a Group they are belonging to. The Group Filter is only active for Tickets, Workorders and Components.

If no User or Group is assigned to a Ticket, Workorder or Component, then it is visible to all Users.

12.7 Editing Text

The main menu **Edit** of forms can be used to perform general text editing functions like **Cut**, **Copy** or **Paste** for the content of fields within the form, based on the situation.

In addition, for Tickets and Workorders the entry **Copy Link** can be used to place an URL into the clipboard that points to the current Ticket or Workorder.

12.8 Emails

Emails received by the Service Workbench can be automatically assigned to Tickets or Workorders (see *Section 5.9* and *Section 6.7*).

Emails can also be directly send from the forms of various dataset types (see *Section 12.8.1*).

12.8.1 Sending Emails

The forms for various dataset types contain fields that hold an email address.

Emails can be send to these addresses from within the respective form. The email sending form is either opened by pressing a **Send** button located beside the according field (e.g. the field **Mail** within a Customer form), or by using a matching entry within the form‘s main menu **Extras** (e.g. the entry **Send mail to: Customer** of a Customer form). In some cases more than one email recipient can be selected within the **Extras** menu.

When the email sending form opens, the recipient is already entered based on the content of the respective field, and the current User is entered as the **Sender**. The **Receiver** and the **CC** can be manually modified or selected from the list of known Customers and Companies by pressing the respective button and performing a substring search.

The **Subject** of the email can be entered into the respective field and the content of the email can be entered on the page **Edit** within the **Message** panel. Both, the *Subject* and the *Message*, may include variables that are substituted by the content of selected fields (see *Section 12.8.2*). The page **Preview** within the same panel can be used to see the actual email with the substituted variables.

Templates can be defined and used (see *Section 12.8.3*) to avoid the manual creation of similar emails.

File attachments can be added or deleted by using the **+** or **X** button within the **Attachments** panel.

Pressing the **Send** button will deliver the email to the selected *Recipients* and *CCs* using the configured mail server. Currently undelivered emails can be displayed as a list by selecting the entry **Mail Monitor** from the main menu **Administration**.

12.8.2 Email Variables

The content of the various fields of a form can be included into the subject or message of an email by using variables.

Variables can be added by using the context menu of the **Subject** or the context menu of **Edit** page within the **Message** panel.

Within the context menu, the menu **Insert Variable** provides entries for the fields of the current form. The menu **User** provides variables representing data of the current User.

An exception are Uplink Viewer fields (see *Section 12.3*). These fields only contain a reference to data located within another dataset. This reference is a unique key that is of no use to the recipient of the email.

Individual data of a referenced dataset can be displayed by using the menu item **Insert Relation Template**.

This entry places the string `%LocalID:TargetType:TargetField%` into the email, which is meant as a reminder for the format that is needed to represent the data from the referenced dataset. The string consists of three components that have to be manually replaced by the actual values.

`LocalID` stands for the variable of the Uplink Viewer itself.

`TargetType` is the dataset type that is referenced by the link.

`TargetField` is the variable for the field within the referenced dataset that should be displayed.

For example:
The string
`Dear %customerId:Customer:firstname% %customerId:Customer:lastname%`
would generate a string consisting of the word `Dear` and the `firstname` and `lastname` of the `Customer` that is referenced by the Uplink Viewer `customerId`.

Variables that could be replaced with the current data are shaded in light blue. Variables that cannot be replaced (because the variable is wrong or the respective field has no content) are shaded in red.

12.8.3 Email Templates

Templates for often needed emails can be generated and integrated into new emails.

Creating Email Templates:

Email templates can be created or deleted by using the entry **Mail-Templates** from the main menu **Configuration**.

This entry opens a window in which the templates created by the current User are listed on the page **My Templates**. The buttons located on the right can be used to **Delete** or **Edit** the currently selected template.

The button **Add** can be used to create a new template. This opens a window in which the dataset type, for which the template will be provided, can be selected. The **Subject** and **Message** of the template are created like an individual email. This also includes the usage of variables. Only an identifying **Name** for the template has to be provided.

Administrators can use the page **System Templates** to edit, delete or create templates that are available for all Users.

Using Email Templates:

Templates can be used by pressing the button **Mail-Templates** on the **Edit** page of an email sending form.

The button opens a window in which one of the email templates that have been defined for the current dataset type can be selected. Pressing **Ok** will copy the content of the template into the email sending form. It can be selected whether the template text should replace the current email text, or if it should be appended to the current email text.

12.9 Phone Calls

If the requirements are met and the Call Support is activated (see *Section 12.9.3*), the Service Workbench supports automatic calls started directly from forms (see *Section 12.9.1*) and the automatic display of information for incoming calls (see *Section 12.9.2*).

Important Note:
Currently this feature is only supported for Service Workbench Clients that are running on Windows systems. Linux systems are not supported.

12.9.1 Outgoing Calls

A number of forms (e.g. the Customer form) contain fields which hold a phone number.

When the Call Support is activated and configured for the local Service Workbench Client, the buttons located on the right side of these fields are active and can be used to automatically start a call to the phone number contained within the respective field.

12.9.2 Incoming Calls

When the Call Support is activated and configured for the local Service Workbench Client, the phone numbers of incoming phone calls will be checked by the Service Workbench.

When the target phone number of an incoming call matches the phone number of the User that is currently logged in at the local Service Workbench Client, the Client opens a window that contains information about the incoming call.

When the calling phone number is assigned to a Customer, information known about the Customer is displayed in this window. Amongst others, this information includes the Tickets, Workorders and Components that are assigned to the Customer. This enables a service technician that receives a call from a Customer to directly access the data relevant for this Customer.

In addition the information window provides buttons to generate Tickets and Workorders directly for this Customer which fastens the process to register new problems within the Service Workbench.

12.9.3 Requirements for Call Support

The direct phone support for the Service Workbench can only be used when the following requirements on the Server and on the individual Clients are met:

Server:

The CTI (Computer Telephony Integration) support has to be configured by an *Administrator* on the Service Workbench Server (see separate *Service Workbench Administrator Documentation*).

Clients:

A TAPI Service has to be configured and running on each local system from which a *Service Workbench* Client should use the Call Support. The TAPI Service is configured within the Windows interface and is needed to connect the *Service Workbench* CTI-Client with the telephone system.

The Service Workbench CTI-Client is responsible for the connection of the Service Workbench Client to the local TAPI Service. It also needs to be installed and configured on each system from which the Call Support should be used.

12.10 Tray Bar Icon

When a Service Workbench Client is started on a Windows system, a Tray Bar Icon will be displayed for each Client that is running on the local system.

The Tray Bar Icon provides basic functions (see *Section 12.10.1*) and is used to alert the current User (see *Section 12.10.2*).

12.10.1 General Tray Bar Menu Entries

The Tray Bar Icon that belongs to a Service Workbench Client provides the following context menu entries:

- The menu **Windows** contains one entry for each window currently opened by the Client. The entries can be used to put the respective window into the foreground.
- The entry **Create Desktop Shortcut** creates a shortcut on the desktop. This shortcut can be used to start a Service WorkbenchClient.

- The entry **Info** corresponds to the main menu entry of the same name (see *Section 12.11.1*). It displays a version info for the running Client.
- The entry **Exit** closes the respective Client.

12.10.2 User Notification

The Tray Bar Icon is used to notify the User that is currently running a Service Workbench Client about the fact that new information is available that might concern him.

A notification will be made whenever a Ticket or Workorder is newly assinged to the User. It will also be made when a new Comment is added to a Ticket or Workorder that is assigned to the User.

In these cases the Tray Bar Icon shows a flashing warning symbol and a message will be displayed. In addition an entry for the respective Ticket or Workorder will be provided by the context menu of the symbol. If more than 12 Tickets and Workorders are affected, only the last 12 are represented by menu entries.

The entries can be used to open the respective Ticket or Workorder form. When the form is opened, the according entry is removed from the context menu.

The entry **Acknowledge** within the context menu of the icon can be used to remove the warning symbol.

12.11 Help Menu

The main menu **Help** contains functions with information about the currently used version of the Service Workbench.

In addition the entries **Reset Window Sizes** and **Reset Table Headers** can be used to reset the respective settings to the default setting.

12.11.1 Info

The entry **Info** within the main menu **Help** displays information about the currently used version of the Service Workbench.

The Info window also contains information about active data filters (see *Section 12.6*).

12.11.2 License

The entry **License** within the main menu **Help** displays the currently active license for the Service Workbench.

12.11.3 Logging

The entry **Logging** within the main menu **Help** displays the current level of logging, the path to the client‘s logging and configuration file and the update interval.

By using the button **View** the current logfile of the client will be displayed. The log level defines the grade of detail of the logging output.

It has to be kept in mind, that the levels *DEBUG* and *TRACE* provide a high level of details for the logging output. This may result in a high amount of data within the log file.

12.11.4 Online-Help

The entries of the menu **Online-Help** within the main menu **Help** display the online help for the **User Guide** or the **Administrator User Guide**.

The latter can only be displayed by Users with *Administrator* rights.

13 Administration

The main menu **Administration** contains entries for various administrative tasks like the management of Users, the definition of Working Hours and SLAs or the configuration of the Fault Management connection.

Important Note:
This menu is only visible for Users with *Administrator* access. More about the administrative tasks can be found in the separate *Service Workbench Administrator Documentation*.

This chapter only describes the usage of the related datasets or functions.

13.1 Categorization

Categories are used to categorize Tickets and Solutions to relate them to specific subjects. SubCategories can be assigned to Categories to provide a more detailed classification. The Category Tree can be modified by Users with *Administrator* rights (see *Administrator User Guide*).

If a new Ticket is created, the assignment of a Category may ease the search for a Solution. Since the known Solutions are also assigned to the same Categories, matching Solutions can be found more efficiently.

Categories also help to assign a Ticket to a User Group, because for each Category a Default Group can be defined. If no Group or User is assigned to a new Ticket, the pre-defined Default Group for the selected Category will be automatically assigned, if available.

Within the Service Portal (see *Chapter 14*) Customers have the option to select a Main Category (First level Category within the Category Tree) for their reported problems. This allows an automatic initial assignment of a responsible Group.

13.2 Users

Within the Service Workbench **Users** are the accounts that may log into the Service Workbench.

These accounts are assigned to the individual service technicians. But there are also accounts that are used by external software to log in. For example, there may be a User that is used by the OpenScape FM to automatically create objects within the Service Workbench.

Different access rights can be assigned to the individual Users to restrict the access to the Service Workbench on demand. An *Administrator* can restrict the rights for individual data types to read-only.

Users can be aggregated to Groups (see *Section 13.3*).

13.3 Groups

Within the Service Workbench **Groups** are a cumulation of Users (see *Section 13.2*).

Groups can be configured by Administrators (see *Administrator Documentation*).

Groups are used when a number of service technicians form a team with a common task. For example, all Users that should handle problems for a specific region can be assigned to a specific Group. Tickets for this region can then be assigned to this Group instead of to individual Users. Every member of the Group will see the problem and can react on it.

13.4 Locations

Besides the information about a Location itself, within the Service Workbench **Locations** are a part of the access management.

By using a selection menu, Locations can be assigned to individual datasets of the types User, Company, Customer or Component.

An *Administrator* can define for every Location, which Users or Groups may see datasets that are assigned to the respective Location.

13.5 Holidays

Holidays are reoccurring or one-time dates defined by an *Administrator* that are omitted by the Service Workbench when e.g. the escalation times of Tickets are calculated.

13.6 Working Hours

Working Hours define the hours within a normal day (not a Holiday - see *Section 13.5*), during which problems should be handled by service technicians.

The escalation of Tickets is in effect only during Working Hours.

13.7 Status Transitions

Status Transitions are the Status changes of a Ticket. For example from *New* to *Working*.

Undesired Status Transitions (e.g. from *Closed* to *New*) can be prohibited by an *Administrator*.

13.8 SLA (Service Level Agreement)

To each Ticket an **SLA** with one or more service agreements can be assigned. Depending on the set Priority, these agreements define reminders for the service technicians to ensure that a specific Status is reached in time.

For example the reaction time for a Ticket of high Priority might be one hour and it has to be solved within three hours

Only the defined Working Hours will be used for the calculation (see *Section 13.6*).

13.9 Itemized Lists

Itemized Lists contain entries defined by an *Administrator* that are displayed as selection lists in numerous forms.

13.10 Component Catalog Types

Component Catalog Types provide the means to assign individual Component Catalogs (see *Section 8.1*) to certain base functions (e.g. printers, monitors). They are a special Itemized List that can also be modified by an *Administrator*.

13.11 Services (Portal)

Within the area **Services** the Service Workbench Service Portal displays predefined tiles that provide various functions. For example, the tile **Report Problem** provides the means for a Customer to generate a new Ticket by using the Service Portal.

Service Templates represent the particular functions. Using a wizard, an *Administrator* can select, modify or create them. They can be individually activated for the Customers.

13.12 Cost Centers

A **Cost Center** can be assigned to various datasets. Cost Centers can be defined by an *Administrator* and e.g. include the account number that represents the specific Cost Center.

13.13 Fault Management Connector

If the Service Workbench has been licensed for Fault Management, an *Administrator* can configure the interface to the Fault Management. The configuration window can be opened by using the menu entry **Fault Management Connector**.

13.14 Workflow

The menu entries **Workflow Timer**, **Workflow Start Node**, and **Workflow Configuration** are only visible if the Workflow Engine has been activated (see *Chapter 10*).

Workflow Timer

This menu entry opens a window in which all timers that are used by the Workflow Engine are displayed. The list provides an overview about the Workflow Engine and points at timers that have not been finished in the regular time.

Workflow Start Node

This menu entry opens a window in which the Start Nodes for the various actions can be defined.

Workflow Configuration

This menu entry opens the configured Workflow View within the client of the Workflow Engine.

13.15 Mail Monitor

The menu entry **Mail Monitor** opens a window which displays all emails that have **not** been delivered by the system.

13.16 Changelog

The menu entry **Changelog** opens a window which can be used to search for changes that have been performed by the system. These changes concern the creation, modification and deletion of datasets.

13.17 Server-Log

The menu entry **Show Server-Log** can be used to display the log file of the Service Workbench.

13.18 Server Configuration

The menu Server Configuration contains entries that display various configuration files of the Service Workbench.

14 Service Portal

The Service Workbench includes a web interface named **Service Portal**.

The Service Portal is a tool for Users to report problems to the respective service technicians in an easy and uncomplicated manner. The delivered data will be directly stored as Tickets in the Service Workbench. The creation of Tickets is therefore independent of business hours of the service technicians.

This chapter explains the functions of the web-based user interface. It is described how Tickets can be created, how the Status of their execution can be monitored, and how a Solution can be evaluated when a Ticket is considered *Solved*.

14.1 Launch and Login

The Service Workbench Service Portal is a purely web-based application and requires only a web browser to be used.

Entering the URL for the web service starts a connection to the Service Portal. This URL is generally composed as follows:
`https://<servername>:<port>/ServicePortal`

Where `<servername>` is the name or IP address of the server on which the Service Workbench has been installed, and `<port>` is the connection port configured for the Service Portal (default is `8080`).

The URL opens a login page where the Customer ID (field **Login**) and the **Password** have to be entered.

Hint:
The Password *is* case-sensitive, the Login *is not* case-sensitive.

Alternatively, the Service Workbench Customer login details can be authenticated by the Windows domain controller. If this is the case, the login uses the Windows domain account. If several domains are configured, the appropriate domain can be prefixed to the login name, separated by a backslash. Example: `domain\user` .

Forgotten Password: (optional feature)

If internal authentication is activated and a Customer has forgotten his password, a password reset can be requested by clicking the link **Forgot Password?**. The request will deliver an email with new login credentials to the email address that was saved for the Customer with the entered login.

When the email is received, a login with the new password can be performed. This new password should be changed at the first opportunity (see *Section 14.5*).

14.2 Central View of the Service Portal

The user interface of the Service Portal consists of two major pages that can be selected from the **Selection Bar** located at the top of the window.

- The page **Overview** (see *Section 14.3*) displays a list of the Tickets and Workorders that are released for the current User (see *Section 5.8* and *Section 6.6*).
- The page **Services** (see *Section 14.4*) offers a number of functions that can be started by selecting the respective tile.

The right side of the Selection Bar displays three buttons:

- The **Host Name** of the machine from which the login was performed. If the host is represented as a Component within the Service Workbench, the button will display the Component‘s data.
- The **User Name** of the currently connected User. The button displays the data of the User dataset for the current User.
- The button **Logout** can be used to disconnect the current User from the Service Portal.

14.3 Overview

The page **Overview** is used to display lists of Tickets and Workorders that are of relevance for the current Customer.

The menus **Ticket** and **Workorder** contain entries for the possible Status values for Tickets and Workorders. The entry **All** stands for all Status values, and the entry **Open** for all Status values except *Closed*.

If such an entry is selected, a list of those Tickets or Workorders is displayed that are released for the current Customer (see *Section 5.8* and *Section 6.6*), and that have the selected Status.

The displayed list can be restricted further by using the Search Field located below the menus.

Only Tickets or Workorders are displayed that contain the entered string as a substring of their *Number*, *Status* or *Short Description*. For example, it can be directly searched for a Ticket with a specific Ticket number or for a Workorder for the topic ‚Printer‘.

If the button **Refresh** is pressed, the content of the table will be updated.

The menu **Select Table Columns** can be used to hide or show individual columns. This has no effect on the display restrictions triggered by the Search Field entries.

The tooltips of the individual list entries contain additional information about the Tickets or Workorders.

Clicking a list entry within the Overview opens a window that contains additional **Information** like the complete description and the attachments. Within this window, Comments can be generated (see *Section 14.3.1*) or Solutions can be evaluated (see *Section 14.3.2*).

14.3.1 Commenting Tickets or Workorders

The window, described in the previous section, for the representation of the detail information for Tickets and Workorders can also be used to communicate with the respective service technician.

If a Comment for the Ticket or Workorder has been created by the service technician, the window contains the page **Comments**. This page lists the individual Comments, shows the contact data for the respective creator, and by using the button **Reply** allows the creation of a direct response to the respective Comment.

A new Comment can be generated by using the button **Add Comment**.

If a new Comment is created or an old Comment is replied to, a message for the respective service technician will be generated.

14.3.2 Displaying and Evaluating Solutions

For Tickets, the window for the display of detailed information described in *Section 14.3* can also be used to view Solutions that are assigned to the Ticket and to evaluate them if applicable.

The Solution will be displayed on the page **Solution**:

- if the Ticket has the Status *Solved* or *Closed*.
- if, independent of the Status of the Ticket, a Solution has been assigned to the Ticket (see *Section 5.7*) and the Solution has been released for the Ticket (see *Section 5.8*).

If, in addition, the Ticket has the Status *Solved*, the Customer can use the button **Decline** to reject the Solution and start a reevaluation, or the button **Accept** to allow the closure of the Ticket.

14.4 Services

The page **Services** contains tiles that provide access to various functions.

The default configuration of the Service Workbench provides four tiles that can be individually activated by an Administrator.

Additional tiles with new functions can also be defined by an Administrator (see *Administrator User Guide*).

The following four sections explain the functions provided by the pre-configured tiles.

14.4.1 Report Problem

The Tile **Report Problem** on the page **Services** can be used to create a new Ticket.

Clicking the tile opens a window in which the necessary data can be entered:

- **Urgency**: a selection menu to define the priority of the Ticket from the Customer‘s point of view.
- **Category**: a selection menu to preselect a Category. The entries are the top level Categories (see *Section 13.1*). A selection will send the problem to the service technician Group that is assigned to the Category.

- **Affected Component**: a selection menu that lists all Components for which the current Customer is either the *Contact Person* or the *Customer*. Here a Component related to the problem can be selected.
- **Working Group**: a selection menu to directly assign a service technician Group to the problem. This overrides the assignment by a selected Category (see above). This menu has to be activated by an *Administrator*.
- **Short Description**: Keywords or short description to define the problem. Restricted to 250 characters.
- **Ticket Description**: Detailed description of the problem. By default, restricted to 4000 characters.
- **Attachments**: Here *Files* or *URLs* with additional information (e.g. a log file containing error messages) can be added to the problem description. By default the maximum size for the files is 5 MB.

If the button **Save** is pressed, the data will be stored into a new Ticket within the Service Workbench and an overview of the Ticket will be displayed. The overview can be used to add a Comment to the Ticket (see *Section 14.3.1*).

The Ticket Number should be noted and used for future references to this Ticket (e.g. in phone calls).

14.4.2 Find Solution

Before a Ticket is created, it should be checked whether an appropriate Solution is already available in the Solution database. In some cases, a Solution might be available that allows the Customer to solve a problem by himself.

The Solution finder can be started by clicking the Tile **Find Solution** on the page **Services** which will open the search window for Solutions.

The search window lists all Solutions that are generally released for the Service Portal (see *Section 7.3*) and have the Status *Reviewed*.

The menu **Category** can be used to restrict the list to Solutions stored for the selected Category.

The field below can be used to enter the search text. The displayed list will be restricted to Solutions that contain the text in their *Number*, *Ticket Description*, *Ticket Short Description*, *Solution Description* or *Solution Short Description*.

The selection in the upper left can be used to define the search modus. It can either look for exact terms (**Search for exact term**) or for Solutions that contain at least one word entered in the field (**Search for single words**).

The list entries can be clicked to show a detailed view of the respective Solution.

Within this detailed view, the button **Helped** (located in the lower right) should be clicked if a Solution was helpful. This helps the service team to identify the more helpful Solutions.

In the detailed view, the button **Add Comment** (located in the lower left) can be used to create a new Comment for the Solution. This is handled like the creation of Comments for Tickets or Workorders (see *Section 14.3.1*).

14.4.3 Find Person

The Tile **Find Person** on the page **Services** can be used to find Customers based on a search string.

The search string can be entered into a field and is checked against the following data of the Customer datasets:

Firstname, *Lastname*, *Login*, *Mail*, *Phone*, *Mobile*, *Fax*, *Pager*, *City*, *Country*, *Department*, *Street*, *Zip*, *Room*, *Location*, *Costcenter* and *Company*.

Matching Customers will be displayed below the search field.

The selection in the upper left can be used to define the search modus. It can either look for exact terms (**Search for exact term**) or for datasets that contain at least one word entered in the field (**Search for single words**).

14.4.4 Show Report

By using the tile **Show Report** on the page **Services** the Reports can be generated that have been defined and released by an *Administrator* for the current Customer (see *Administrator User Guide*).

The tile opens a window with a list of the released Reports.

After the selection of an output format (*PDF* or *MS-Excel*), the selection of the Reports and the input of the Report parameters, the output can be started.

14.5 Changing Password

If password changes are allowed by an *Administrator* (e.g. if automatic authentication is not used), the Selection Bar contains the entry **Properties**.

This entry can be used to open the password change window.

The old password has to be entered, and the new password has to be entered twice.

The new password can only be stored, when the two entries are identical.

Note:
If the Service Workbench Service Portal uses a Windows domain controller for authentication, by default this function is disabled.

A Status Transitions

Status transitions describe the changes permitted for the Status of a Ticket. Users with Administrator rights can amend the permissible status transitions at any time.

The following Status transitions are selected by default:

Status Transition	Descriptions
New - New	Newly created Ticket remains in the status *New* to allow additions to be made.
New - Forwarded	When the Ticket is passed on to a specific Group and/or User; the Ticket is no longer shown in the *Not Assigned Tickets* section.
New - Working	Someone has started processing the Ticket. The Ticket is no longer shown in the *Not Assigned Tickets* section. This signifies that the Ticket is being processed; the Customer will be sent an email if the option was selected.
New - Solved	The Ticket was solved straight away; the Customer will be sent an email if the option was selected.
Forwarded - Forwarded	The Ticket was reassigned to another Group or User.
Forwarded - Working	Someone has started processing the Ticket; the Customer will be sent an email if the option was selected.
Working - Forwarded	A User has forwarded the Ticket to another User or Group.
Working - Working	To be able to record the worksteps performed, the User has saved the Ticket without changing the Status.
Working - Deferred	The Ticket will not be processed for the time being; Escalation is suspended.
Working - Solved	The User has solved the Ticket; the Customer will be sent an email if the option was selected.

Deferred - Working	Processing of the Ticket has resumed; the period of deferment is added to the Escalation Date.
Deferred - Deferred	Additional information can be added.
Solved - Forwarded	The Customer does not agree with the Solution; the Ticket is returned to the Group/User for re-processing.
Solved - Working	The Customer does not agree with the Solution; the Ticket is re-processed immediately.
Solved - Closed	The Customer accepts the Solution to the problem; no further changes to this Ticket are permitted. In some configurations, a Ticket can be changed automatically to the status *Closed* after a predefined period of time.

B Keyboard Controls

The keyboard controls are divided into two groups, one for the main window and the other for the data forms.

Keyboard Shortcuts in the Main Window:

Action	Keyboard shortcut
File	ALT + D
- Print Preview	CTRL + P
- New Login	ALT + F5
- Close	ALT + F4
Edit	ALT + B
- Cut	CTRL + X
- Copy	CTRL + C
- Paste	CTRL + V
Dataset	ALT + T
- Open	CTRL + O
- Duplicate	CTRL + D
- Delete	CTRL + L
Administration	ALT + A
Synchronisation	ALT + S
Configuration	ALT + G
Reports	ALT + R
Window	ALT + W
Help	ALT + H
New	ALT + N

Selection in Tree	Keyboard Shortcut
- Overview Page	ALT GR + F5
- Ticket (My Tickets)	ALT + F6
- Customer	ALT + F7
- Company	ALT + F8
- Catalogue	ALT + F9
- Component	ALT + F10
- Room	ALT + Shift + F10
- Solution	ALT + F11
- Work Order (Mine)	ALT + F12
- Detected Software	ALT + P
- Contract	ALT + C
- Licence	ALT + L
- Licence Type	ALT + Y
Open / Close Search	ALT + F
Start New Search	CTRL + F

Keyboard shortcuts in the forms:

Action	Keyboard shortcut
Refresh	F5
Close	ESC
File	ALT + D
- Save	CTRL + S
- Save and New	CTRL + N
- Save and Close	CTRL + I
- Duplicate	CTRL + D

Edit	ALT + B
- Cut	CTRL + X
- Copy	CTRL + C
- Paste	CTRL + V
Extras	ALT + E
Actions	ALT + A
Extras (form: Tickets)	
- E-mail to Customer	ALT + Shift + F3
- Ticket Printout	ALT + P
- Use Template	ALT + Shift + V
- Save Template	ALT + Shift + S
Extras (form: Workorders)	
- E-mail to Customer of Ticket	ALT + Shift + F3
- Work or Printout	ALT + P
- Use Template	ALT + Shift + V
- Save Template	ALT + Shift + S
Extras (form: Customers)	
- Send e-mail to Customer	ALT + M
Extras (form: Components)	
- Use Template	ALT + Shift + V
- Save Template	ALT + Shift + S
Extras (form: Detected Software)	
- Change Update Flag	ALT + H
- Create New Licence Type	ALT + Y

New in the *File* Menu	
- Ticket	ALT + F6
- Customer	ALT + F7
- Company	ALT + F8
- Catalogue	ALT + F9
- Component	ALT + F10
- Room	ALT + Shift + F10
- Solution	ALT + F11
- Work Order	ALT + F12
- Contract	ALT + C
- Licence	ALT + L
- Licence Type	ALT + Y
- Software (Components only)	ALT + S
- Hardware (Components only)	ALT - H
Bring Main Window to Front	CTRL + M

Index