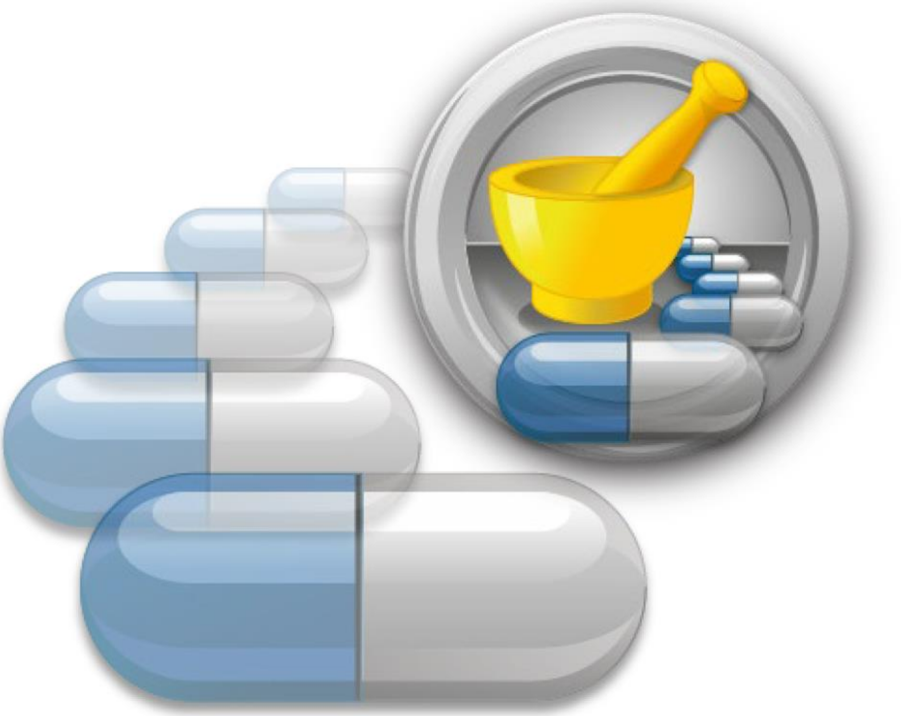
FT PharmaSuite®



DATA MANAGER

RELEASE 10.02.00

USER MANUAL

PUBLICATION PSDM-UM006C-EN-E-SEPTEMBER-2021
Supersedes publication PSDM-UM006B-EN-E

Rockwell Automation

Contact Rockwell See contact information provided in your maintenance contract.

Copyright Notice

Trademark Notices

Other Trademarks

Warranty This product is warranted in accordance with the product license. The product's performance may be affected by system configuration, the application being performed, operator control, maintenance, and other related factors. Rockwell Automation is not responsible for these intervening factors. The instructions in this document do not cover all the details or variations in the equipment, procedure, or process described, nor do they provide directions for meeting every possible contingency during installation, operation, or maintenance. This product's implementation may vary among users.

This document is current as of the time of release of the product; however, the accompanying software may have changed since the release. Rockwell Automation, Inc. reserves the right to change any information contained in this document or the software at any time without prior notice. It is your responsibility to obtain the most current information available from Rockwell when installing or using this product.

Contents

Chapter 1 Data Manager 1

Typographical Conventions 2

Screen Layout 3

Panel Management 4

Dialogs 6

Mismatch Details 6

Help Access 7

About PharmaSuite 8

Editors 10

BigDecimal Editor 11

BigDecimal Editor for Double Data Types in Automation 13

BigDecimal Editor for Float Data Types in Automation 17

BigDecimal Editor for Integer Data Types in Automation 20

BigDecimal Editor for Historian 24

Boolean Editor for Automation 25

CleaningRules Editor 26

Date/Time Picker Editor 27

Duration Editor 29

Expression Editor 29

Expression Editor for Transition Conditions 30

Expression Editor for Transition Actions 31

Expression Panel 32

Operators 42

Functions 48

FlexibleAttributeDefinition Editor 53

FlexibleStateModel Editor 55
FlexibleTagDefinition Editor for Automation 56
GraphStatusChange Editor 58
Label Layout Selection Editor 58
Measured Value Editor 59
Multi-line Text Editor 59
Ranges Editor 60
ScaleConfiguration Editor 61
ScaleTestAndCalibration Editor 65
Searchable Option List Editor 67
String Editor for Automation 67
String Editor for Historian 69
Tag Browser Editor for Automation 70
Graphical Elements 71
Toolbar Buttons 71
Action Buttons 73
Marker Icons 76
Cursors 77
Expression Editor Icons 78
Keyboard Operation 79
Framework Navigation Shortcuts 79
Action Shortcuts 80
Expression Editor Shortcuts 82
Basic Operations 87
Start, Login, Logout, and Password Change 87
Searching 92
Editing 100
Signature Requests 101
Exporting and Importing Data 103

Chapter 2 Managing Equipment 121

What Is an Equipment Requirement? 121

What Is an Equipment Class? 121

What Is a Template Equipment Entity? 122

What Is an Equipment Entity? 123

What Is an Equipment Graph? 126

What Is a Property Type? 126

Status Handling of Equipment 126

Equipment Class Statuses 127

Template Equipment Entity Statuses 128

Equipment Entity Statuses 129

Equipment Graph Statuses 130

Changing the Status of Equipment 130

Mass Change Operations on (Template) Entities 133

Graph Transition Triggering on Entities 139

Usage Lists of Classes 141

Managing Equipment - Smart Search 143

Menus and Toolbars 143

Detail Data 156

Managing Equipment - Basic Search 207

Menus and Toolbars 207

Detail Data 221

Chapter 3 Managing Work Centers 271

What Is a Work Center? 271

What Is a Station? 271

Menus and Toolbars 271

Menus 271

Toolbars 276

Detail Data 277

Work Center 277

Station 279

Index 281

Figures

Figure 1: Screen layout of Data Manager - Equipment - Smart Search 3

Figure 2: Screen layout of Data Manager - Equipment - Basic Search 3

Figure 3: Screen layout of Data Manager - Work Center.. 4

Figure 4: Mismatch warning dialog 6

Figure 5: Help window navigation 8

Figure 6: About PharmaSuite 9

Figure 7: PharmaSuite Installation Details.. 9

Figure 8: BigDecimal editor for property type 11

Figure 9: BigDecimal editor for property of equipment class............................. 12

Figure 10: BigDecimal editor for property of equipment entity 12

Figure 11: Automation BigDecimal (Double) editor for property type.................... 14

Figure 12: Automation BigDecimal (Double) editor for property of equipment class 14

Figure 13: Automation BigDecimal (Double) editor for property of equipment class with verification 14

Figure 14: Automation BigDecimal (Double) editor for property of equipment entity 15

Figure 15: Automation BigDecimal (Double) editor for property of equipment entity with verification.. 15

Figure 16: Automation BigDecimal (Float) editor for property type 17

Figure 17: Automation BigDecimal (Float) editor for property of equipment class...... 17

Figure 18: Automation BigDecimal (Float) editor for property of equipment class with verification... 18

Figure 19: Automation BigDecimal (Float) editor for property of equipment entity 18

Figure 20: Automation BigDecimal (Float) editor for property of equipment entity with verification... 19

Figure 21: Automation BigDecimal (Integer) editor for property type 20

Figure 22: Automation BigDecimal (Integer) editor for property of equipment class 21

Figure 23: Automation BigDecimal (Integer) editor for property of equipment class with verification 21

Figure 24: Automation BigDecimal (Integer) editor for property of equipment entity 22

Figure 25: Automation BigDecimal (Integer) editor for property of equipment entity with verification 22

Figure 26: Historian BigDecimal editor for property of equipment class 24

Figure 27: Historian BigDecimal editor for property of equipment entity 24

Figure 28: Automation Boolean editor for property of equipment class 25

Figure 29: Automation Boolean editor for property of equipment entity 25

Figure 30: CleaningRules editor for property type 26

Figure 31: CleaningRules editor for property 27

Figure 32: Date/Time Picker editor 28

Figure 33: Date/Time Picker editor 28

Figure 34: Duration editor 29

Figure 35: Expression editor for transition conditions 30

Figure 36: Expression editor for transition actions 31

Figure 37: Expression with comment 33

Figure 38: Expression with sections (all open) 34

Figure 39: Expression with sections (all closed) 34

Figure 40: Expression panel with comment and auto-completion 36

Figure 41: Expression panel with errors and message tooltip 37

Figure 42: Three-valued truth table 46

Figure 43: Truth table - A AND B 47

Figure 44: Truth table - A OR B 47

Figure 45: Truth table - NOT A 48

Figure 46: FlexibleAttributeDefinition editor for property type 53

Figure 47: FlexibleAttributeDefinition editor for property of equipment class 54

Figure 48: FlexibleAttributeDefinition editor for property of equipment entity 54

Figure 49: FlexibleStateModel editor for property type 55

Figure 50: FlexibleStateModel editor for property 55

Figure 51: Automation FlexibleTagDefinition editor for property type 56

Figure 52: Automation FlexibleTagDefinition editor for property of equipment class 56

Figure 53: Automation FlexibleTagDefinition editor for property of equipment entity 57

Figure 54: GraphStatusChange editor for graph of entity 58

Figure 55: Label Layout Selection editor 58

Figure 56: Measured Value editor 59

Figure 57: Multi-line Text editor 59

Figure 58: ScaleRanges editor for property 60

Figure 59: ScaleConfiguration editor for property of class 64

Figure 60: ScaleConfiguration editor for property of entity 64

Figure 61: ScaleTestAndCalibration editor for property 66

Figure 62: Searchable Option List editor 67

Figure 63: Automation String editor for property of equipment class 67

Figure 64: Automation String editor for property of equipment entity 68

Figure 65: Historian String editor for property of equipment class 69

Figure 66: Historian String editor for property of equipment entity 69

Figure 67: Tag Browser editor for automation properties 70

Figure 68: PharmaSuite webstart page 87

Figure 69: Login form 88

Figure 70: PharmaSuite welcome page 89

Figure 71: Change password 91

Figure 72: Search tabs with Close button 92

Figure 73: Card of equipment class 93

Figure 74: Card of work center 93

Figure 75: Tile of equipment class with tooltip 94

Figure 76: Tile of work center with tooltip 94

Figure 77: Mini tile of equipment class with tooltip 95

Figure 78: Mini tile of work center with tooltip 95

Figure 79: Search criteria panel - Equipment - Smart Search 97

Figure 80: Search criteria panel - Work Center 98

Figure 81: Search criteria panel - Equipment - Basic Search 99

Figure 82: Single electronic signature 102

Figure 83: Single electronic signature with pre-defined comment 102

Figure 84: Double electronic signature to support witness role 103

Figure 85: Export dialog for class 106

Figure 86: Export dialog for entity 106

Figure 87: Log dialog of class export 107

Figure 88: Export Automation Tags dialog for class 108

Figure 89: Export Automation Tags dialog for entity 109

Figure 90: Log dialog of automation tag export 109

Figure 91: Import dialog 113

Figure 92: Import Content dialog 114

Figure 93: Import Checks dialog 114

Figure 94: Log dialog of import 115

Figure 95: Import Automation Tags dialog 117

Figure 96: Log dialog of automation tag import 118

Figure 97: Verify Automation Tags dialog 119

Figure 98: Change Status dialog 131

Figure 99: Change Status dialog with warnings 132

Figure 100: Change Status dialog with errors 132

Figure 101: Update (Template) Entities - Expected Changes 134

Figure 102: Update (Template) Entities - Update Results 136

Figure 103: Trigger Graph Transition dialog for single entity 140

Figure 104: Trigger Graph Transition dialog for parent entity of group 140

Figure 105: Usage List dialog 142

Figure 106: Example graph 195

Figure 107: Example graph 258

Chapter 1

Data Manager

Data Manager of PharmaSuite is a graphical workbench for creating and maintaining master data. With its function for change control, it covers the entire life cycle of master data. It is available in two modes. As Data Manager - Equipment, it allows to create and maintain equipment classes, template equipment entities, equipment entities, property types, and equipment graphs and their assignments between one another. As Data Manager - Work Center, it allows to create and maintain work center and station objects and the hierarchical relationship that exists between a work center and its station or stations.

This section contains important information about the basic principles of working with Data Manager. Please read this section carefully, because it provides a solid background for all operations you may wish to perform with your system.

Later sections will explain how to perform the specific tasks in the system. We assume you are familiar with the conventions described in the following sections and the fundamentals of working with a personal computer.

Typographical Conventions

This documentation uses typographical conventions to enhance the readability of the information it presents. The following kinds of formatting indicate specific information:

Bold typeface

Designates user interface texts, such as

- window and dialog titles
- menu functions
- panel, tab, and button names
- box labels
- object properties and their values (e.g., status).

Italic typeface

Designates technical background information, such as

- path, folder, and file names
- methods
- classes.

CAPITALS

Designate keyboard-related information, such as

- key names
- keyboard shortcuts.

`Monospaced typeface`

Designates code examples.

Screen Layout

The basic screen layout of Data Manager holds the following user interface components:

- menus and toolbars for accessing the functions of Data Manager
- Search window with search criteria panel and search results panel
- Details window for viewing and editing the data of objects selected from the search results
- status bar with general information such as the logged-in user with his full name, the database name, and the local time and time zone information.

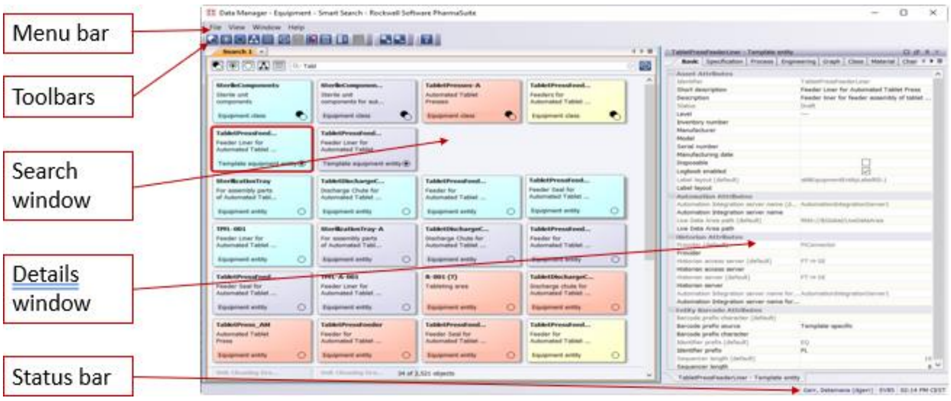

Menu bar
Toolbars
Search window
Details window
Status bar
Data Manager - Equipment - Smart Search - Rockwell Software PharmaSuite

Figure 1: Screen layout of Data Manager - Equipment - Smart Search

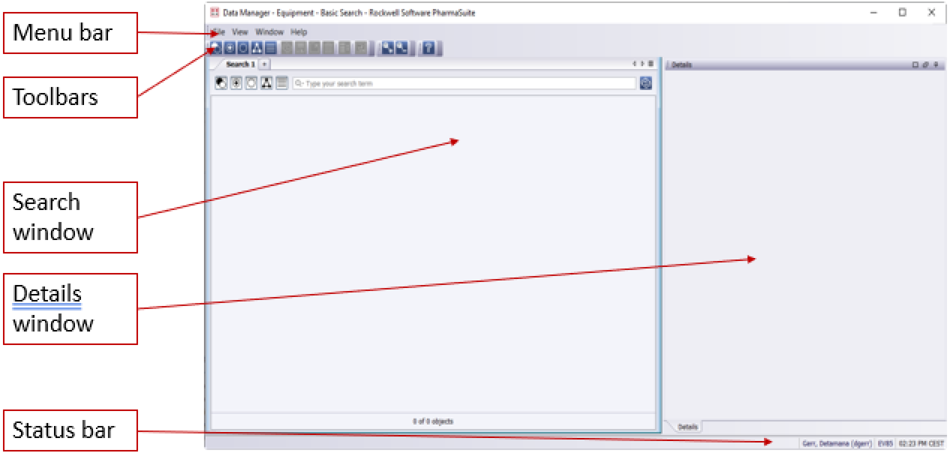


Figure 2: Screen layout of Data Manager - Equipment - Basic Search

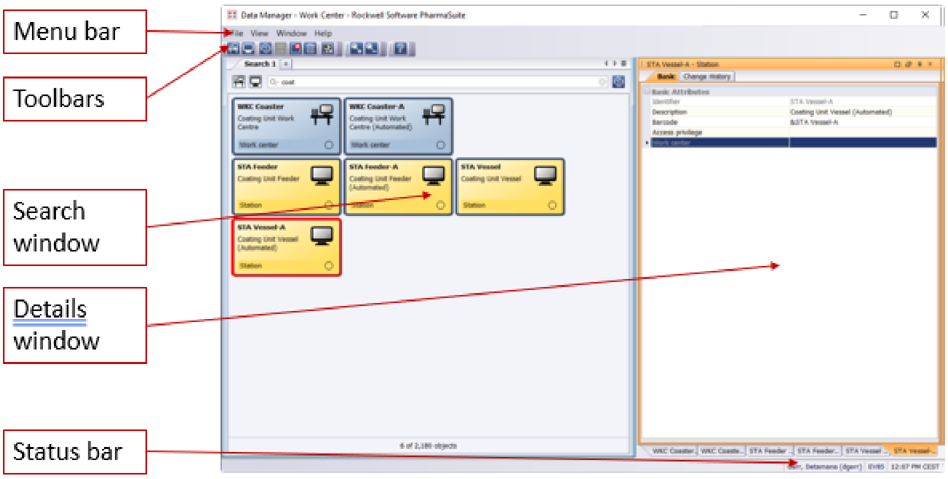


Figure 3: Screen layout of Data Manager - Work Center

Panel Management

When you start Data Manager for the first time, it opens with the default screen layout, which holds the dockable main menu bar and main toolbars at the top, the Search window taking up the left and center part of the screen, and the Details window on the right. You can, however, rearrange the panels if you prefer a different layout.

When you exit the system, Data Manager saves the current panel layout and will start your next session with this layout. You can, however, always revert back to the default layout (page 5).

To change the position and size of the panels, click a panel header bar and drag the panel to the desired position on the screen. The system displays the new panel size and position with a gray preview frame. Panels can be

- free floating
- docked horizontally or vertically
- nested into another panel, accessible for switching by tab.

TIP

To move a menu bar or a toolbar you have to use their drag handles, which are located at the left margins of horizontal bars and at the top margins of vertical bars. If you pull a toolbar out as floating panel, you can close the panel.
To reopen it, proceed as follows:

- Right-click anywhere on the menu bar to open a shortcut menu.
- Select the toolbar. The system displays it again at the location where you closed it before.

With the other functions of the shortcut menu, you can configure if the bars are **Rearrangeable**, **Hidable**, or **Floatable**.

Each panel header provides a toolbar for further panel resizing and movements:

□ ❐ Maximize/restore
Maximizes a panel at its current position or restores it to its original size.

❐ ▣ Toggle floating
Floats or unfloats a panel. An unfloated panel reverts to its original position and size.

⊥ ⊣ Toggle auto-hide
Auto-hides a docked panel. It can be accessed by tab. If you revoke the auto-hide, the panel reverts to its original position and size.

☒ Close
Closes a panel. To reopen it, open the **View** menu and select the panel for display.

When the number of open tabs on a tab bar exceeds the available width of the tab bar, the system displays the following buttons to support your navigation:

◀ Scroll backward
Scrolls the tab bar to the left.

▶ Scroll forward
Scrolls the tab bar to the right.

☰ Close
Displays a list of all open tabs from which you can select a tab you wish to close. The currently active tab is indicated by bold font.

To manage your screen layout changes, the system provides the **Window** menu with the following functions:

- Load user layout
  Loads the last layout you have saved on this computer with the **Save user layout** function.

- Redo layout change
  Redoes the last layout change you have revoked with the **Undo layout change** function. You can redo up to 100 actions, thus you can step by step redo the last 100 layout changes you have revoked. Once you have performed a layout change, the menu function changes to a more precise description of the undo action, such as **Redo resizing** or **Redo dragging**.

- Reset layout
  Resets the window layout to the system-defined default layout. This function does not affect the saved user layout, which can be restored with the **Load user layout** function.

- Save user layout
  Saves the current window layout and overwrites the layout that was last saved by you on this computer.

- Undo layout change
  Revokes the last layout change you have performed. You can undo up to 100 actions, thus you can step by step revoke the last 100 layout changes you have performed. Once you have performed a layout change, the menu function changes to a more precise description of the undo action, such as **Undo resizing** or **Undo dragging**.

Dialogs

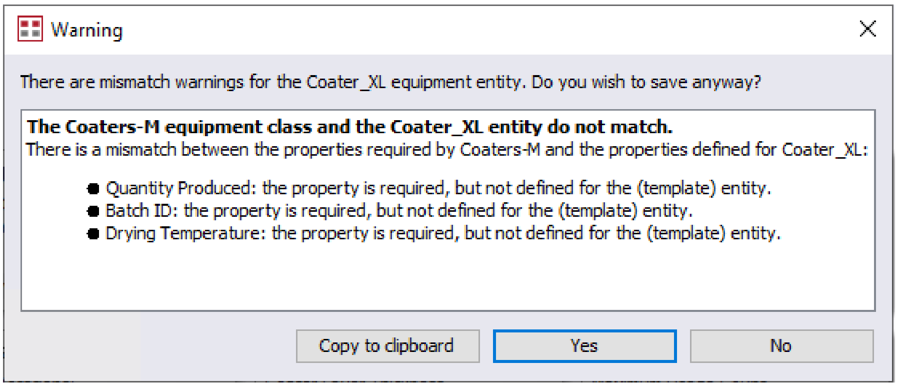
Data Manager uses dialogs or pop-up windows on top of its basic screen areas to cover a variety of functions:

- Error and warning dialog (page 6) for indicating mismatching data
- Context-sensitive Help system (page 7)
- About dialog (page 8) for further information on PharmaSuite and its system environment.

Mismatch Details

To prevent data inconsistencies that may result from unsuitable assignments between objects, Data Manager runs a series of consistency checks whenever you save an object. As long as at least one of the affected objects is in an editable status, such as **Draft**, so that mismatches are still solvable, the system reports mismatch warnings.

An object's unresolved mismatch warnings can prevent the object from changing its status to a non-editable status, such as **Verification**.



There are mismatch warnings for the Coater_XL equipment entity. Do you wish to save anyway?

The Coaters-M equipment class and the Coater_XL entity do not match.
There is a mismatch between the properties required by Coaters-M and the properties defined for Coater_XL:

- Quantity Produced: the property is required, but not defined for the (template) entity.
- Batch ID: the property is required, but not defined for the (template) entity.
- Drying Temperature: the property is required, but not defined for the (template) entity.

Figure 4: Mismatch warning dialog

TIP

If the consistency checks return several issues, click the **Copy to clipboard** button to copy the list and then paste it into a system-external text editor. This way you can keep the list open to address the individual issues one after another.
To quickly locate an affected object, copy its identifier and paste it into the **Search** box.

Help Access

The help system of Data Manager is context-sensitive on panel and window level. This means that clicking the help button or pressing the F1 key will open a web browser to display a help window with information relevant to the panel you have currently focused. The help window is non-modal and resizable.

In order to access other topics than the one directly related to the current context, use the navigation arrows located at the top of the help page, or related topic links located at the bottom of the page, if available.

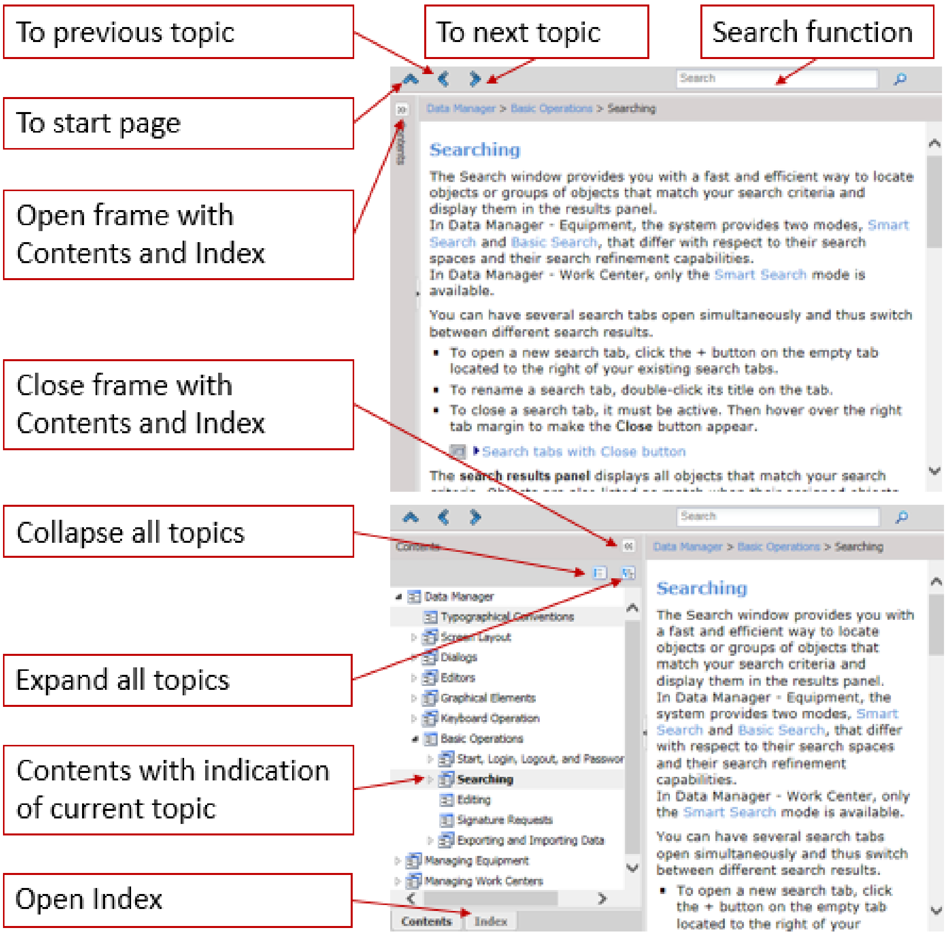
The following additional features support your use of the help system:

- To access an overview of all available topics, open the **Contents** tab in the **Contents and Index** frame. The system additionally provides **Expand all** and **Collapse all** buttons to facilitate easier navigation in the contents tree.
- To access the index, open the **Index** tab in the **Contents and Index** frame.
- To use the **Search** function

  1. type the term you are looking for in the **Search** box and
  2. click the **Search** button or press the ENTER key.
     The system will display all occurrences of the search term in a third tab in the **Content and Index** frame.

- To print the page that is currently displayed in the help window, click the **Print this page** button. The system displays a print preview of the page along with the default Windows **Print** dialog.

  TIP
  Please note that printing is only available from the stand-alone format of the help system and not from within the application.



Searching

The Search window provides you with a fast and efficient way to locate objects or groups of objects that match your search criteria and display them in the results panel.
In Data Manager - Equipment, the system provides two modes, Smart Search and Basic Search, that differ with respect to their search spaces and their search refinement capabilities.
In Data Manager - Work Center, only the Smart Search mode is available.

You can have several search tabs open simultaneously and thus switch between different search results.

- To open a new search tab, click the + button on the empty tab located to the right of your existing search tabs.
- To rename a search tab, double-click its title on the tab.
- To close a search tab, it must be active. Then hover over the right tab margin to make the **Close** button appear.

Search tabs with Close button

The **search results panel** displays all objects that match your search

Figure 5: Help window navigation

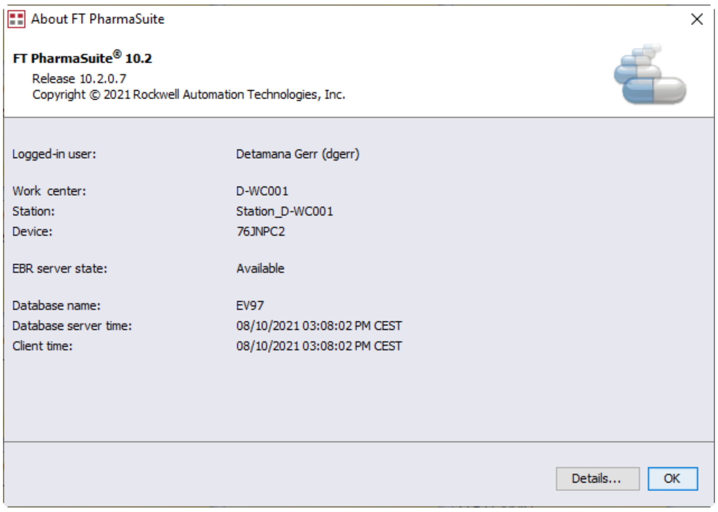
About PharmaSuite

The **About PharmaSuite** function opens the **About PharmaSuite** dialog to display system-related information, such as the current system version and build, the logged-in user, work center, and database-related information.

TIP

Please note that the dialog also indicates the **EBR server state**. Only when the EBR server is available can orders or workflows be processed in PharmaSuite for Production Execution.

Click the **Details** button to view more specific technical information on the system and its environment.



About FT PharmaSuite

FT PharmaSuite® 10.2
Release 10.2.0.7
Copyright © 2021 Rockwell Automation Technologies, Inc.

Logged-in user: Detamana Gerr (dgerr)

Work center: D-WC001
Station: Station_D-WC001
Device: 76JNPC2

EBR server state: Available

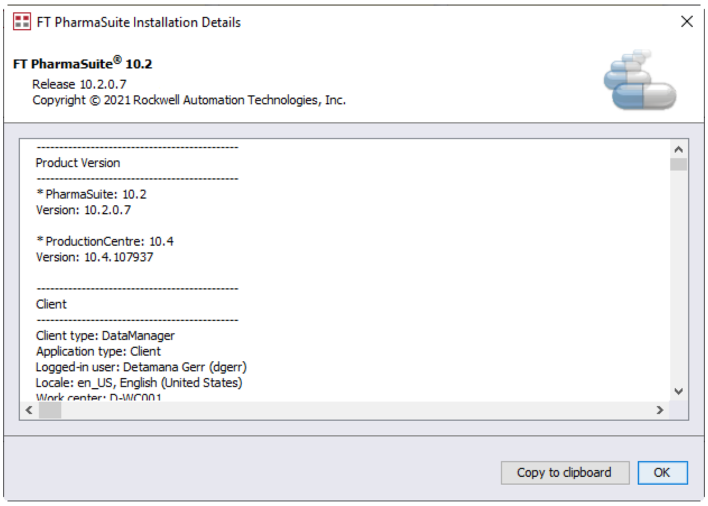
Database name: EV97
Database server time: 08/10/2021 03:08:02 PM CEST
Client time: 08/10/2021 03:08:02 PM CEST

Figure 6: About PharmaSuite

DETAILS

From the **PharmaSuite Installation Details** dialog, you can copy the listed detail data to the clipboard.

> **TIP**
>
> Please note that the path to the PharmaSuite log files is given in the last section of the listed detail data.



FT PharmaSuite Installation Details

FT PharmaSuite® 10.2
Release 10.2.0.7
Copyright © 2021 Rockwell Automation Technologies, Inc.

```
---------------------------------------------
Product Version
---------------------------------------------
* PharmaSuite: 10.2
Version: 10.2.0.7

* ProductionCentre: 10.4
Version: 10.4.107937

---------------------------------------------
Client
---------------------------------------------
Client type: DataManager
Application type: Client
Logged-in user: Detamana Gerr (dgerr)
Locale: en_US, English (United States)
Work center: D-WC001
```

Figure 7: PharmaSuite Installation Details

Editors

Data Manager provides a number of editors to facilitate easy editing of properties that make use of specific data types.

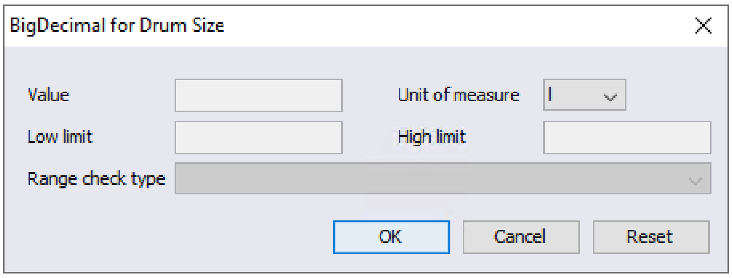
> **TIP**
>
> Please note that some editors are only available in the **Equipment** mode as they are specific to the data types of equipment objects.

- BigDecimal (page 11)
  floating point number, used for numbers and ranges. It is suitable for handling numbers qualified by a unit of measure.
- BigDecimal for automation integration accommodating three **Live Data types**:
  - Double (page 13)
  - Float (page 17)
  - Integer (page 20) (signed and unsigned)
- BigDecimal (page 24) for historical data access.
- Boolean (page 25) for automation integration.
- CleaningRules (page 26)
  used for defining the cleaning rules and cleaning demands of equipment, such as rooms.
- Date/Time (page 27)
  used for entering a date with or without time.
- Duration (page 29)
  used for displaying time spans and for time-related calculations.
- Expression (page 29)
  used for creating transition conditions and actions.
- FlexibleAttributeDefinition (page 53)
  used for defining a bundle of runtime attributes.
- FlexibleStateModel (page 55)
  status graph with statuses that are linked by transitions. Several individual statuses form a status group, which represents a collection of semantically connected statuses.
- FlexibleTagDefinition (page 56) for automation integration
  used to define a group of automation-relevant values to communicate as value bundle from an equipment entity for monitoring purposes.

- GraphStatusChange (page 58)
  used for changing graph-controlled statuses of entities including their expiry dates.
- Label Layout Selection (page 58)
  used for selecting a label layout for barcode labels of entities.
- Measured Value (page 59)
  used for entering values with a unit of measure.
- Multi-line Text (page 59)
  used for texts that typically exceed the length of a regular input box.
- Ranges (page 60)
  used for defining up to three weighing ranges per scale equipment.
- ScaleConfiguration (page 61)
  used for defining driver and connection data of a scale equipment.
- ScaleTestAndCalibration (page 65)
  used for defining test weight and operator instructions for scale tests and calibration.
- Searchable Option List (page 67)
  used for option lists with a large number of selectable items.
- String (page 67) for automation integration.
- String (page 69) for historical data access.
- Tag Browser (page 70) for automation tag selection.

BigDecimal Editor

The BigDecimal editor supports you with entering a number or a range between two numbers. It allows to define a unit of measure for the value or range.


BigDecimal for Drum Size

Value

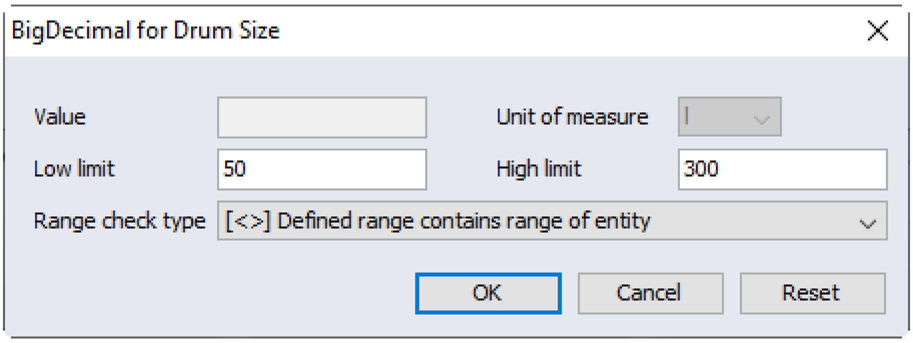
Unit of measure: l

Low limit

High limit

Range check type

Figure 8: BigDecimal editor for property type


BigDecimal for Drum Size

Value

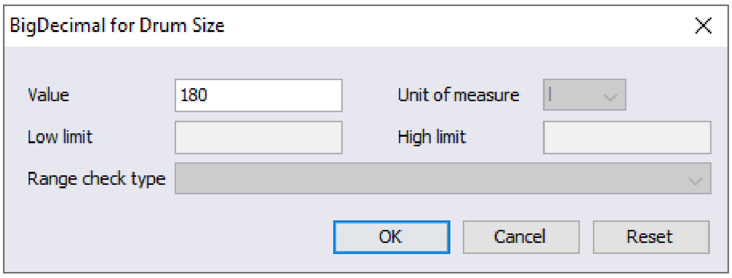
Unit of measure: l

Low limit: 50

High limit: 300

Range check type: [<>] Defined range contains range of entity

Figure 9: BigDecimal editor for property of equipment class


BigDecimal for Drum Size

Value: 180

Unit of measure: l

Low limit

High limit

Range check type

Figure 10: BigDecimal editor for property of equipment entity

- When you define a property type you can only select a unit of measure. If your property type is not based on a measured value or its unit of measure is not included as option in the list of available units of measure, you can leave the box blank.
  All other definitions need to be made when you assign the property type to a class or entity.
- When you define a property for an equipment class, you can either give an exact value or a range including its range check type. The unit of measure is specified with the property type and cannot be changed.
  You can leave non-required boxes blank.
  To save your value and close the BigDecimal editor, click the **OK** button.
- When you define a property for an equipment entity, you can fill the text boxes as required, either giving an exact value or a range. The unit of measure is specified with the property type and cannot be changed.
  You can leave non-required boxes blank.

TIP
Please note that you cannot define any values for properties of the **Runtime** usage. All input boxes are read-only then.

TIP

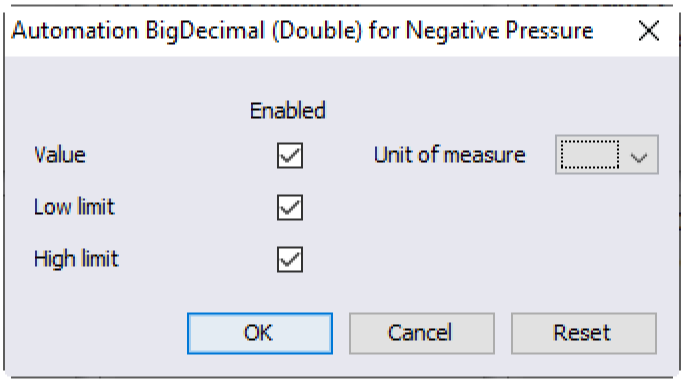
When defining a range, you can set the range's low and high limits and specify how the defined range is matched against the actual value or range provided by an equipment entity identified during execution:

- [<>] Defined range contains range of entity
  The option provides for equipment requirements where the actual value or range of an equipment entity must lie within the defined limits of minimum and maximum values.
  This check type would be suitable to cover a situation when you have an equipment unit that can be run with vessels of various sizes. Your requirement would, for example, specify a range between 300 l and 500 l and the volume of the identified vessel would have to be within this range. So, vessels with a volume of 350 l or 500 l could be identified successfully, whereas a vessel with a volume of 550 l would cause an exception.
- <[]> Defined range is contained in range of entity
  The option provides for equipment requirements where the actual range of an equipment entity must at least be able to cover the defined range.
  This check type would be suitable to cover a situation when you have a mixer that can run with a wide range of speeds. Your requirement would, for example, specify a range between 300 rpm and 600 rpm since the volatile material to be processed must be treated with mixing cycles at 350, 450, and 550 rpm but not at a lower speed to prevent lumping and not at a higher speed to prevent overheating. So, a mixer capable of running between 50 rpm and 2000 rpm would be identified successfully, whereas a mixer that runs at ranges between 200 rpm and 550 rpm would cause an exception.

- To save your data and close the BigDecimal editor, click the **OK** button.
- To close the BigDecimal editor without saving your changes, click the **Cancel** button.
- To reset your value or range to their last saved form, click the **Reset** button.

BigDecimal Editor for Double Data Types in Automation

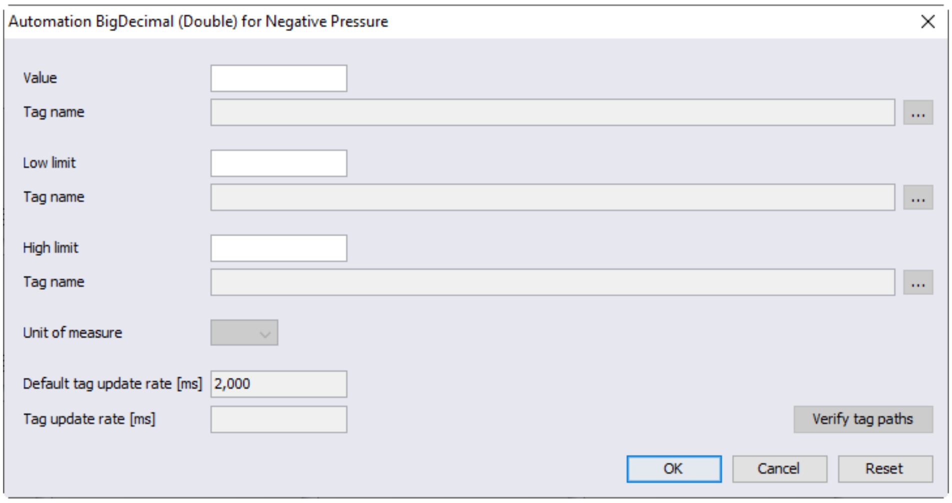
The BigDecimal editor supports you with entering a number or a range between two numbers. It allows to define a unit of measure for the value or range. To ensure a correct match with the automation tags provided by individual equipment entities, you need to indicate a suitable Live Data type (**Double**, **Float**, **Integer**). In addition, you can select verification schemas for tags that verify a value's tag health, its simulation status, or its maintenance status.



Automation BigDecimal (Double) for Negative Pressure

	Enabled		
Value	☑	Unit of measure	
Low limit	☑		
High limit	☑		

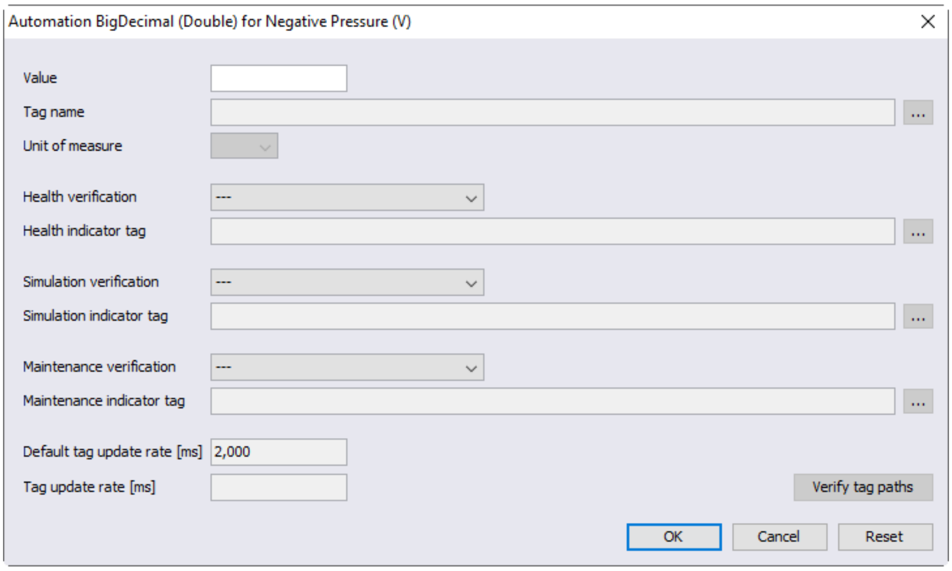
Figure 11: Automation BigDecimal (Double) editor for property type



Automation BigDecimal (Double) for Negative Pressure

Value
Tag name
Low limit
Tag name
High limit
Tag name
Unit of measure
Default tag update rate [ms] 2,000
Tag update rate [ms]

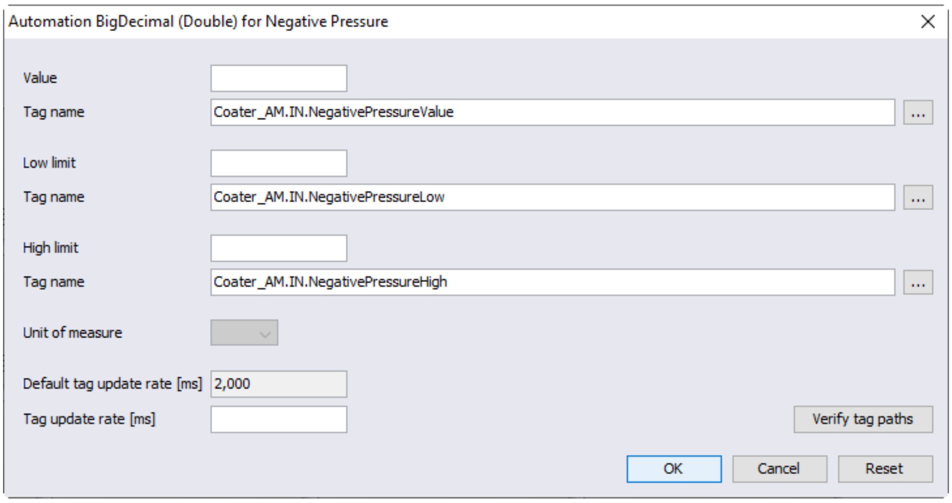
Figure 12: Automation BigDecimal (Double) editor for property of equipment class



Automation BigDecimal (Double) for Negative Pressure (V)

Value
Tag name
Unit of measure
Health verification ---
Health indicator tag
Simulation verification ---
Simulation indicator tag
Maintenance verification ---
Maintenance indicator tag
Default tag update rate [ms] 2,000
Tag update rate [ms]

Figure 13: Automation BigDecimal (Double) editor for property of equipment class with verification



Automation BigDecimal (Double) for Negative Pressure

Value:

Tag name: Coater_AM.IN.NegativePressureValue

Low limit:

Tag name: Coater_AM.IN.NegativePressureLow

High limit:

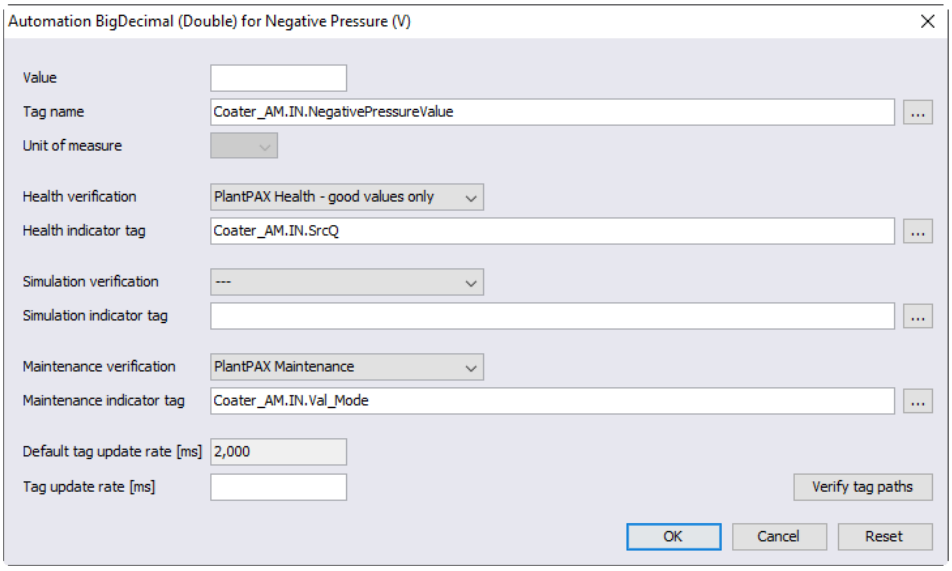
Tag name: Coater_AM.IN.NegativePressureHigh

Unit of measure:

Default tag update rate [ms]: 2,000

Tag update rate [ms]:

Figure 14: Automation BigDecimal (Double) editor for property of equipment entity



Automation BigDecimal (Double) for Negative Pressure (V)

Value:

Tag name: Coater_AM.IN.NegativePressureValue

Unit of measure:

Health verification: PlantPAX Health - good values only

Health indicator tag: Coater_AM.IN.SrcQ

Simulation verification: ---

Simulation indicator tag:

Maintenance verification: PlantPAX Maintenance

Maintenance indicator tag: Coater_AM.IN.Val_Mode

Default tag update rate [ms]: 2,000

Tag update rate [ms]:

Figure 15: Automation BigDecimal (Double) editor for property of equipment entity with verification

- When you define a property type, you need to specify which of the theoretically available values of an equipment entity (**Value**, **High limit**, **Low limit**) are actually going to be used during processing.
  Additionally, you can select a unit of measure. If your property type is not based on a measured value or its unit of measure is not included as option in the list of available units of measure, you can leave the box blank.

  > TIP
  > If you wish to use verification schemas to check the health, simulation status, and maintenance status of your tags, make sure to select **Value** as the only option. The system does not provide the verification feature for high and low limits.

- When you define a property for an equipment class, you can fill the text boxes for all the values specified with the property type. If the property type only considers the **Value** without limits, you can also select verification schemas. This data, however, only has informational value and is neither matched against the data defined for entities of the class nor against data defined with a property requirement during recipe or workflow design.
  The unit of measure is specified with the property type and cannot be changed.

- When you define a property for an equipment entity, you can fill the text boxes for all the values specified with the property type. If the property type only considers the **Value** without limits, you can also select verification schemas. The unit of measure is specified with the property type and cannot be changed.
  At this point you need to define to which tags of your equipment you wish to map the defined value, range, or schema and with which update rate the system is to access the tags.

  > TIP
  > To see the list of tags available at your equipment and to select the suitable tag, click the **...** button to open the Tag Browser editor (page 70).

  You can leave non-required boxes blank.
  To check if the indicated tag names correspond to existing tag names on the equipment entity, click the **Verify tag paths** button.

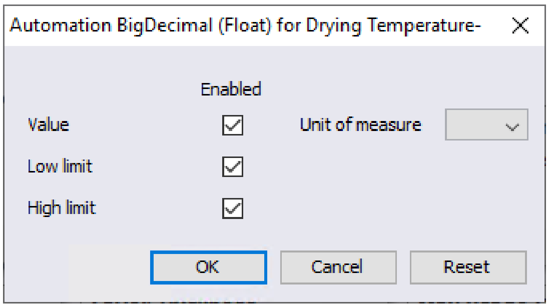
- To save your data and close the Automation BigDecimal editor, click the **OK** button.

- To close the Automation BigDecimal editor without saving your changes, click the **Cancel** button.

- To reset your data to its last saved form, click the **Reset** button.

BigDecimal Editor for Float Data Types in Automation

The BigDecimal editor supports you with entering a number or a range between two numbers. It allows to define a unit of measure for the value or range. To ensure a correct match with the automation tags provided by individual equipment entities, you need to indicate a suitable Live Data type (**Double**, **Float**, **Integer**). In addition, you can select verification schemas for tags that verify a value's tag health, its simulation status, or its maintenance status.



Automation BigDecimal (Float) for Drying Temperature-

Enabled

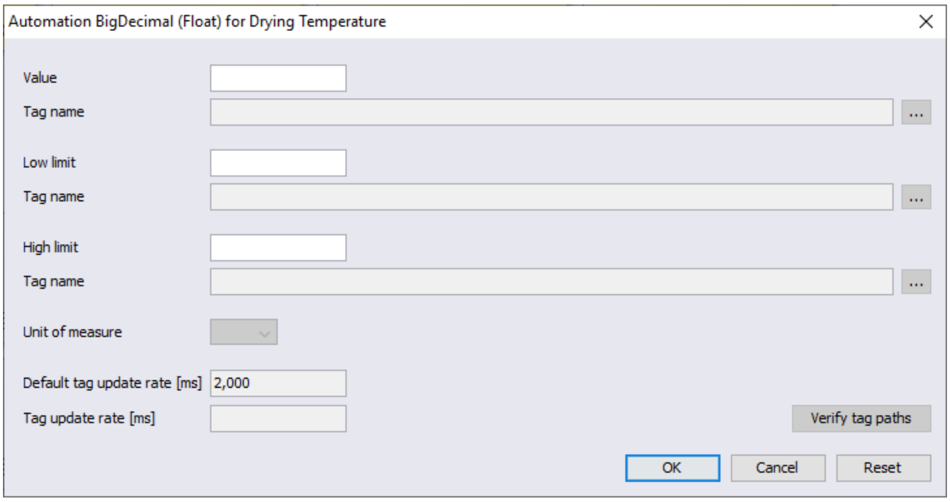
Value ☑

Low limit ☑

High limit ☑

Unit of measure

Figure 16: Automation BigDecimal (Float) editor for property type



Automation BigDecimal (Float) for Drying Temperature

Value

Tag name

Low limit

Tag name

High limit

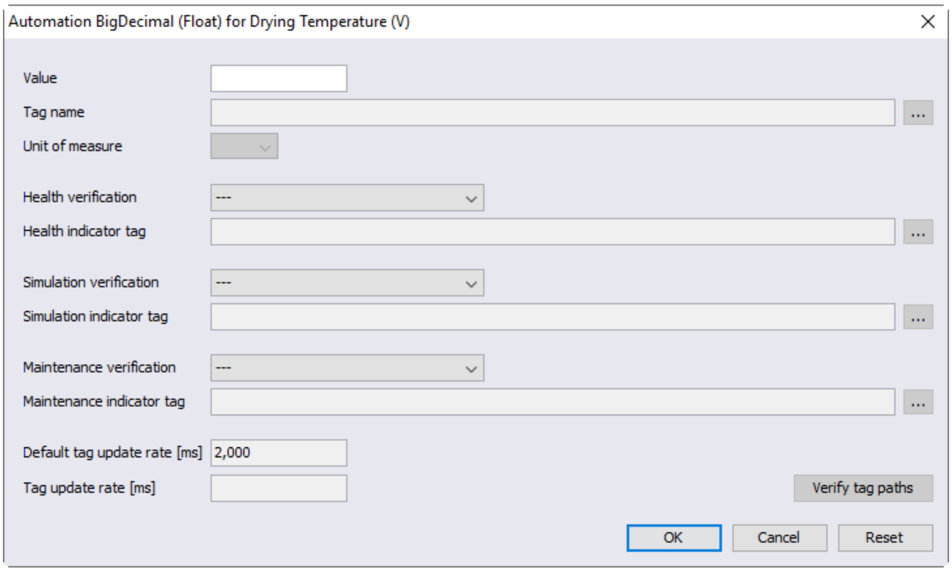
Tag name

Unit of measure

Default tag update rate [ms] 2,000

Tag update rate [ms]

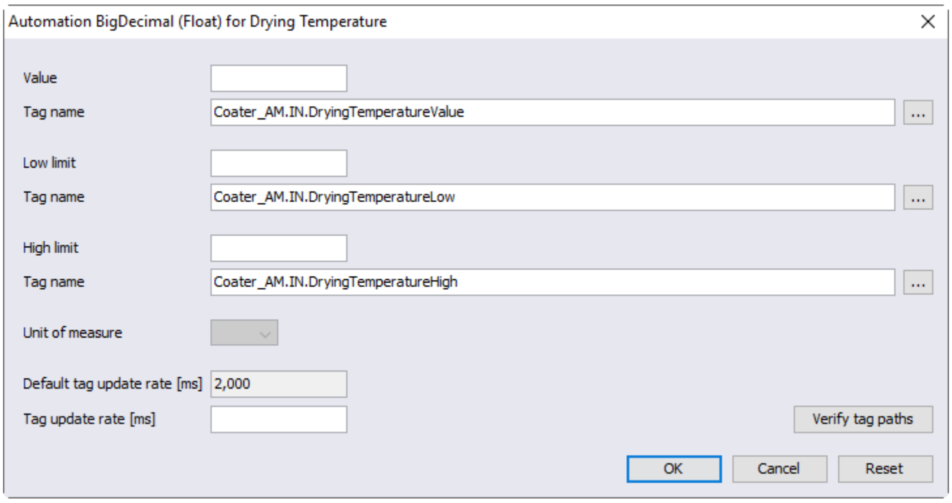
Figure 17: Automation BigDecimal (Float) editor for property of equipment class



Automation BigDecimal (Float) for Drying Temperature (V)

Value	
Tag name	
Unit of measure	
Health verification	---
Health indicator tag	
Simulation verification	---
Simulation indicator tag	
Maintenance verification	---
Maintenance indicator tag	
Default tag update rate [ms]	2,000
Tag update rate [ms]	

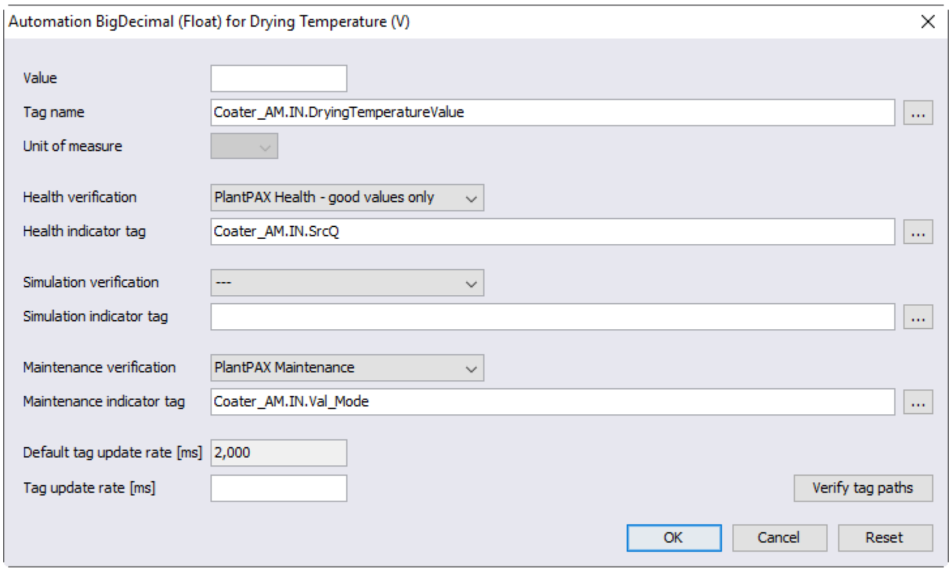
Figure 18: Automation BigDecimal (Float) editor for property of equipment class with verification



Automation BigDecimal (Float) for Drying Temperature

Value	
Tag name	Coater_AM.IN.DryingTemperatureValue
Low limit	
Tag name	Coater_AM.IN.DryingTemperatureLow
High limit	
Tag name	Coater_AM.IN.DryingTemperatureHigh
Unit of measure	
Default tag update rate [ms]	2,000
Tag update rate [ms]	

Figure 19: Automation BigDecimal (Float) editor for property of equipment entity



Automation BigDecimal (Float) for Drying Temperature (V)

Value	
Tag name	Coater_AM.IN.DryingTemperatureValue
Unit of measure	
Health verification	PlantPAX Health - good values only
Health indicator tag	Coater_AM.IN.SrcQ
Simulation verification	---
Simulation indicator tag	
Maintenance verification	PlantPAX Maintenance
Maintenance indicator tag	Coater_AM.IN.Val_Mode
Default tag update rate [ms]	2,000
Tag update rate [ms]	

Figure 20: Automation BigDecimal (Float) editor for property of equipment entity with verification

- When you define a property type, you need to specify which of the theoretically available values of an equipment entity (**Value**, **High limit**, **Low limit**) are actually going to be used during processing.
  Additionally, you can select a unit of measure. If your property type is not based on a measured value or its unit of measure is not included as option in the list of available units of measure, you can leave the box blank.

> **TIP**
> If you wish to use verification schemas to check the health, simulation status, and maintenance status of your tags, make sure to select **Value** as the only option. The system does not provide the verification feature for high and low limits.

- When you define a property for an equipment class, you can fill the text boxes for all the values specified with the property type. If the property type only considers the **Value** without limits, you can also select verification schemas. This data, however, only has informational value and is neither matched against the data defined for entities of the class nor against data defined with a property requirement during recipe or workflow design.
  The unit of measure is specified with the property type and cannot be changed.
- When you define a property for an equipment entity, you can fill the text boxes for all the values specified with the property type. If the property type only considers the **Value** without limits, you can also select verification schemas. The unit of measure is specified with the property type and cannot be changed.
  At this point you need to define to which tags of your equipment you wish to map the defined value, range, or schema and with which update rate the system is to access the tags.

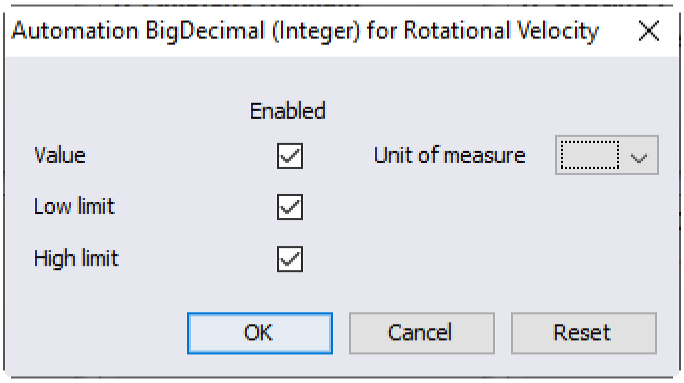
TIP
To see the list of tags available at your equipment and to select the suitable tag, click the **...** button to open the Tag Browser editor (page 70).

You can leave non-required boxes blank.
To check if the indicated tag names correspond to existing tag names on the equipment entity, click the **Verify tag paths** button.

- To save your data and close the Automation BigDecimal editor, click the **OK** button.
- To close the Automation BigDecimal editor without saving your changes, click the **Cancel** button.
- To reset your data to its last saved form, click the **Reset** button.

BigDecimal Editor for Integer Data Types in Automation

The BigDecimal editor supports you with entering a number or a range between two numbers. It allows to define a unit of measure for the value or range. To ensure a correct match with the automation tags provided by individual equipment entities, you need to indicate a suitable Live Data type (**Double**, **Float**, **Integer**). In addition, you can select verification schemas for tags that verify a value's tag health, its simulation status, or its maintenance status.

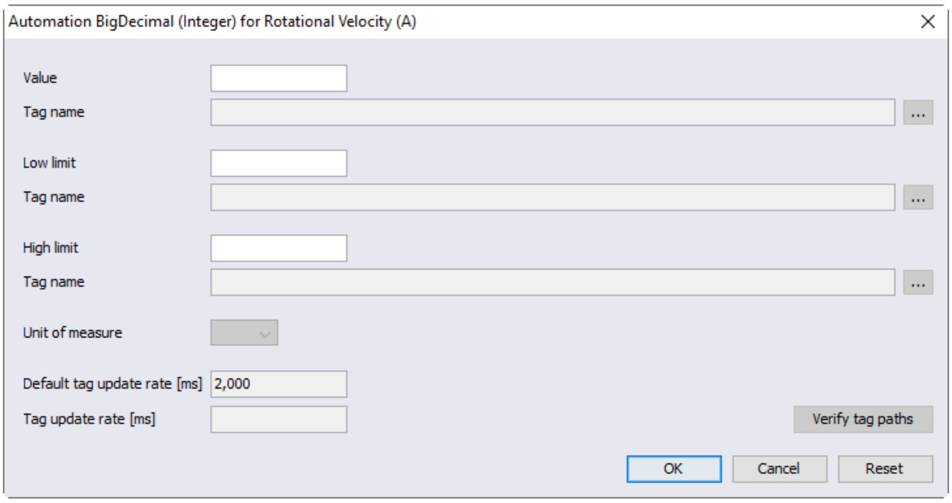


Automation BigDecimal (Integer) for Rotational Velocity

	Enabled
Value	☑
Low limit	☑
High limit	☑

Unit of measure

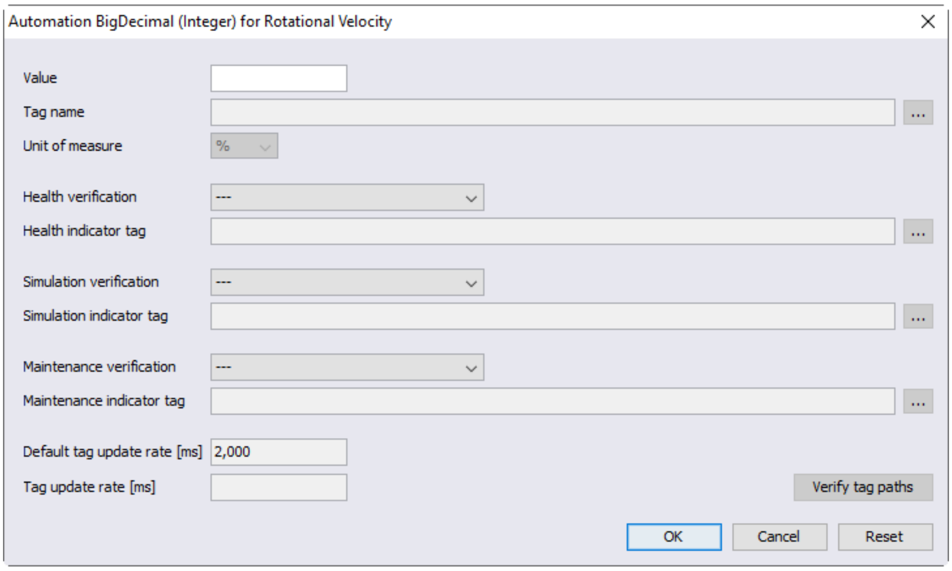
Figure 21: Automation BigDecimal (Integer) editor for property type



Automation BigDecimal (Integer) for Rotational Velocity (A)

Default tag update rate [ms] 2,000

Figure 22: Automation BigDecimal (Integer) editor for property of equipment class



Automation BigDecimal (Integer) for Rotational Velocity

Unit of measure %

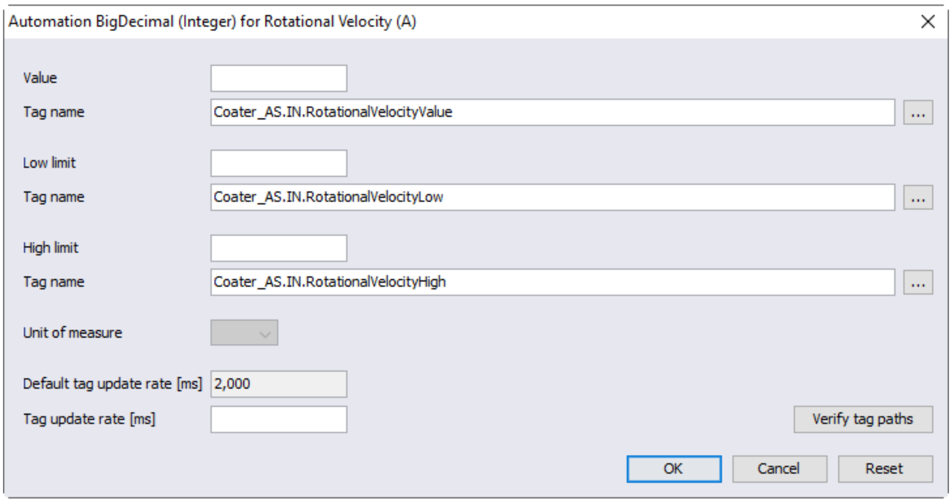
Health verification ---

Simulation verification ---

Maintenance verification ---

Default tag update rate [ms] 2,000

Figure 23: Automation BigDecimal (Integer) editor for property of equipment class with verification



Automation BigDecimal (Integer) for Rotational Velocity (A)

Value:

Tag name Coater_AS.IN.RotationalVelocityValue

Low limit:

Tag name Coater_AS.IN.RotationalVelocityLow

High limit:

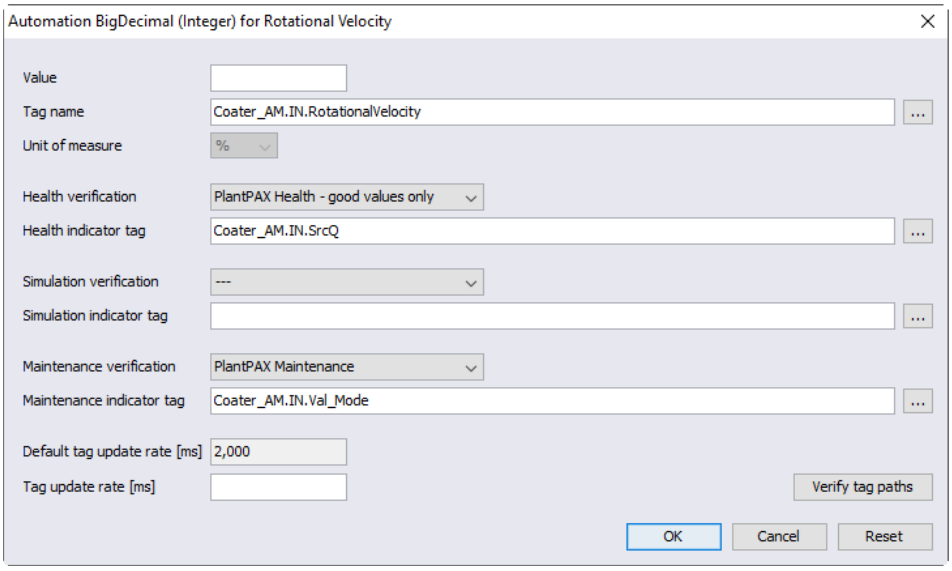
Tag name Coater_AS.IN.RotationalVelocityHigh

Unit of measure:

Default tag update rate [ms]: 2,000

Tag update rate [ms]:

Figure 24: Automation BigDecimal (Integer) editor for property of equipment entity



Automation BigDecimal (Integer) for Rotational Velocity

Value:

Tag name Coater_AM.IN.RotationalVelocity

Unit of measure: %

Health verification: PlantPAX Health - good values only

Health indicator tag: Coater_AM.IN.SrcQ

Simulation verification: ---

Simulation indicator tag:

Maintenance verification: PlantPAX Maintenance

Maintenance indicator tag: Coater_AM.IN.Val_Mode

Default tag update rate [ms]: 2,000

Tag update rate [ms]:

Figure 25: Automation BigDecimal (Integer) editor for property of equipment entity with verification

- When you define a property type, you need to specify which of the theoretically available values of an equipment entity (**Value**, **High limit**, **Low limit**) are actually going to be used during processing.
  Additionally, you can select a unit of measure. If your property type is not based on a measured value or its unit of measure is not included as option in the list of available units of measure, you can leave the box blank.

TIP

If you wish to use verification schemas to check the health, simulation status, and maintenance status of your tags, make sure to select **Value** as the only option. The system does not provide the verification feature for high and low limits.

- When you define a property for an equipment class, you can fill the text boxes for all the values specified with the property type. If the property type only considers the **Value** without limits, you can also select verification schemas. This data, however, only has informational value and is neither matched against the data defined for entities of the class nor against data defined with a property requirement during recipe or workflow design.
  The unit of measure is specified with the property type and cannot be changed.
- When you define a property for an equipment entity, you can fill the text boxes for all the values specified with the property type. If the property type only considers the **Value** without limits, you can also select verification schemas. The unit of measure is specified with the property type and cannot be changed.
  At this point you need to define to which tags of your equipment you wish to map the defined value, range, or schema and with which update rate the system is to access the tags.

  > TIP
  > To see the list of tags available at your equipment and to select the suitable tag, click the **...** button to open the Tag Browser editor (page 70).

  You can leave non-required boxes blank.
  To check if the indicated tag names correspond to existing tag names on the equipment entity, click the **Verify tag paths** button.
- To save your data and close the Automation BigDecimal editor, click the **OK** button.
- To close the Automation BigDecimal editor without saving your changes, click the **Cancel** button.
- To reset your data to its last saved form, click the **Reset** button.

BigDecimal Editor for Historian

The BigDecimal editor for Historian supports you with retrieving numeric values from a historical data archive. It allows to define a unit of measure for the values.

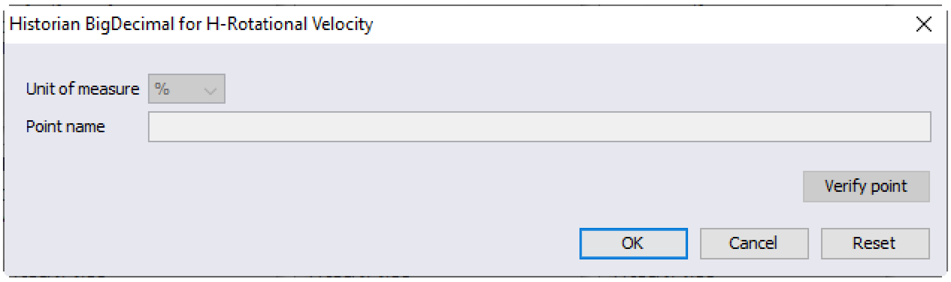
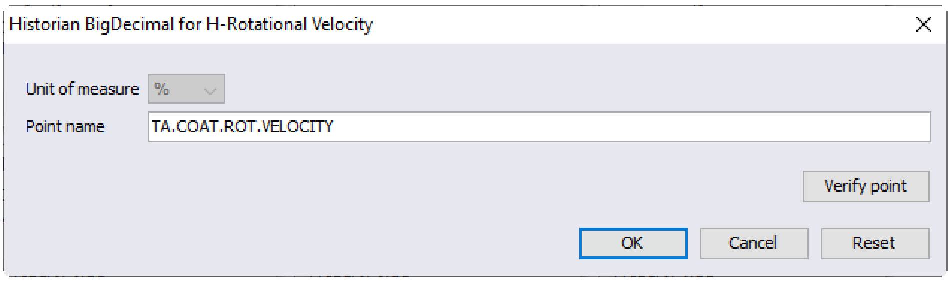


Figure 26: Historian BigDecimal editor for property of equipment class



Unit of measure: %

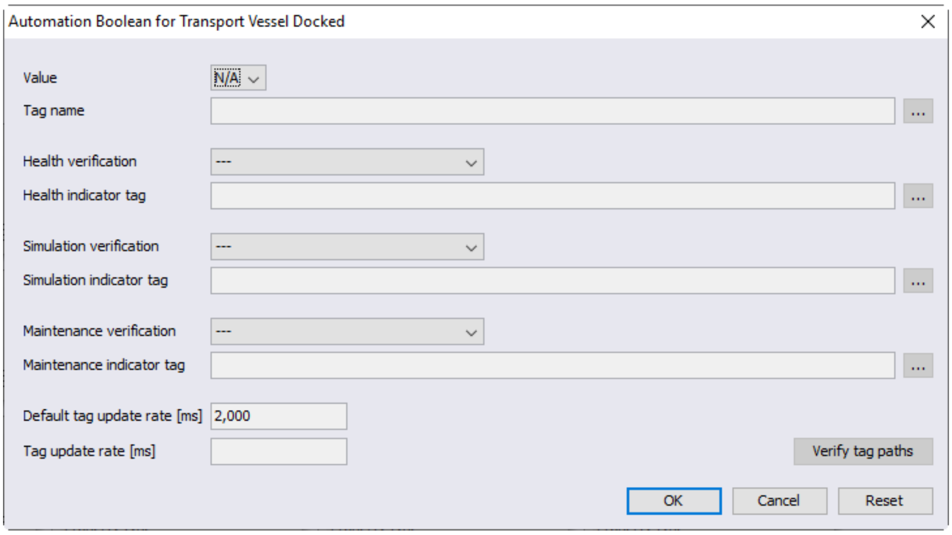
Point name: TA.COAT.ROT.VELOCITY

Figure 27: Historian BigDecimal editor for property of equipment entity

- When you define a property for an equipment class, you can only view the unit of measure defined with the property type.
- When you define a property for an equipment entity, you need to define which point of your Historian server you wish access to retrieve values. The unit of measure is specified with the property type and cannot be changed.
  To check if the indicated point name corresponds to an existing point defined on the Historian server, click the **Verify point** button.
- To save your data and close the Historian BigDecimal editor, click the **OK** button.
- To close the Historian BigDecimal editor without saving your changes, click the **Cancel** button.
- To reset your data to its last saved form, click the **Reset** button.

Boolean Editor for Automation

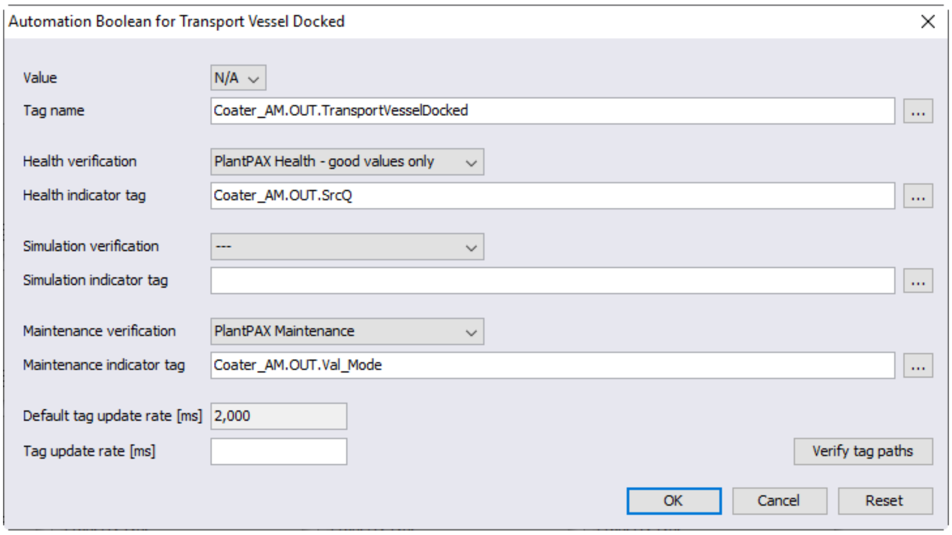
The Boolean editor supports you with handling Boolean values (**Yes**, **No**, and **N/A**), for example to define specific true/false conditions for an equipment entity like "Transport Vessel docked", which need to be communicated to or from an automated equipment entity. In addition, you can select verification schemas for tags that verify a value's tag health, its simulation status, or its maintenance status.



Automation Boolean for Transport Vessel Docked

Value	N/A
Tag name	
Health verification	---
Health indicator tag	
Simulation verification	---
Simulation indicator tag	
Maintenance verification	---
Maintenance indicator tag	
Default tag update rate [ms]	2,000
Tag update rate [ms]	

Figure 28: Automation Boolean editor for property of equipment class



Automation Boolean for Transport Vessel Docked

Value	N/A
Tag name	Coater_AM.OUT.TransportVesselDocked
Health verification	PlantPAX Health - good values only
Health indicator tag	Coater_AM.OUT.SrcQ
Simulation verification	---
Simulation indicator tag	
Maintenance verification	PlantPAX Maintenance
Maintenance indicator tag	Coater_AM.OUT.Val_Mode
Default tag update rate [ms]	2,000
Tag update rate [ms]	

Figure 29: Automation Boolean editor for property of equipment entity

- When you define a property for an equipment class, you can set a value and select verification schemas. The data only has informational value, however and is neither matched against the data defined for entities of the class nor against data defined with a property requirement during recipe or workflow design.

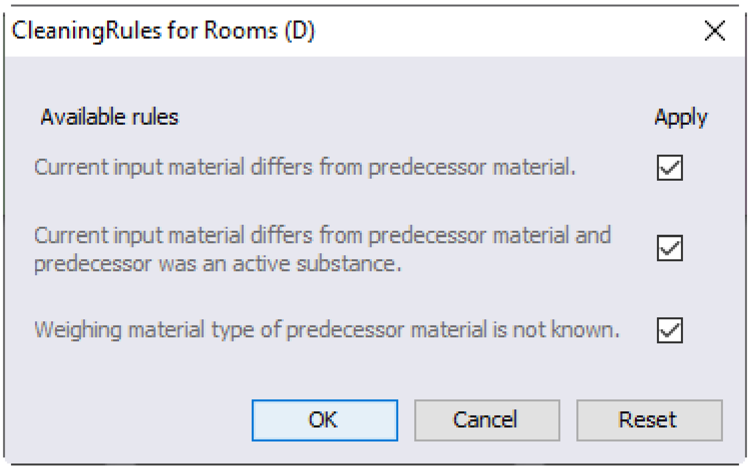
- When you define a property for an equipment entity, you can set a value and select verification schemes. At this point you also need to define to which tags of your equipment you wish to map the defined value and schemes, and with which update rate the system is to access the tags.

  > TIP
  > To see the list of tags available at your equipment and to select the suitable tag, click the **...** button to open the Tag Browser editor (page 70).

  To check if the indicated tag name corresponds to an existing tag name on the equipment entity, click the **Verify tag paths** button.
- To save your data and close the Automation Boolean editor, click the **OK** button.
- To close the Automation Boolean editor without saving your changes, click the **Cancel** button.
- To reset your data to its last saved form, click the **Reset** button.

CleaningRules Editor

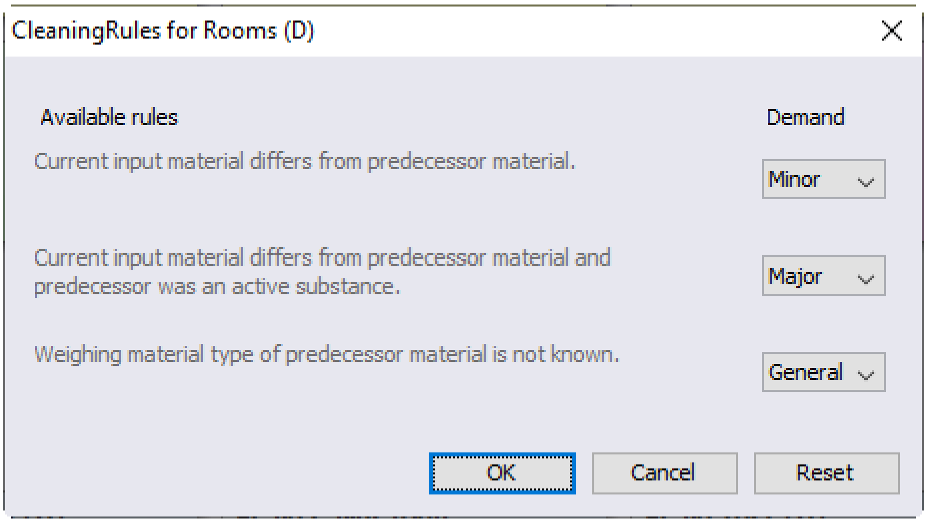
The CleaningRules editor supports you with defining which of the available cleaning rules are to apply to an entity and which cleaning demand they involve.



CleaningRules for Rooms (D)

Available rules	Apply
Current input material differs from predecessor material.	☑
Current input material differs from predecessor material and predecessor was an active substance.	☑
Weighing material type of predecessor material is not known.	☑

Figure 30: CleaningRules editor for property type



CleaningRules for Rooms (D)

Available rules	Demand
Current input material differs from predecessor material.	Minor
Current input material differs from predecessor material and predecessor was an active substance.	Major
Weighing material type of predecessor material is not known.	General

Figure 31: CleaningRules editor for property

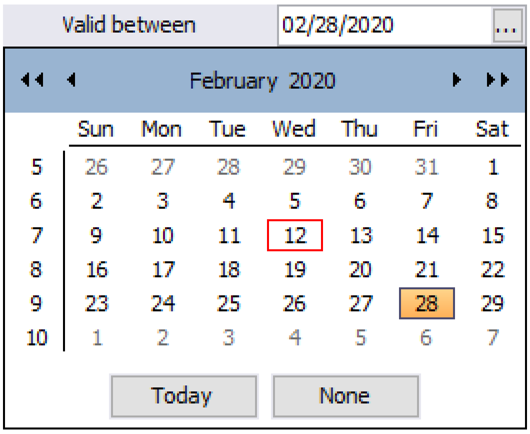
- When you define a property type, you need to specify which of the available pre-defined rules are relevant to the intended use of the property and select them to apply.
- When you define a property for an equipment class or entity, you can set the cleaning demand (**None**, **Minor**, **Major**, **General**) that must be fulfilled when one of the rules applies during execution.
- To save your data and close the CleaningRules editor, click the **OK** button.
- To close the CleaningRules editor without saving your changes, click the **Cancel** button.
- To reset your data to its last saved form, click the **Reset** button.

Date/Time Picker Editor

The Date/Time Picker editor supports you with entering a date with or without a time. You can either type the date or time directly in the input box or open the calendar control to pick it. Depending on the data expected by the input box you are about to fill, the calendar either only provides date picking or it holds an additional spin control to pick a time as well. The calendar opens with the date (and time) that is currently displayed in the input box. It highlights the selected date in orange and displays the current date with a red border.

For occurrences that only require a date, the calendar control closes automatically when you have entered the date.

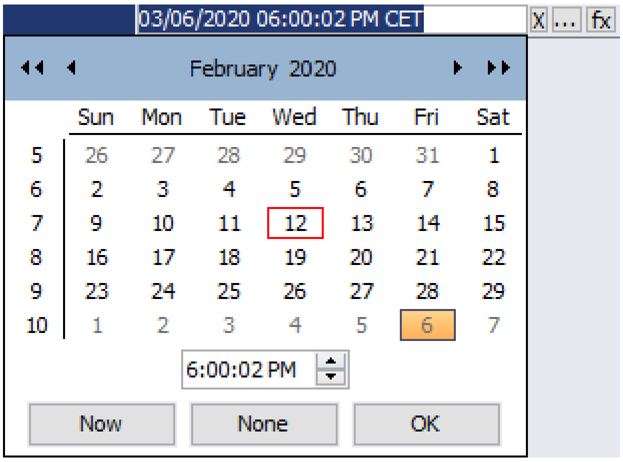
For occurrences that require both a date and a time, the calendar control remains open after you have made an entry to allow you to set all data at once. The editor provides an **OK** button to close it when you have entered both date and time.



Valid between 02/28/2020

	Sun	Mon	Tue	Wed	Thu	Fri	Sat
5	26	27	28	29	30	31	1
6	2	3	4	5	6	7	8
7	9	10	11	12	13	14	15
8	16	17	18	19	20	21	22
9	23	24	25	26	27	28	29
10	1	2	3	4	5	6	7

Figure 32: Date/Time Picker editor



03/06/2020 06:00:02 PM CET

	Sun	Mon	Tue	Wed	Thu	Fri	Sat
5	26	27	28	29	30	31	1
6	2	3	4	5	6	7	8
7	9	10	11	12	13	14	15
8	16	17	18	19	20	21	22
9	23	24	25	26	27	28	29
10	1	2	3	4	5	6	7

6:00:02 PM

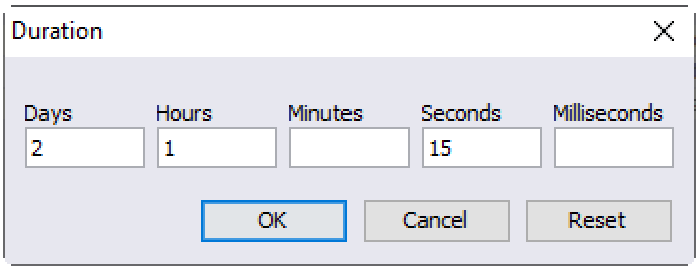
Figure 33: Date/Time Picker editor

- To navigate the calendar to other months or years, use the controls in the calendar header.
  - Months: SINGLE ARROW keys or option list that appears when you click the current month name.
  - Years: DOUBLE ARROW keys or spin control that appears when you click the current year.
- To edit a time, click the position (hours, minutes, seconds) in the time text box you wish to change. Then use the spin control or the UP and DOWN ARROW keys to increase or decrease the number.
- In the Date Picker, to select the current date, click the **Today** button.
- In the Date and Time Picker, to select the current date and time, click the **Now** button.
- To clear the date and or time from the input box, click the **None** button.

- To close the calendar control without changing the date or time, click anywhere in the dialog outside the calendar control, press the ESC key, or click the **OK** button (if present).

Duration Editor

The Duration editor supports you with entering a duration with days as largest and milliseconds as smallest unit.



Duration

Days	Hours	Minutes	Seconds	Milliseconds
2	1		15	

Figure 34: Duration editor

- Fill the text boxes as required. You can leave non-required boxes blank. To save your value and close the Duration editor, click the **OK** button or press the ENTER key.

> **TIP**
> Please note that the system does not allow you to type values for the various time units that exceed the maximum number of each time unit and would move it to the next higher unit. Thus, for hours the maximum is 23, for minutes and seconds it is 59, and for milliseconds it is 999.

- To close the Duration editor without saving your changes, click the **Cancel** button or press the ESC key.
- To reset your duration values to their last saved form, click the **Reset** button.

Expression Editor

The Expression editor supports you with defining

- conditions of graph transitions (page 30), which determine if the expected circumstances are met so that the transition can be executed.
- actions of graph transitions (page 31), which are performed along with the transition.

Expression Editor for Transition Conditions

The Expression editor for transitions conditions consists of

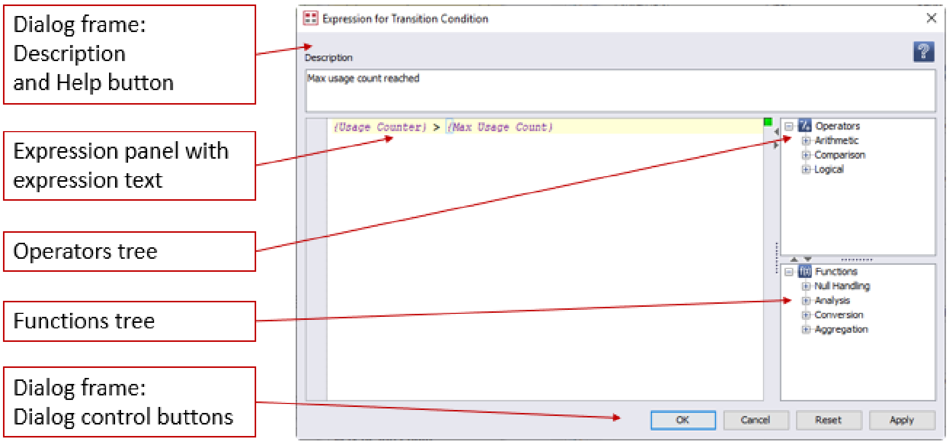
- the dialog frame for expression meta data
- the expression panel (page 32) that holds an expression with a Boolean result that determines if a transition is executed or not:
  - if the result is **undefined**, the transition cannot be executed and leads to an error.

> **TIP**
> This can happen if the condition references a runtime property that has not yet been initialized with a value. Unless you are sure that the property has been initialized before the system executes the condition, make it more robust by using the **nvl** function (page 48).
> Example:
> nvl({Usage Counter}, 0BDecimal) < 5BDecimal

  - if the result is **true**, the transition is executed.

> **TIP**
> Please note that an empty condition returns **true**, which executes the transition.

  - if the result is **false**, the transition is not executed.
- two resizable tree panels that list available operators (page 42) and functions (page 48).


Dialog frame: Description and Help button
Expression panel with expression text
Operators tree
Functions tree
Dialog frame: Dialog control buttons
Expression for Transition Condition
Operators
Arithmetic
Comparison
Logical
Functions
Null Handling
Analysis
Conversion
Aggregation
OK
Cancel
Reset
Apply

Description

Max usage count reached

```
{Usage Counter} > {Max Usage Count}
```

Figure 35: Expression editor for transition conditions

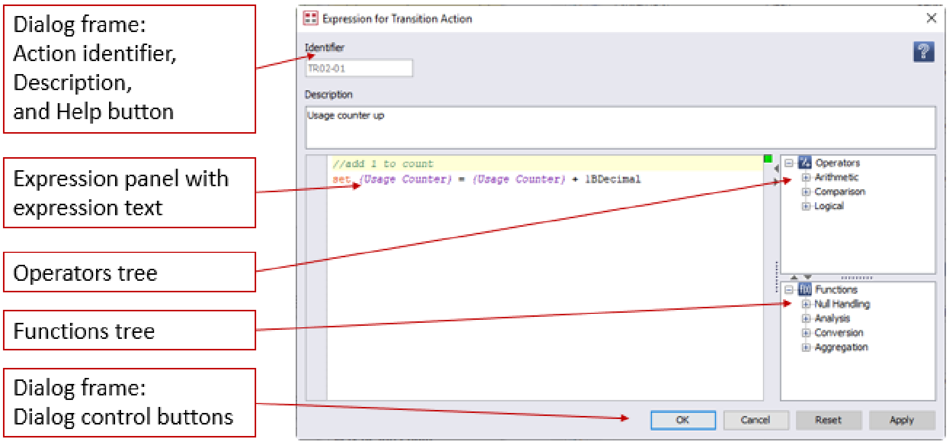
Above the expression panel, the frame provides a text box where you can type a textual description of the condition. The system shows the description in the respective table cell on the **Transition** tab of a graph's Details window.
The **Help** button opens the context-sensitive help system.
Below the expression panel, the dialog control buttons provide the following actions:

- To save your expression and close the Expression editor, click the **OK** button.
- To close the Expression editor without saving your changes, click the **Cancel** button.
- To reset your expression to its last saved form, click the **Reset** button.
- To save your expression without closing the Expression editor, click the **Apply** button.

Expression Editor for Transition Actions

The Expression editor for transitions actions consists of

- the dialog frame for expression meta data
- the expression panel (page 32) that holds an action expression with a command and a reference to the property type value to be affected by the command.
- two resizable tree panels that list available operators (page 42) and functions (page 48).



Identifier
TR02-01

Description
Usage counter up

```
//add 1 to count
set (Usage Counter) = (Usage Counter) + 1BDecimal
```

Figure 36: Expression editor for transition actions

Above the expression panel, the frame provides the following elements:

- The **Identifier** box shows the system-generated identifier, which is created when you save the expression for the first time and cannot be edited. The system also displays the identifier in the **ID** column of the action sub-table on the **Transition** tab of the graph's Details window.
- In the **Description** box, you can type a textual description of the transition action. The system also displays the description in the action sub-table on the **Transition** tab of the graph's Details window.
- The **Help** button opens the context-sensitive help system.

Below the expression panel, the dialog control buttons provide the following actions:

- To save your expression and close the Expression editor, click the **OK** button.
- To close the Expression editor without saving your changes, click the **Cancel** button.
- To reset your expression to its last saved form, click the **Reset** button.
- To save your expression without closing the Expression editor, click the **Apply** button.

Expression Panel

The expression panel represents the work area of the Expression editor and is basically an enhanced text editor that supports editing of expressions and references with concurrent syntax and semantics validation.

To format your expressions for better readability you can use blanks, tabs, and line breaks. Your formatting is retained when you save an expression.
The editor supports text manipulation by keyboard with common shortcuts (page 82).

TIP

Please note the following notation conventions for numbers in expressions:

- "." is the only permitted decimal separator as in
  678.934
- There are no digit grouping symbols, as in
  69352990478.749

You can build expressions by using the following elements:

Constants (page 37)

Operators (page 42)

Functions (page 48)

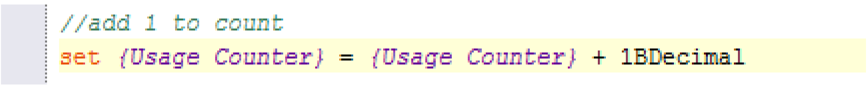
Brackets

- Only round brackets (parentheses) are allowed in expressions as well as the identifiers of referenced attributes. You can use them to override or simply to clarify the precedence of the operators used in an expression.
- Square brackets are not allowed in expressions, but within object identifiers or attributes.
- Pointy brackets cannot be used in expressions, since they would be interpreted as inequality signs (greater than, smaller than). They are allowed within phase and output identifiers.
- Curly brackets (braces) are always interpreted as start or end delimiter of a referenced attribute and are therefore not allowed within the identifiers or attributes.

Comments

You can add comments to your expressions, which the system will disregard when it processes the expression:

- Use // to introduce a single-line comment.
- Use /* **...** */ to enclose a multi-line comment.



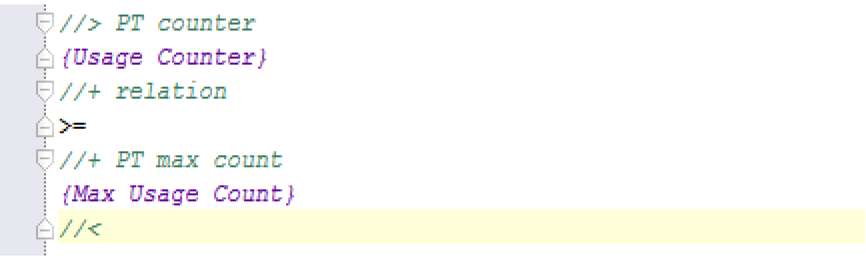
```
//add 1 to count
set {Usage Counter} = {Usage Counter} + 1BDecimal
```

Figure 37: Expression with comment

SECTIONS

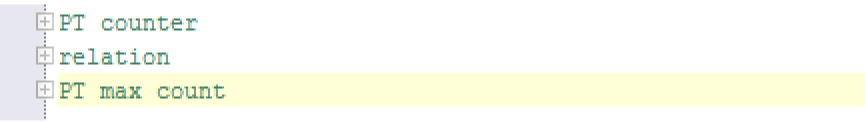
To add further structuring to your expressions, you can convert comments into sections:

- Use //> to start a section.
- Use //+ to continue a section.
- Use //< to close a section.



```
//> PT counter
{Usage Counter}
//+ relation
>=
//+ PT max count
{Max Usage Count}
//<
```

Figure 38: Expression with sections (all open)



```
PT counter
relation
PT max count
```

Figure 39: Expression with sections (all closed)

REFERENCES TO ATTRIBUTE VALUES OF PROPERTY TYPES

An attribute reference consists of the identifier of the property type whose attribute value needs to be retrieved.
For property types that only have one attribute to provide a value, the system automatically retrieves the value. Property types of the FlexibleAttributeDefinition data type, however, can have several attributes. Similarly, Specification property types of the BigDecimal data type can provide a minimum and a maximum in addition to a value. The property type identifiers and attributes are enclosed in curly brackets (braces) with a period as separator between a property type and its attribute.
Examples:

- reference to a property type that provides only one value, such as a Runtime property type of the String data type.
  {Batch ID}
- reference to a property type that provides value, minimum, and maximum, such as a Specification property type of the BigDecimal data type
  {Drum Size}.{minValue}
- reference to a property type that provides more than one value, such as a Runtime property type of the FlexibleAttributeDefinition data type
  {Coater Prepared}.{Target Vessel ID}

TIP
When you hover over a reference, the system displays a tooltip that also shows the usage and the data type of the referenced object.

COMMANDS

Commands are only suited for use in transition actions. They represent the set of instructions available for manipulating the values of runtime property types. In the Expression editor they are displayed in orange.

TIP
Please note that an action must only contain one command.

The following commands are available:

- `set`
  allows to change the value of a property type, either to a constant or to a calculated value.
  Example:
  set {Usage Counter} = {Usage Counter} + 1BDecimal
  Increases the given usage counter by 1.
  This command example can be used as action in a status transition such as from "In use" to "In use" until the maximum count has been reached and the status transition must not be repeated.

TIP
If the right part of the equation is **undefined**, the action cannot be executed and leads to an error.
This can happen if it references a runtime property that has not yet been initialized with a value. Unless you are sure that the property has been initialized before the system executes the action, make it more robust by using the **nvl** function (page 48).
Example:
set {Usage Counter} = nvl({Usage Counter}, 0BDecimal) + 1BDecimal

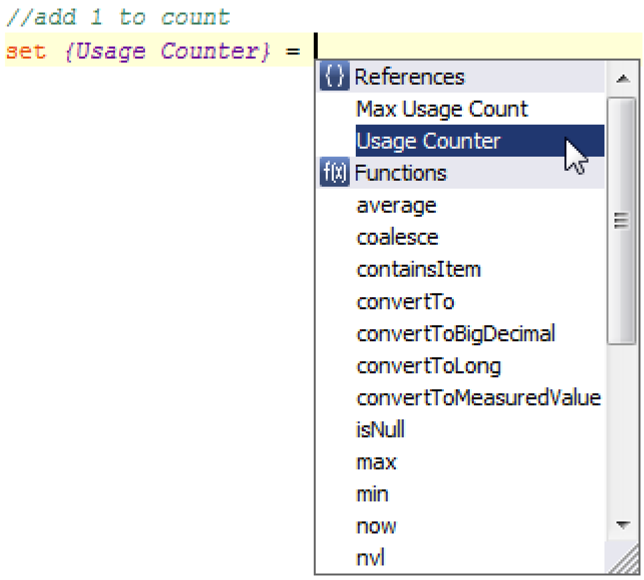
- `clear`
  allows to delete the value of a property type, setting it to **undefined**.
  Example:
  clear {Usage Counter}
  Resets the value of the given usage counter.
  This command example can be used as action in a status transition such as from "Maintenance required" to "Maintenance performed" so that a new usage counter can be started with the next use of the equipment.

- `archive`
  forces the current entity to change its status to **Archived**.
  Example:
  archive
  This command is intended for use with generated entities that are to be archived after use.

For inserting expression elements, you can either type the texts yourself or make use of the intelligent auto-completion feature provided by the editor:

1. When you start typing, the system displays all available, suitable, and matching elements in a drop-down option list. The search is not case-sensitive, and the available options are sorted by their element types.
2. To open the full list of available expression elements, place your cursor in the empty expression panel and press the CTRL+SPACEBAR keyboard shortcut.
3. Click the element you wish to insert or use the ARROW UP or ARROW DOWN keys to select it and then press the ENTER key to insert it.
   To close the drop-down list without inserting an option, click anywhere outside of the list or press the ESC key.
4. You can edit your expression manually at all times. Click anywhere in the expression panel to place your cursor and then type or delete characters as required.

> **TIP**
> Output identifiers are listed along with their respective data types. This information can help you to avoid using non-matching data types accidentally.

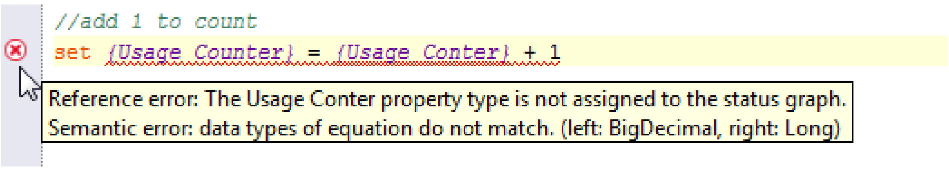


```
//add 1 to count
set {Usage Counter} =
```

Figure 40: Expression panel with comment and auto-completion

The editor validates your input as you type and marks errors with a wavy, red underline as soon as they occur. Additionally, the editor displays an error icon to the left of the affected line.
Another indicator is the global error status marker in the top-right corner of the expression panel that turns red and expands downward to mark individual error positions if the expression contains errors.

To see a more explicit error message, hover your mouse cursor over any of the error markers. The editor will then display the error message in a tooltip.



```
//add 1 to count
set {Usage Counter} = {Usage Conter} + 1
```

Reference error: The Usage Conter property type is not assigned to the status graph.
Semantic error: data types of equation do not match. (left: BigDecimal, right: Long)

Figure 41: Expression panel with errors and message tooltip

Constants

Constants are differentiated by their data types:

- Long
used for integral numbers:
12345

- Float
floating point number, can be used for fractional numbers, but needs a specific markup (**Float**) to be interpreted correctly by the system:
123.45Float

- BigDecimal
floating point number that allows calculating with greater precision than Float. By default, the system interprets fractional numbers as BigDecimal, so a markup (**BDecimal**) is only necessary when you use an integral number but want it to be treated as BigDecimal:
123.45 or **123BDecimal** or **12.3e42** (where **e** represents the exponential notation and stands for "times ten raised to the power of").

> **TIP**
> BigDecimal values support up to 15 integral digits and are rounded to 9 fractional digits when stored to the database.

- Boolean
with the values **true** and **false**.

- String
  used for displaying any sequence of characters, such as
  "**Read instruction text:**"

  TIP
  Please note that a string constant must be enclosed in quotes.

- Timestamp
  used for displaying dates and times and for time-related calculations.
  Its full format is **dd-MMM-yyyy@H:M:s.S**, but you can leave out the either entire time section, or the seconds/milliseconds section, or just the milliseconds:
  - 11-Jan-2013@12:13:54.232
  - 3-Mar-2015
  - 02-Sep-2013@12:13
  - 29-May-2012@12:13:17

  TIP
  Please note that timestamp constants always refer to UTC time (Coordinated Universal Time, which is the equivalent to GMT), to make timestamp constants independent of time zones.

- Duration
  used for displaying time spans and for time-related calculations.
  It supports the following time units: days (**d**), hours (**h**), minutes (**min**), seconds (**s**), and milliseconds (**ms**).
  The following specifics apply to using durations in your expressions:
  - Make sure to write the duration as consecutive string of characters without blanks, since otherwise, the system will not be able to interpret it correctly.
  - You can leave out any units you do not need for specifying your duration, so instead of **0d6h0min30s0ms** you can write **6h30s**.
  - You have to observe the order of the units, which means, for instance, that the system cannot interpret a duration that starts with seconds, followed by minutes and hours.
  - The values you specify for the various time units must not exceed the maximum number of each time unit that would move it to the next higher unit. Thus, for hours (**h**) the maximum is 23, for minutes (**min**) and seconds **(s)** it is 59, and for milliseconds (**ms**) it is 999.
    This restriction, however, does not apply to the highest unit you specify, which means that the system allows a duration such as **49h30min15s**, whereas typing **48h89min75s** in this context would not be valid.

- MeasuredValue
  used for displaying numeric values qualified by a unit of measure.

> TIP
> MeasuredValue values support up to 15 integral digits and are rounded to 9 fractional digits when stored to the database.

The following specifics apply to using measured values in your expressions:

- Make sure to write a measured value as string of characters without blanks as in
  280kg or **29gal**
- If a unit of measure itself contains a blank, you have to replace it with an underscore "_" as in
  125fl_oz
- Measured values that contain special characters, such as °C or IU/ml need to use the **Convert to Measured Value** function to be interpreted correctly by the system.
- You cannot use the MeasuredValue data type for time-related constants, use the Duration data type instead.
- For calculations that have a measured value as result, the system returns the result in the unit of measure of the operand that is expected to provide greater accuracy, either because of the unit's finer granularity or due to a greater precision on account of the number of fractional digits:
  1kg + 100g = 1100g
  1.000kg + 100g = 1.100kg

The precision of plus and minus operations between measured values depends on the number of fractional digits given with the operand that has the unit of measure with the greater accuracy. Thus, the result of **1kg + 100.00g = 1100.00g** whereas **1.00kg + 100g = 1100g**.

- The precision of division operations between a measured value and a number or another measured value is established by adding the second operand's (divisor) base-10 logarithm to the number of fractional digits of the first operand (dividend):
  <number of fractional digits of dividend> + log(<divisor>).
  This means that the system calculates **3 + log(100) = 5** to determine the number of fractional digits for **10.000g / 100**, and consequently returns **0.10000g**.
  The same calculation is used to establish the number of fractional digits in **10.000g / 100g = 0.10000**.

- Similarly, the precision of multiplication operations between a measured value and a number is established by subtracting the number's base-10 logarithm from the number of fractional digits of the measured value: **<number of fractional digits of measured value> + log(<number>)**. So the system calculates **3 - log(100) = 1** to determine the number of fractional digits for **10.000g * 100 and returns 1000.0g**.

TIP

Please note that the system generally rounds logarithms up, so that **log(100) = 2**, whereas **log(101) = 3**.

- The calculation sequence of operators with the same precedence (page 42) is strictly left to right in pairs. Thus **a+b+c+d** is interpreted as **((a+b)+c)+d**.

TIP

It is generally recommended to avoid mixing metric and US customary units of measure in calculations, since the results of these calculation depend on the units of measure, the precision of the individual operands, and their sequence.

Specific to these types of implicit conversion calculations is that after the system has determined the result unit of measure (that of the operand with the greater accuracy) it converts the other operand to match the result unit of measure. For converting, it uses the system-defined conversion relation. After having converted the operand, the system rounds the converted operand to the same precision it had in its initial unit of measure.

Mixed unit examples:

1kg +1000g + 1000000mg + 1.000lb is interpreted as
((1kg +1000g) + 1000000mg) + 1.000lb and returns **3453600mg**.

- In this case, the first two calculations increase the precision to milligrams and return **3000000mg**.
- The third calculation thus is **3000000mg + 1.000lb** and the system performs the following steps:
  1. It determines the result unit of measure:
     mg (has the finer granularity).
  2. It performs the conversion, using the system-defined relation **1lb = 0.453592370kg**. This means a twofold conversion needs to take place: **lb** » **kg** and **kg** » **mg** (conversion relation **1mg = 0.000001000kg**)
     1.000 * 0.453592370 / 0.000001000 = 453592.370000000000, which is then rounded to **453600** to have the same (4-digit) precision as the **lb** value. So, the result of the conversion is **453600mg**.

3. It performs the calculation:
   3000000mg + 453600mg = 3453600mg
   No further changes to the precision are required.

1.000lb + 1kg +1000g + 1000000mg is interpreted as
((1.000lb + 1kg) +1000g) + 1000000mg and returns **3360900mg**.

- In this case, each of the three calculations also involves a conversion.
  The first calculation is **1.000lb + 1kg** and the system runs the three steps:
  1. Result unit of measure: **lb**
  2. Conversion: **lb** » **kg** with conversion relation: **1lb = 0.453592370kg**
     1 * 1.000000000 / 0.453592370 = 2.204622622, rounded to **2.** So, the result of the conversion is **2lb**.
  3. Calculation:
     1.000lb + 2lb = 3.000lb
     The result has a precision of three fractional digits.
- The second calculation is **3.000lb +1000g**:
  1. Result unit of measure: **lb**
  2. Conversion: **g** » **kg** and **kg** » **lb** with conversion relations
     1g = 0.000010000kg and **1lb = 0.453592370kg**
     1000 * 0.001000000 / 0.453592370 = 2.204622622, rounded to **2.205**. So, the result of the conversion is **2.205lb**.
  3. Calculation:
     3.000lb + 2.205lb = 5.205lb
     The result has a precision of three fractional digits.
- The third calculation is **5.205lb + 1000000mg**:
  1. Result unit of measure: **mg**
  2. Conversion: twofold, **lb** » **kg** and **kg** » **mg** with conversion relations
     1mg = 0.000001000kg 1lb = 0.453592370kg and **1mg = 0.000001000kg**
     5.205 * 0.453592370 / 0.000001000 = 2360948.285850000000, rounded to **2360900**. So, the result of the conversion is **2360900mg**.
  3. Calculation:
     2360900mg + 1000000mg = 3360900mg
     No further changes to the precision are required.

Operators

Operators apply to operands, like in **A + B**, where **A** and **B** are the operands to which the + operator applies.
A **unary** operator applies to only one operand, whereas a **binary** operator applies to two operands.

When it comes to evaluating expressions, it is important to be aware of the precedence rule that applies to the operations and defines the order in which the operations of an expression are performed. The following list shows the order of operations, starting with the highest-precedence operation:

1. unary plus (+) and minus (-)
2. multiply (*) and divide (/)
3. binary plus (+) and minus (-)
4. all comparison operators:
   - equal to (==)
   - not equal to (!=)
   - less than (<)
   - less than or equal to (<=)
   - greater than (>)
   - greater than or equal to (>=)
5. logical NOT
6. logical AND
7. logical OR

TIP

When your expression contains a sequence of operations that have the same precedence, it is recommended to use parentheses to disambiguate the order in which they are processed.

- To insert an operator at the current cursor position in the expression panel (page 32), double-click its node in the tree.

Arithmetic Operators

Arithmetic operators allow you to perform calculations between numeric values.

- Plus (+)
  binary, is used for additions or string concatenations:
  (**{Check ph value 01}.{Instance count}** + **{Check ph value 02}.{Instance count}**) <= 4
  "Date of execution: " + convertToDisplayString(now())

  Operations with binary plus for time-related operands, measured values, and strings and their results:

Left Operand	Right Operand	Operation Result
Duration	Duration	Duration
Timestamp	Duration	Timestamp
MeasuredValue	MeasuredValue	MeasuredValue (can return **Undefined**)
String	String	String

TIP
Please note that an operation with incompatible operands, such as lengths (e.g. meters) and weights (e.g. pounds) returns **Undefined** as result.

- Plus (+)
  unary, is rarely used, since numbers without a sign are assumed to be positive.

  Operations with unary plus for time-related operands and/or measured values and their results:

Left Operand	Right Operand	Operation Result
N/A	Duration	Duration
N/A	MeasuredValue	MeasuredValue

- Minus (-)
  binary, is used for subtractions:
  {Adjust ph value}.{Instance count} - {Check ph value 01}.{Instance count} >= -1

Operations with binary minus for time-related operands and/or measured values and their results:

Left Operand	Right Operand	Operation Result
Duration	Duration	Duration
Timestamp	Duration	Timestamp
Timestamp	Timestamp	Duration
MeasuredValue	MeasuredValue	MeasuredValue (can return **Undefined**)

TIP
Please note that an operation with incompatible operands, such as lengths (e.g. meters) and weights (e.g. pounds) returns **Undefined** as result.

- Minus (-)
  unary, is used for negations:
  {Adjust ph value}.{instance count} **-** {Check ph value 01}.{instance count} >= **-1**

  Operations with unary minus for time-related operands and/or measured values and their results:

Left Operand	Right Operand	Operation Result
N/A	Duration	Duration
N/A	MeasuredValue	MeasuredValue

- Multiply (*)
  binary, is used for multiplications:
  {Adjust ph value}.{Instance count} * 2 <= {Check ph value 01}.{Instance count}

  Multiplication operations with time-related operands and/or measured values and their results:

Left Operand	Right Operand	Operation Result
Number (Long, Float, BigDecimal)	Duration	Duration
Duration	Number (Long, Float, BigDecimal)	Duration
MeasuredValue	Number (Long, Float, BigDecimal)	MeasuredValue
Number (Long, Float, BigDecimal)	MeasuredValue	MeasuredValue

TIP
Please note that you can turn a Number data type into a MeasuredValue by multiplying the numeric value with 1<unit of measure>.
Example:
20 * 1kg = 20kg

- Divide (/)
  binary, used for divisions:
  {Adjust ph value}.{Instance count} / {Check ph value 01}.{Instance count} > 1

TIP
Please note that when the right operand of your division is a reference that returns **0** or **undefined**, the result of the division is also **undefined**.

Division operations with time-related operands and/or measured values and their results:

Left Operand	Right Operand	Operation Result
Duration	Duration	BigDecimal
Duration	Number (Long, Float, BigDecimal)	Duration
MeasuredValue	MeasuredValue	BigDecimal (can return **Undefined**)
MeasuredValue	Number (Long, Float, BigDecimal)	MeasuredValue

TIP
Please note that an operation with incompatible operands, such as lengths (e.g. meters) and weights (e.g. pounds) returns **Undefined** as result.

Comparison Operators

Comparison operators are necessarily all binary since they compare two operands.

TIP
Please note that if one or both of the operands are **undefined**, the result of the comparison itself is **undefined**.

- Equal to (==)
  {Read instruction}.{out} == OK

- Not equal to (!=)
  {Read instruction}.{out} != OK

- Less than (<)
  {Adjust ph value}.{Instance count} < 3

- Less than or equal to (<=)
  {Check ph value}.{value} <= 5.5

- Greater than (>)
  {Adjust ph value}.{Instance count} > 3

- Greater than or equal to (>=)
  {Check ph value}.{value} >= 5.5

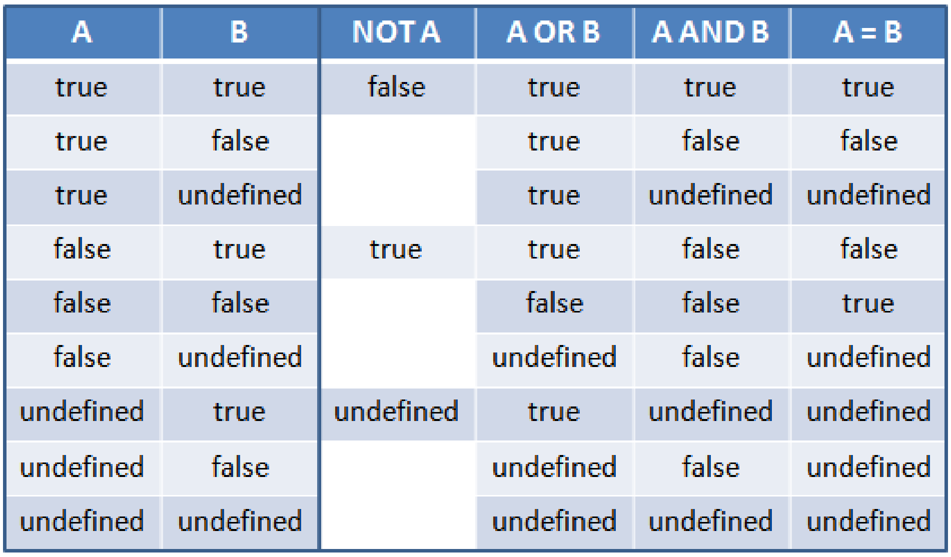
Logical Operators

Logical operators always return **true**, **false**, or **undefined** as result. **Undefined** (null) can happen, for instance, when one of the operands of the logical operation is **undefined**.

TIP

Please note the order of precedence that applies to logical operations, with NOT preceding AND and OR, which has the lowest precedence.
This means that **NOT A OR B** is interpreted as **(NOT A) OR B**.

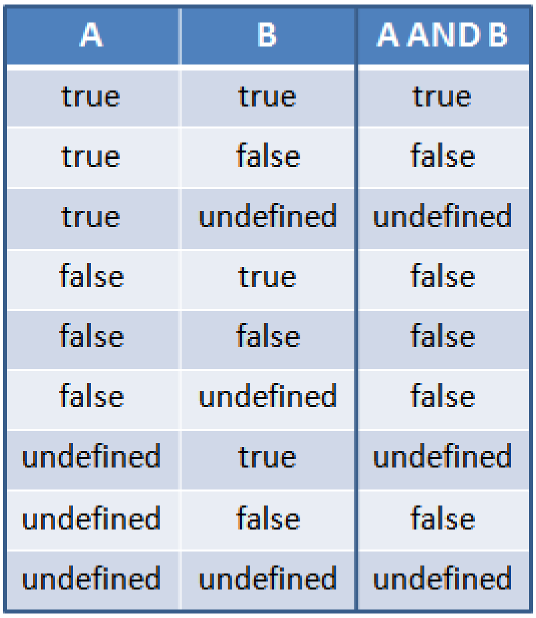
The results of logical operations correspond to the three-valued logic used by SQL.



A	B	NOT A	A OR B	A AND B	A = B
true	true	false	true	true	true
true	false		true	false	false
true	undefined		true	undefined	undefined
false	true	true	true	false	false
false	false		false	false	true
false	undefined		undefined	false	undefined
undefined	true	undefined	true	undefined	undefined
undefined	false		undefined	false	undefined
undefined	undefined		undefined	undefined	undefined

Figure 42: Three-valued truth table

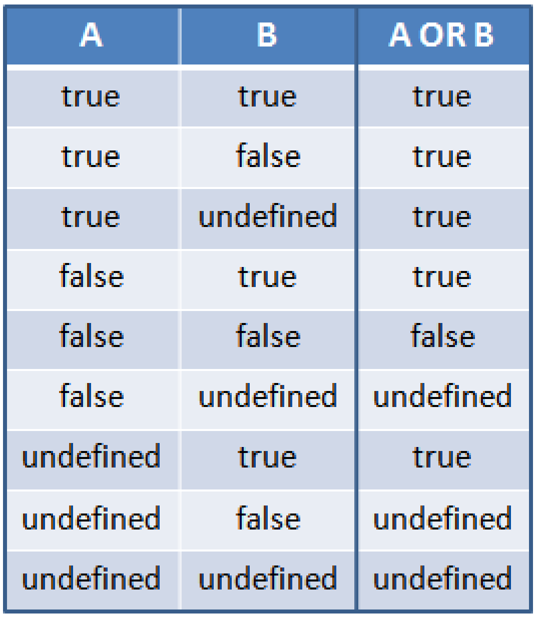
- Conjunction (AND)
  binary, an expression **A AND B** returns true only if both A and B are true.



A	B	A AND B
true	true	true
true	false	false
true	undefined	undefined
false	true	false
false	false	false
false	undefined	false
undefined	true	undefined
undefined	false	false
undefined	undefined	undefined

Figure 43: Truth table - A AND B

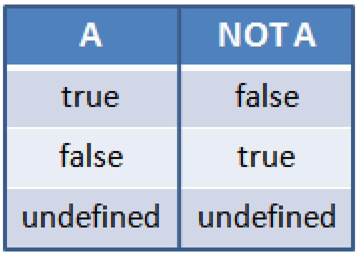
- Disjunction (OR)
  binary; an expression **A OR B** returns true if either A or B or both A and B are true.



A	B	A OR B
true	true	true
true	false	true
true	undefined	true
false	true	true
false	false	false
false	undefined	undefined
undefined	true	true
undefined	false	undefined
undefined	undefined	undefined

Figure 44: Truth table - A OR B

- Negation (NOT)
  unary, an expression **NOT A** returns true for any value (not undefined) other than A.



A	NOT A
true	false
false	true
undefined	undefined

Figure 45: Truth table - NOT A

Functions

The Expression editor provides a number of functions whose implementation with the available operators would either lead to very complex expressions or would not be possible at all.
A function consists of its name typically followed by one or more comma-separated arguments in parentheses:
function_name(arg1, arg2, ...)

TIP

Please note that you can also use references as arguments. This may, however, cause an argument to be **undefined**.

- To insert a function with sample arguments at the current cursor position in the expression panel (page 32), double-click its node in the tree.
- To view a function with sample arguments, hover your mouse cursor over the function. The system displays the entire function as tooltip.

Null Handling Functions

Null handling functions provide the means for dealing with phases that return **undefined** as output.

- Undefined - one of two (nvl)
  inserted as **nvl(arg1, arg2)**
  It has the following characteristics:
  - It expects exactly two arguments of the same data type.
  - It evaluates the arguments from left to right and returns the first argument that is not **undefined** as result.
  - If all arguments are **undefined**, its result is also **undefined**.

- Undefined - one of many (coalesce)
  inserted as **coalesce(arg1, arg2, ...)**
  It has the following characteristics:
  - It expects at least two arguments of the same data type.
  - It evaluates the arguments from left to right and returns the first argument that is not **undefined** as result.
  - If all arguments are **undefined**, its result is also **undefined**.
- Is undefined (isNull)
  inserted as **isNull(arg1)**
  It has the following characteristics:
  - It expects exactly one argument of any data type.
  - It returns **true** as result if the argument is **undefined**. In all other cases it returns **false** as result.

Analysis Functions

Analysis functions are a collection of functions for retrieving and inserting external information, such as a timestamp, or for parsing external information to return a result.

- Current Date and Time (now)
  inserted as **now()**
  It has the following characteristics:
  - It does not expect any argument.
  - At evaluation time, i.e. during execution, it returns a Timestamp data type, which provides the then current time of the database server.
- Contains Item (containsItem)
  inserted as **containsItem(arg1, arg2 [, arg3])**
  It has the following characteristics:
  - It expects at least two arguments of the String data type. The third argument is optional.
  - It parses a separator-delimited list (**arg1**) for occurrences of a specified list item (**arg2**).
  - By default, it uses semicolon (;) as separator character. With **arg3**, however, you can declare another separator character.
  - If it locates an occurrence of **arg2**, it returns **true**, if not, it returns **false**.
  - If either of the three arguments is **undefined**, its result is also **undefined**.

> **TIP**
> This function is especially suited to deal with lists of tags returned from automated equipment. For these purposes, you would use the full list of potential return values of a FlexibleTagDefinition property as **arg1** and the identifier of the tag you wish to monitor as **arg2**.
> Examples:
> containsItem("AlarmHighTemp;AlarmLowPressure;AlarmEmergencyShutOff", "AlarmLowPressure") returns **true**
> containsItem("AlarmHighTemp:AlarmLowPressure:AlarmEmergencyShutOff", "AlarmPressureLimit", ":") returns **false**

Conversion Functions

Conversion functions are a collection of functions that allow to convert the data type of a value to another data type.

- Convert to Unitless Number (convertTo)
  inserted as **convertTo(arg1, arg2)**
  It has the following characteristics:
  - It expects a measured value or a duration as first argument and as second argument a suitable, convertible unit of measure given as string. It returns a BigDecimal data type.

    > **TIP**
    > Please note that some phases may return measured values without a unit of measure. Even though these values look like a BigDecimal, their data type is still MeasuredValue, which you need to convert to BigDecimal in order to allow calculations. In this case, you have to provide an empty string as second argument.

    Examples:
    - convertTo(15m, "m") returns 15
    - convertTo(1h5min,"s") = 3900
    - convertTo(12min,"h") = 0.2
    - convertTo({<phase identifier>}.{<output>}, "") returns the unitless value from the phase output as BigDecimal value.
  - If the unit of measure defined for the second argument does not exist, the result is **undefined**.
  - If one of the arguments is **undefined**, its result is also **undefined**.
- Convert to BigDecimal (convertToBigDecimal)
  inserted as **convertToBigDecimal(arg1)**
  It has the following characteristics:

  - It expects an argument of a numeric data type, such as a Long or Float and converts it to a BigDecimal data type.
    Examples:
    - convertToBigDecimal(12345) returns 12345BDecimal
    - convertToBigDecimal(123.45Float) returns 123.45
  - If the argument is **undefined**, its result is also **undefined**.

- Convert to Long (convertToLong)
  inserted as **convertToLong(arg1, arg2)**
  It has the following characteristics:
  - It expects a fractional number, typically a BigDecimal, as first argument and as second argument a rounding mode, given as string.
  - The following rounding modes are available:
    - HALF_UP:
      rounds towards the nearest integral number, unless there are two integrals at exactly the same distance. In this case, it rounds away from zero.
    - CEILING:
      always rounds up.
    - FLOOR:
      always rounds down.

    Examples:
    - convertToLong(2.5, "HALF_UP") returns 3
    - convertToLong(-1.6, "HALF_UP") returns -2
    - convertToLong(1.3, "CEILING") returns 2
    - convertToLong(-1.7, "CEILING") returns -1
    - convertToLong(2.6, "FLOOR") returns 2
    - convertToLong(-1.1, "FLOOR") returns -2
  - If either or both arguments are **undefined**, its result is also **undefined**.

- Convert to MeasuredValue (convertToMeasuredValue)
  inserted as **convertToMeasuredValue(arg1 [, arg2])**
  It has the following characteristics:
  - It expects at least one argument of a numeric data type (Long, Float, or BigDecimal) as first argument and as second argument a unit of measure given as string. The second argument is optional.

- It converts the numeric value into a MeasuredValue data type with the unit of measure indicated as **arg2**. If you do not define the second argument, the system still converts the numeric value, so that it returns MeasuredValue without unit of measure.

> TIP
> Please note that you need to use the function when your expression contains measured values whose unit includes a special character, such as °C or IU/ml. Otherwise the system cannot interpret the measured value correctly.

  Examples:

  - convertToMeasuredValue(15, "m") returns 15 m
  - convertToMeasuredValue(34.8) returns 34.8
  - convertToMeasuredValue(22, "°C") returns 22 °C

- If the unit of measure defined for the second argument does not exist, the result is **undefined**.
- If the numeric value argument (**arg1**) is **undefined**, its result is also **undefined**.

Aggregation Functions

Aggregation functions provide the means to evaluate two and more output values to determine the highest, lowest, or the average of all values.

- Average (average)
  inserted as **average(arg1, arg2, ...)**
  It has the following characteristics:
  - It expects at least two arguments of the same numeric data type.
  - It ignores arguments that are **undefined**.
  - It calculates the average of all usable (not **undefined**) arguments.
  - If only one argument is usable, its result equals the argument.
  - If all arguments are **undefined**, its result is also **undefined**.

> TIP
> When you use the **Average** function with arguments of the MeasuredValue data type, the system returns the result in the unit of measure of the first operand.

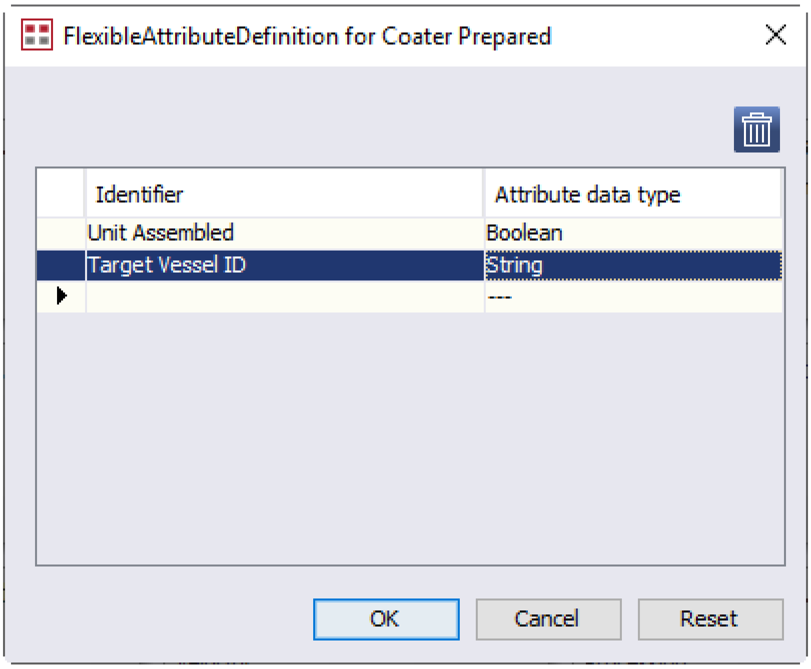
- Maximum (max)
  inserted as **max(arg1, arg2, ...)**
  It has the following characteristics:
  - It expects at least two arguments of the same, comparable data type.
  - It ignores arguments that are **undefined**.

  - It determines the largest of all usable (not **undefined**) arguments.
  - If only one argument is usable, its result equals the argument.
  - If all arguments are **undefined**, its result is also **undefined**.

- Minimum (min)
  inserted as **min(arg1, arg2, ...)**
  It has the following characteristics:
  - It expects at least two arguments of the same, comparable data type.
  - It ignores arguments that are **undefined**.
  - It determines the smallest of all usable (not **undefined**) arguments.
  - If only one argument is usable, its result equals the argument.
  - If all arguments are **undefined**, its result is also **undefined**.

FlexibleAttributeDefinition Editor

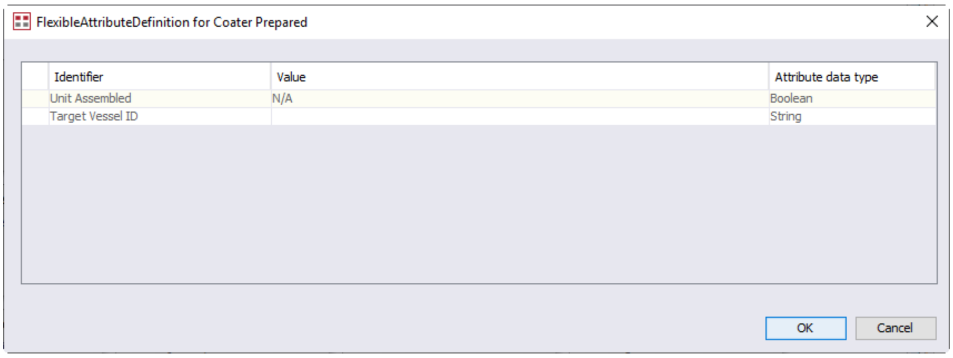
The FlexibleAttributeDefinition editor supports you with creating and configuring a bundle of runtime attributes.



FlexibleAttributeDefinition for Coater Prepared

Identifier	Attribute data type
Unit Assembled	Boolean
Target Vessel ID	String

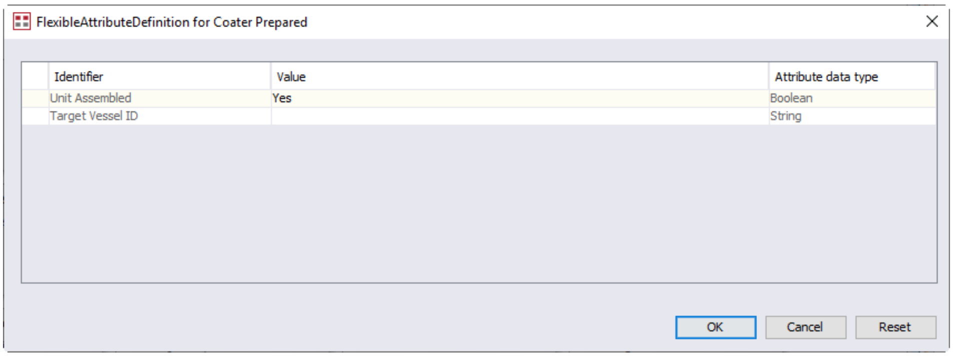
Figure 46: FlexibleAttributeDefinition editor for property type



FlexibleAttributeDefinition for Coater Prepared

Identifier	Value	Attribute data type
Unit Assembled	N/A	Boolean
Target Vessel ID		String

Figure 47: FlexibleAttributeDefinition editor for property of equipment class



FlexibleAttributeDefinition for Coater Prepared

Identifier	Value	Attribute data type
Unit Assembled	Yes	Boolean
Target Vessel ID		String

Figure 48: FlexibleAttributeDefinition editor for property of equipment entity

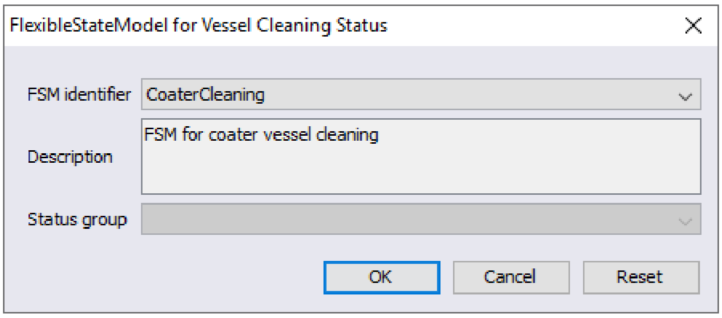
- When you define a property type, you need to create the list of runtime attributes you wish to group. For each property you have to specify an identifier and set its attribute data type (**Boolean**, **DateTime**, **Duration**, **Long**, **MeasuredValue**, **String**).

> TIP
> Please note that the identifier you define is displayed later during processing to the operator on the shop floor. So, it makes sense to create a human-readable, non-cryptic identifier.

- When you define a property for an equipment class, you can only view the list of attributes defined with the property type.
- When you define a property for an equipment entity, you can set values for the individual attributes.
  The identifiers and attribute data types are specified with the property type and cannot be changed.
- To save your data and close the FlexibleAttributeDefinition editor, click the **OK** button.
- To close the FlexibleAttributeDefinition editor without saving your changes, click the **Cancel** button.
- To reset your data to its last saved form, click the **Reset** button.

FlexibleStateModel Editor

The FlexibleStateModel editor supports you with defining a status graph to govern an equipment entity and with setting the required status the equipment entity must have to be suitable for use.



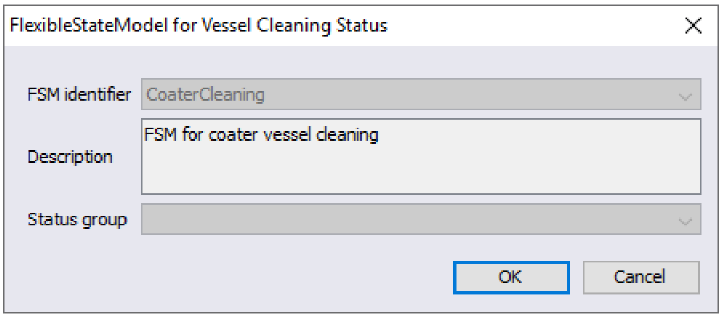
FlexibleStateModel for Vessel Cleaning Status

FSM identifier: CoaterCleaning

Description: FSM for coater vessel cleaning

Status group:

Figure 49: FlexibleStateModel editor for property type



FlexibleStateModel for Vessel Cleaning Status

FSM identifier: CoaterCleaning

Description: FSM for coater vessel cleaning

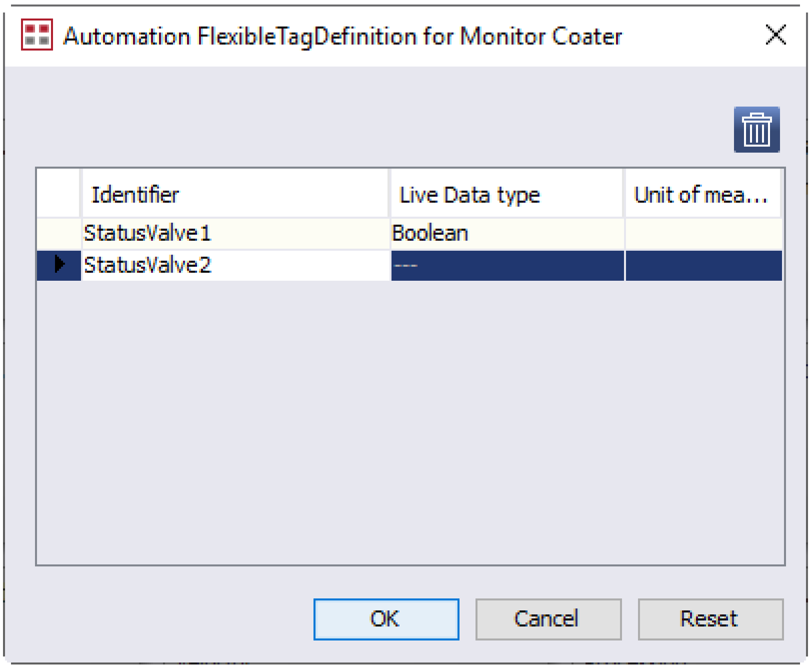
Status group:

Figure 50: FlexibleStateModel editor for property

- When you define a property type you can only select the FlexibleStateModel and thus the status graph with its semantic status groups.
- When you assign the property type to a class or entity, you cannot make any further definitions on the property. Selecting a status group is only available when you turn the property into a requirement by using it during recipe or workflow design.
- To save your data and close the FlexibleStateModel editor, click the **OK** button.
- To close the FlexibleStateModel editor without saving your changes, click the **Cancel** button.
- To reset the FlexibleStateModel definition to its last saved form, click the **Reset** button.

FlexibleTagDefinition Editor for Automation

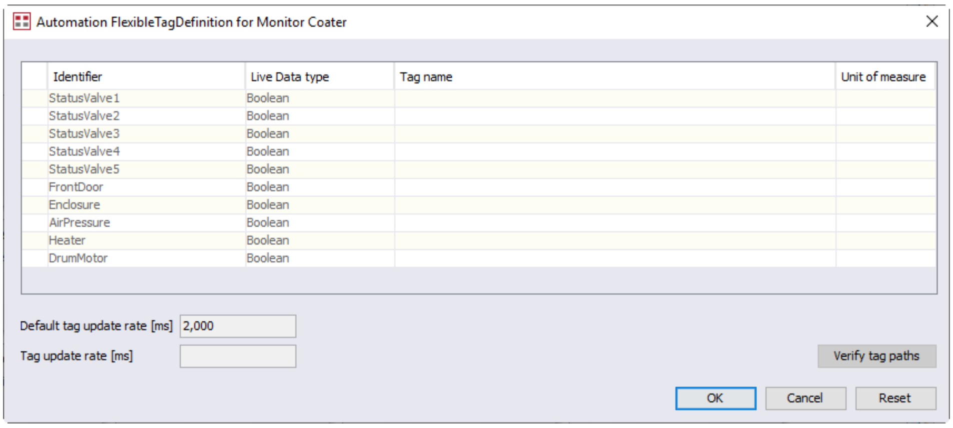
The FlexibleTagDefinition editor supports you with creating and configuring a bundle of automation properties mapped to individual tags of an equipment entity.



Automation FlexibleTagDefinition for Monitor Coater

	Identifier	Live Data type	Unit of mea...
	StatusValve1	Boolean	
▶	StatusValve2	---	

Figure 51: Automation FlexibleTagDefinition editor for property type



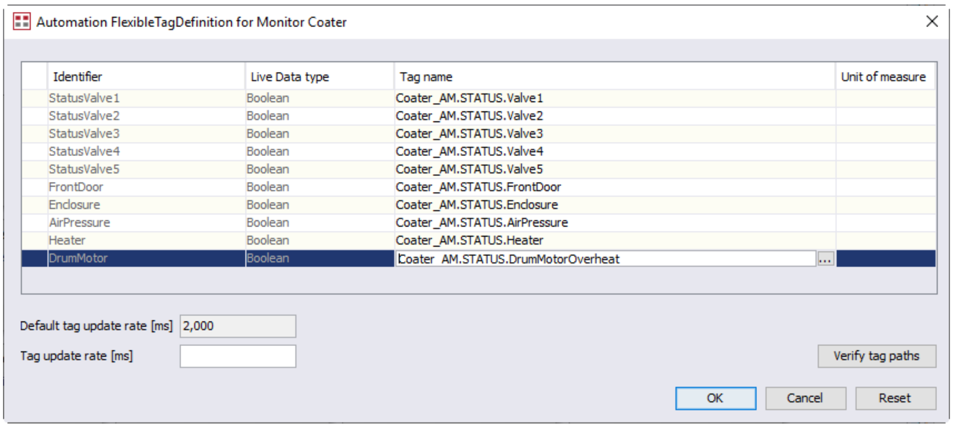
Automation FlexibleTagDefinition for Monitor Coater

Identifier	Live Data type	Tag name	Unit of measure
StatusValve1	Boolean		
StatusValve2	Boolean		
StatusValve3	Boolean		
StatusValve4	Boolean		
StatusValve5	Boolean		
FrontDoor	Boolean		
Enclosure	Boolean		
AirPressure	Boolean		
Heater	Boolean		
DrumMotor	Boolean		

Default tag update rate [ms] 2,000

Tag update rate [ms]

Figure 52: Automation FlexibleTagDefinition editor for property of equipment class



Identifier	Live Data type	Tag name	Unit of measure
StatusValve1	Boolean	Coater_AM.STATUS.Valve1	
StatusValve2	Boolean	Coater_AM.STATUS.Valve2	
StatusValve3	Boolean	Coater_AM.STATUS.Valve3	
StatusValve4	Boolean	Coater_AM.STATUS.Valve4	
StatusValve5	Boolean	Coater_AM.STATUS.Valve5	
FrontDoor	Boolean	Coater_AM.STATUS.FrontDoor	
Enclosure	Boolean	Coater_AM.STATUS.Enclosure	
AirPressure	Boolean	Coater_AM.STATUS.AirPressure	
Heater	Boolean	Coater_AM.STATUS.Heater	
DrumMotor	Boolean	Coater AM.STATUS.DrumMotorOverheat ...	

Default tag update rate [ms] 2,000

Tag update rate [ms]

Figure 53: Automation FlexibleTagDefinition editor for property of equipment entity

- When you define a property type, you need to create the list of properties you wish to monitor on the equipment. For each property you have to specify an identifier, set its Live Data type (**Boolean**, **Double**, **Float**, **Integer**, **String**), and select a unit of measure, if necessary. If there is no suitable unit available in the option list, you can leave the cell blank.

TIP
Please note that the identifier you define is displayed later during processing to the operator on the shop floor. So, it makes sense to create a human-readable, non-cryptic identifier.

- When you define a property for an equipment class, you can only view the list of properties defined with the property type.
- When you define a property for an equipment entity, you need to specify to which tags of your equipment you wish to map the properties defined in your list and with which update rate the system is to access the tags.

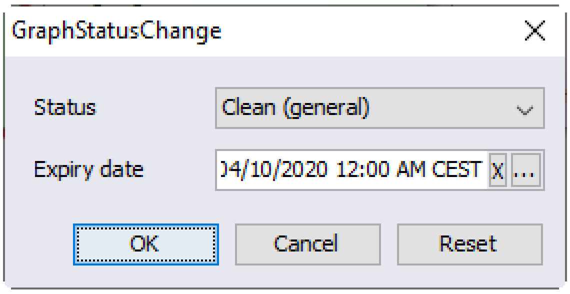
TIP
To see the list of tags available at your equipment and to select the suitable tag, click the **...** button to open the Tag Browser editor (page 70).

To check if the indicated tag names correspond to existing tag names on the equipment entity, click the **Verify tag paths** button.
The identifiers, Live Data types, and units of measure are specified with the property type and cannot be changed.

- To save your value and close the FlexibleTagDefinition editor, click the **OK** button.
- To close the FlexibleTagDefinition editor without saving your changes, click the **Cancel** button.
- To reset your data to its last saved form, click the **Reset** button.

GraphStatusChange Editor

The GraphStatusChange editor supports you with changing the current status of an entity along with the expiry date of the status, if it can expire.



GraphStatusChange

Status: Clean (general)

Expiry date: 04/10/2020 12:00 AM CEST

Figure 54: GraphStatusChange editor for graph of entity

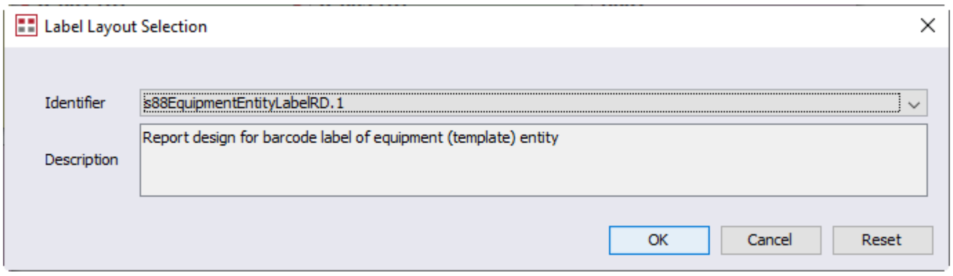
- To save the status changes and close the GraphStatusChange editor, click the **OK** button.
- To close the GraphStatusChange editor without saving the status changes, click the **Cancel** button.
- To reset the status to its last saved form, click the **Reset** button.

Label Layout Selection Editor

The Label Layout Selection editor supports you with selecting the layout for the labels of entities.

TIP

Please note that the available label layouts are maintained outside of PharmaSuite as Report Design objects in FactoryTalk ProductionCentre Process Designer.



Label Layout Selection

Identifier: s88EquipmentEntityLabelRD.1

Description: Report design for barcode label of equipment (template) entity

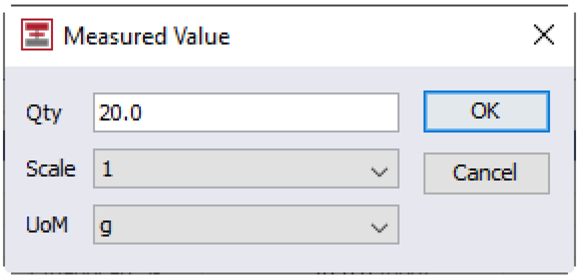
Figure 55: Label Layout Selection editor

- To save your selection and close the Label Layout Selection editor, click the **OK** button.
- To close the Label Layout Selection editor without saving your changes, click the **Cancel** button.

- To reset your selection to its last saved form, click the **Reset** button.

Measured Value Editor

The Measured Value editor supports you when you enter a value with a unit of measure. **Scale** indicates the number of fractional digits.



Qty 20.0

Scale 1

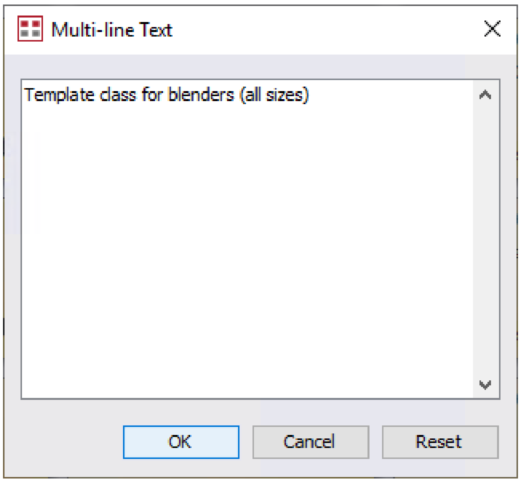
UoM g

Figure 56: Measured Value editor

- To save your value and close the Measured Value editor, click the **OK** button.
- To close the Measured Value editor without saving your changes, click the **Cancel** button.

Multi-line Text Editor

The Multi-line Text editor supports you when you need to write text that is too long for a simple input box.



Template class for blenders (all sizes)

Figure 57: Multi-line Text editor

- To save your text and close the Multi-line Text editor, click the **OK** button.
- To close the Multi-line Text editor without saving your changes, click the **Cancel** button.
- To reset your text to its last saved form, click the **Reset** button.

Ranges Editor

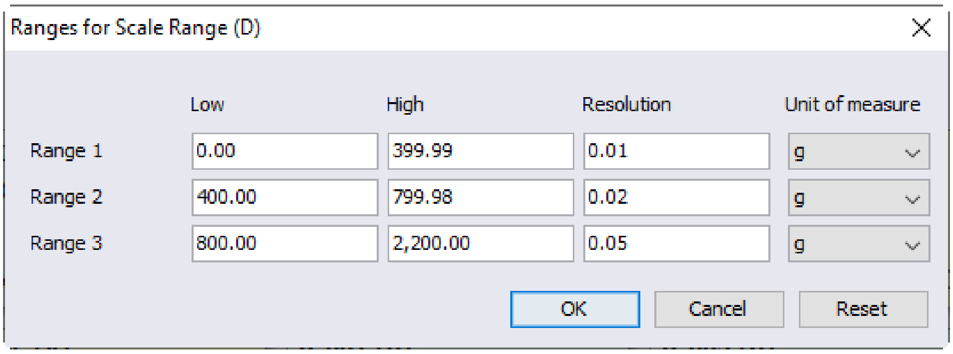
The Ranges editor supports you with defining the ranges of your equipment. You can define up to three ranges per equipment by specifying the high and low limits, the resolution, and the unit of measure of each range.

When used with a scale equipment entity during execution, the system evaluates the ranges defined for the connected scales and proposes the scale and range that is best suited to weigh the target quantity.

TIP

Please note the following restrictions when defining scale ranges:

- Ranges must be defined to mirror the range features of the physical scale equipment and must not overlap.
- Ranges must be defined in order: lowest range first, then next higher range, then highest range.
- Resolutions must differ for different ranges: fine resolution for lower range, coarser resolutions for higher ranges.
- The system expects the same unit of measure for all ranges.



Ranges for Scale Range (D)

	Low	High	Resolution	Unit of measure
Range 1	0.00	399.99	0.01	g
Range 2	400.00	799.98	0.02	g
Range 3	800.00	2,200.00	0.05	g

Figure 58: ScaleRanges editor for property

- When you define a property for an equipment class, you can set all values. The system checks that entities of the class match the defined values.

  TIP

  If you set attribute values to **Undefined**, for example by leaving an input box blank, the entities assigned to the class can have any value.

- When you define a property for an equipment entity, you can set all values and thus provide the information necessary for using the entity during execution.
- To save your data and close the Ranges editor, click the **OK** button.
- To close the Ranges editor without saving your changes, click the **Cancel** button.
- To reset your data to its last saved form, click the **Reset** button.

ScaleConfiguration Editor

The ScaleConfiguration editor supports you with entering scale driver and scale connection data.

- Manual scale
  Select this option if the scale is always operated manually and never connected to the system.

  TIP
  Please note that the only data you need to configure for a manual scale is its **Resolution factor** setting (page 62). All other settings or options are ignored by the system.

- Scale driver
  Select the scale driver to be used.

  TIP
  Make sure to select a driver that supports all driver features you need to select.

- COM port
  The COM port number.

  TIP
  If the scale connects to the system by Ethernet (TCP/IP protocol), the COM port must be left blank.

- TCP server / port
  The TCP server and port number of the device that connects the scale to the system. This can either be the scale itself or the terminal to which the scale connects.

  TIP
  If the scale connects to the system by COM port, the TCP server and port must be left blank.

- Terminal number
  If the scale connects directly to the computer at the work center, the setting typically is 0. If the work center setup includes a terminal that connects to several scales, the terminal number reflects the scale number within the terminal.

- Resolution factor
  Must be greater than 0 and less than 1000 and has a default value of 10. The factor is used in **Release scale** phases of Weighing or Dispense operations to calculate the tolerance around 0 that determines if a release scale check passes or fails. The factor is multiplied with the resolution defined for the lowest range of the scale that is about to be released.
  Example:
  During the **Weigh** phase of a Dispense operation the scale range used has a resolution of 0.02, by definition in the **Range** property (page 60) of the scale. The resolution factor is 5.
  When executing the release scale check of the subsequent **Release scale** phase, the system calculates 0.02 * 5= 0.1 and derives a tolerance range of -0.1 to 0.1 from the result. If the scale value received from the scale is within this range, the release check passes; if not, it fails.

- Zoeroing

  Select this option if the scale can zero.

  TIP
  This feature is required to zero a scale after the **Select scale** phase of a Weighing ode Dispense operation has completed.

- Manual tare
  Select this option if the scale supports manual taring, which means that a tare value can be sent to the scale.

  TIP
  This feature is required for the **Gross** and **Pallet** weighing methods of a Weighing or Dispense operation.

- Tare
  Select this option if the scale can process tare commands.

  TIP
  This feature is required for the **Net** and **Net removal** weighing methods of a Weighing or Dispense operation.

- Weighing
  Select this option if the scale can process weigh commands. This option controls, for example, if the scale control continuously updates the displayed scale value.

  TIP
  This feature is required for all weighing methods of a Weighing or Dispense operation.

- Get serial number
  Select this option if the scale can be identified by its serial number when it is accessed.

  TIP
  Please note that the product phases of PharmaSuite do not support this feature.

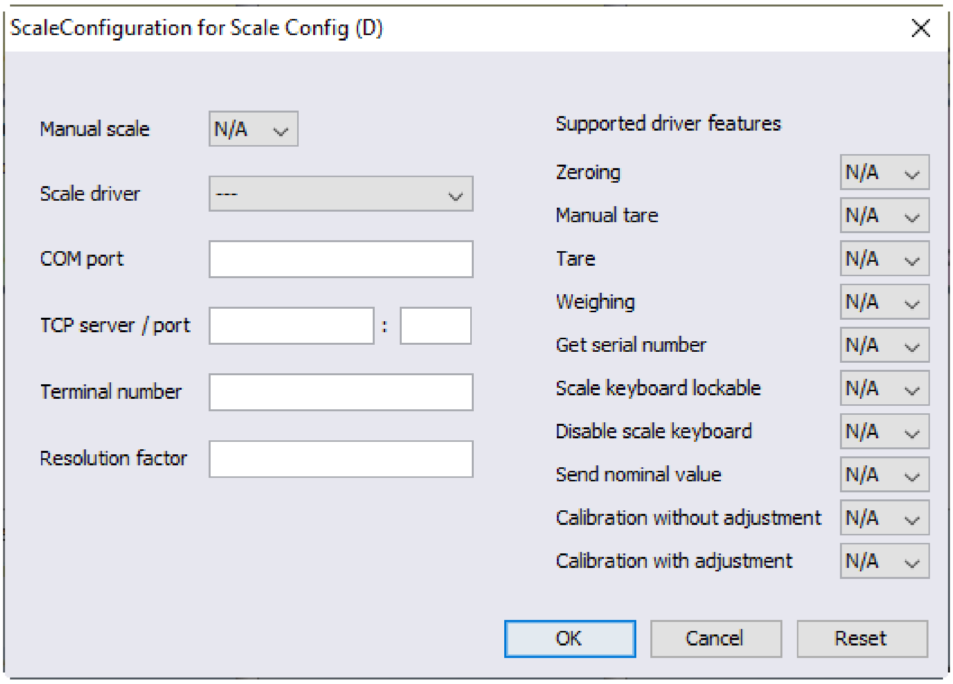
- Scale keyboard lockable
  Select this option if the scale's keyboard can be locked and its driver has the control of this option implemented. It is a required feature if you wish to select the **Disable scale keyboard** option (page 63).

- Disable scale keyboard
  Select this option if the scale keyboard is to be disabled when the scale is used during execution. For this configuration the scale driver must also support the **Scale keyboard lockable** feature (page 63).

- Send nominal value
  Select this option if the scale can receive nominal values to display them.

- Calibration with adjustment
  Select this option if the scale performs calibration with subsequent adjustment.

- Calibration without adjustment
  Select this option if the scale supports a calibration procedure that only establishes the result of the performed test weighing, but does not adjust the scale.



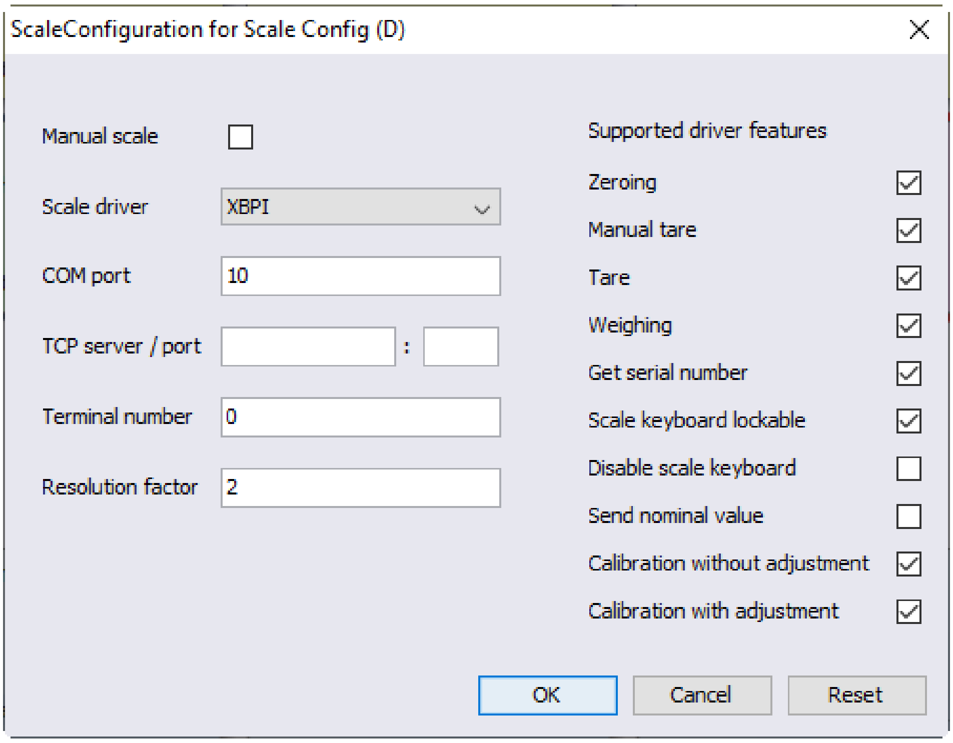
ScaleConfiguration for Scale Config (D)

Manual scale	N/A
Scale driver	---
COM port	
TCP server / port	:
Terminal number	
Resolution factor	

Supported driver features

Zeroing	N/A
Manual tare	N/A
Tare	N/A
Weighing	N/A
Get serial number	N/A
Scale keyboard lockable	N/A
Disable scale keyboard	N/A
Send nominal value	N/A
Calibration without adjustment	N/A
Calibration with adjustment	N/A

Figure 59: ScaleConfiguration editor for property of class



ScaleConfiguration for Scale Config (D)

Manual scale	□
Scale driver	XBPI
COM port	10
TCP server / port	:
Terminal number	0
Resolution factor	2

Supported driver features

Zeroing	☑
Manual tare	☑
Tare	☑
Weighing	☑
Get serial number	☑
Scale keyboard lockable	☑
Disable scale keyboard	□
Send nominal value	□
Calibration without adjustment	☑
Calibration with adjustment	☑

Figure 60: ScaleConfiguration editor for property of entity

- When you define a property for an equipment class, you can set all values. The system checks that entities of the class match the defined values.

  > TIP
  > If you set attribute values to **Undefined**, for example by selecting **---** or **N/A** as list option, the entities assigned to the class can have any value.

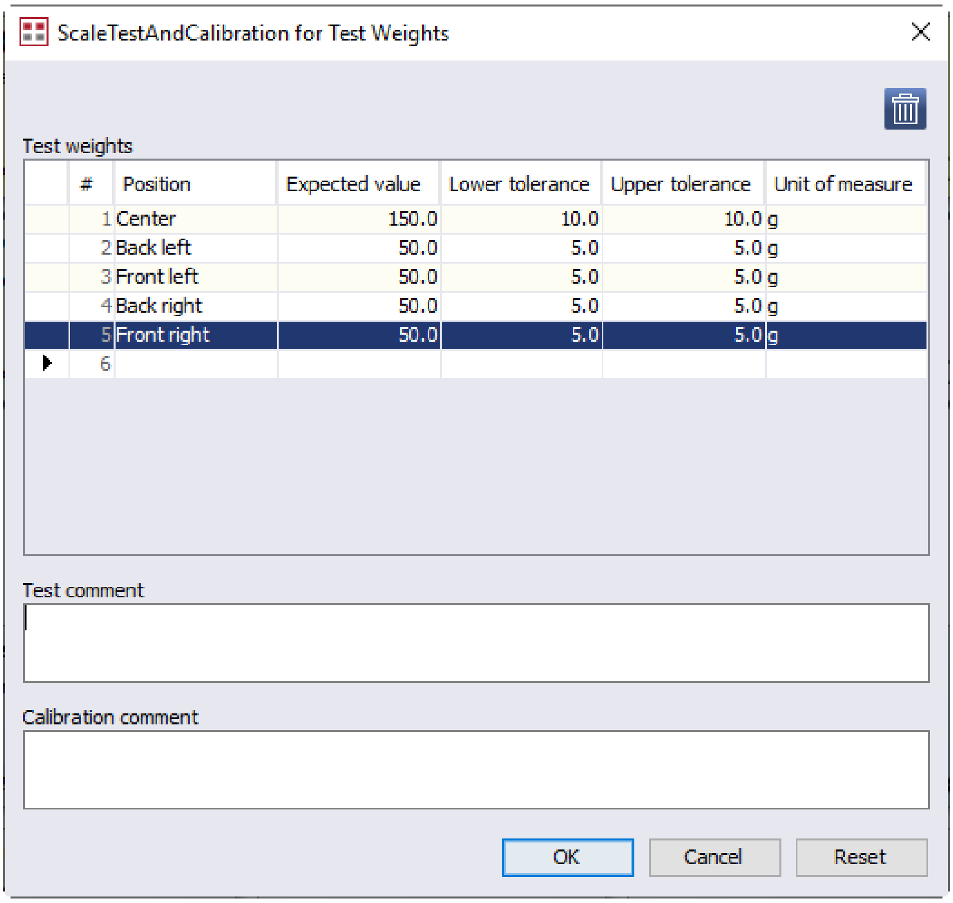
- When you define a property for an equipment entity, you can set all values and thus provide the information necessary for using the entity during execution.
- To save your data and close the ScaleConfiguration editor, click the **OK** button.
- To close the ScaleConfiguration editor without saving your changes, click the **Cancel** button.
- To reset your data to its last saved form, click the **Reset** button.

ScaleTestAndCalibration Editor

The ScaleTestAndCalibration editor supports you with defining the test weights and additional test and calibration information. You can define up to twenty test weights per scale by specifying the where on the scale an operator shall place the weight (**Position**), the **Expected value**, as well as the **Upper tolerance** and the **Lower Tolerance**, in the unit of measure of the value.

> TIP
>
> Please note that the system expects the same number of fractional digits for the **Expected value**, the **Lower tolerance**, and the **Upper tolerance** of a test weight.

If required, you can define additional comments for test and calibration, which a recipe or workflow author can choose to display as instructions to operators on the shop floor.



Test weights

	#	Position	Expected value	Lower tolerance	Upper tolerance	Unit of measure
	1	Center	150.0	10.0	10.0	g
	2	Back left	50.0	5.0	5.0	g
	3	Front left	50.0	5.0	5.0	g
	4	Back right	50.0	5.0	5.0	g
	5	Front right	50.0	5.0	5.0	g
▶	6					

Test comment

Calibration comment

Figure 61: ScaleTestAndCalibration editor for property

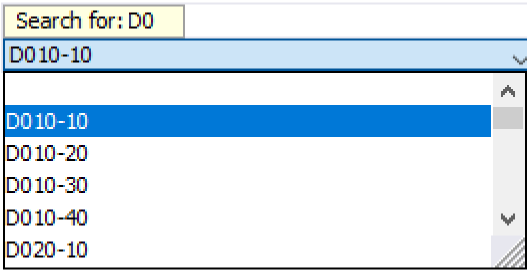
- When you define a property for an equipment class, you can set all values. The system checks that entities of the class match the defined values.

TIP
If you set attribute values to **Undefined**, for example by leaving an input box blank, the entities assigned to the class can have any value.

- When you define a property for an equipment entity, you can set all values and thus provide the information necessary for using the entity during execution.
- To save your data and close the ScaleTestAndCalibration editor, click the **OK** button.
- To close the ScaleTestAndCalibration editor without saving your changes, click the **Cancel** button.
- To reset your data to its last saved form, click the **Reset** button.

Searchable Option List Editor

The Searchable Option List editor supports you with selecting from option lists with a large number of selectable items.



Search for: D0

D010-10

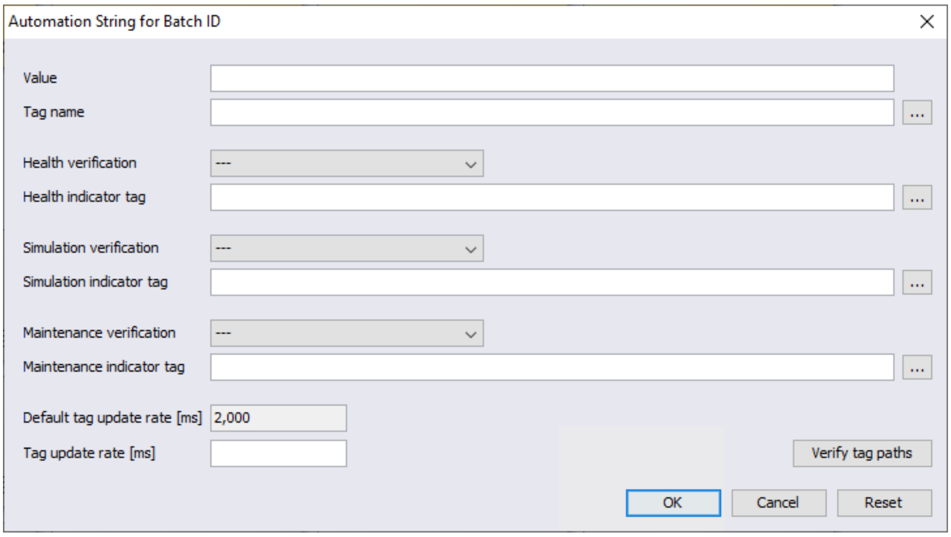
D010-10
D010-20
D010-30
D010-40
D020-10

Figure 62: Searchable Option List editor

- To view all options, open the list and use the scroll bar or enlarge the list by dragging the handle in its lower right corner.
- To find specific options, start typing. The system immediately moves the list to the first option that starts with the character you have typed and selects it.
- To refine your search, type again. The system opens a temporary search box, displaying the search string you are typing and also indicates if there are no list options to match the string.

String Editor for Automation

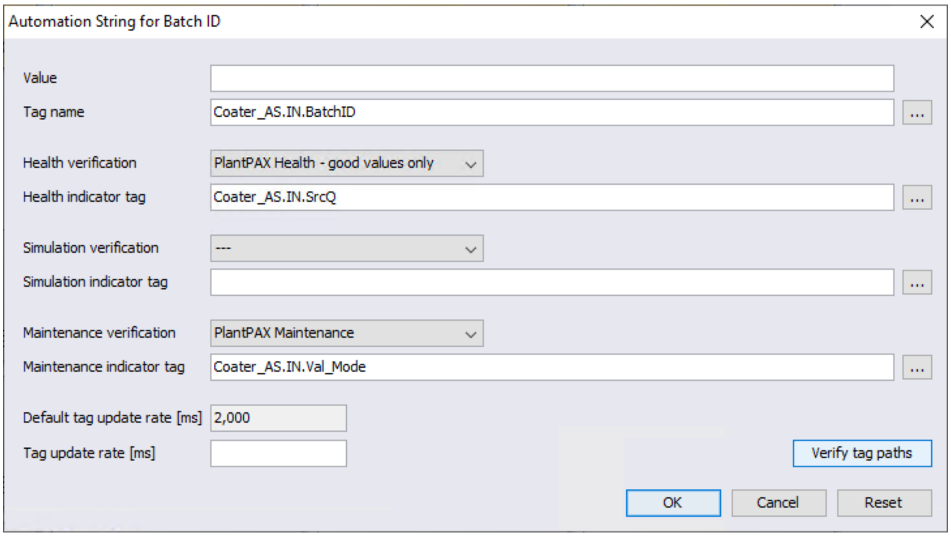
The String editor supports you with handling any sequence of characters, such as a batch identifier, you need to communicate to or from an automated equipment entity. In addition, you can select verification schemas for tags that verify a value's tag health, its simulation status, or its maintenance status.



Automation String for Batch ID

Value	
Tag name	
Health verification	---
Health indicator tag	
Simulation verification	---
Simulation indicator tag	
Maintenance verification	---
Maintenance indicator tag	
Default tag update rate [ms]	2,000
Tag update rate [ms]	

Figure 63: Automation String editor for property of equipment class



Automation String for Batch ID

Value:

Tag name: Coater_AS.IN.BatchID

Health verification: PlantPAX Health - good values only

Health indicator tag: Coater_AS.IN.SrcQ

Simulation verification: ---

Simulation indicator tag:

Maintenance verification: PlantPAX Maintenance

Maintenance indicator tag: Coater_AS.IN.Val_Mode

Default tag update rate [ms]: 2,000

Tag update rate [ms]:

Figure 64: Automation String editor for property of equipment entity

- When you define a property for an equipment class, you can set a value and select verification schemas. The data only has informational value, however and is neither matched against the data defined for entities of the class nor against data defined with a property requirement during recipe or workflow design.
- When you define a property for an equipment entity, you can set a value and select verification schemes. At this point you also need to define to which tags of your equipment you wish to map the defined value and schemes, and with which update rate the system is to access the tags.

> **TIP**
> To see the list of tags available at your equipment and to select the suitable tag, click the **...** button to open the Tag Browser editor (page 70).

- To check if the indicated tag name corresponds to an existing tag name on the equipment entity, click the **Verify tag paths** button.
- To save your data and close the Automation String editor, click the **OK** button.
- To close the Automation String editor without saving your changes, click the **Cancel** button.
- To reset your data to its last saved form, click the **Reset** button.

String Editor for Historian

The String editor supports you with handling any sequence of characters, such as a process step descriptor, you wish to display on a historical data chart.

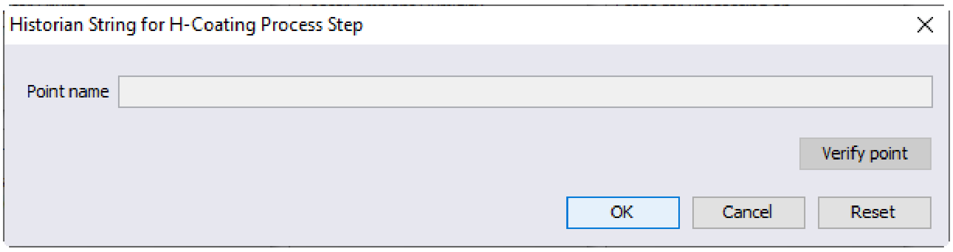
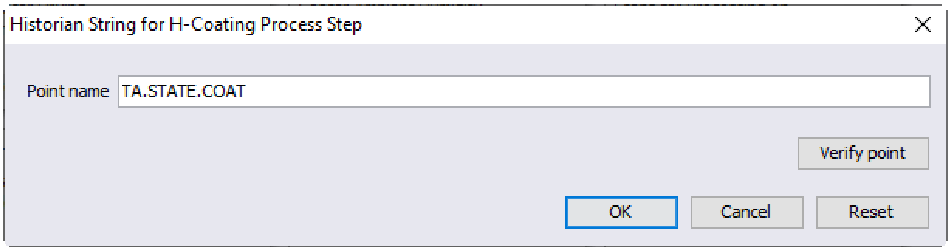


Figure 65: Historian String editor for property of equipment class



Point name: TA.STATE.COAT

Figure 66: Historian String editor for property of equipment entity

- When you define a property for an equipment class, you cannot set any values.
- When you define a property for an equipment entity, you need to define which point of your Historian server you wish access to retrieve values.
  To check if the indicated point name corresponds to an existing point defined on the Historian server, click the **Verify point** button.
- To save your data and close the Historian String editor, click the **OK** button.
- To close the Historian String editor without saving your changes, click the **Cancel** button.
- To reset your data to its last saved form, click the **Reset** button.

Tag Browser Editor for Automation

The Tag Browser editor supports you with selecting the equipment tags to be mapped to the values and schemes you are defining for an automation property of an equipment entity.

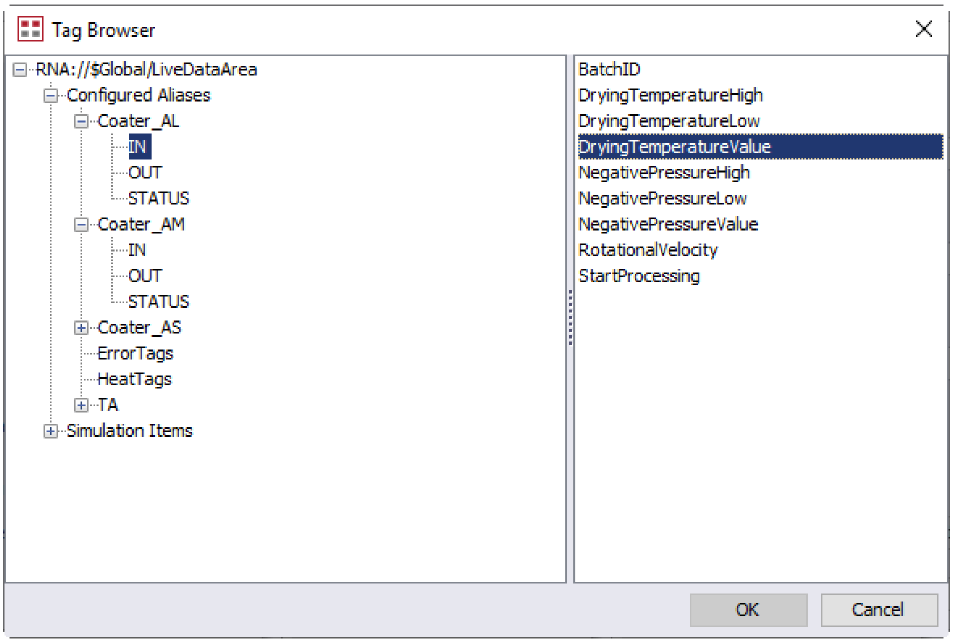


Figure 67: Tag Browser editor for automation properties

- To view the available tags, expand the nodes in the left panel of the editor, which shows the tag structure accessible through the Life Data Area path defined with the **Automation Attributes** on the **Basic** tab of the equipment entity. The system lists the tags available for each node in the right panel of the editor.
- To select a tag and close the Tag Browser editor, double-click the tag, or single-click it and then click the **OK** button.
- To close the Tag Browser editor without selecting a tag, click the **Cancel** button.

Graphical Elements

The following lists contain all buttons, marker icons, and cursors used in Data Manager.

Toolbar Buttons

Action buttons displayed in the toolbars:

Change status

In the Equipment mode of Data Manager, located on the File toolbar, it triggers a status change (page 130) for the object that is currently active in the lower tab bar of the Details window.

Close

Located on the File toolbar, it closes the Details view of the object that is currently active in the lower tab bar.

Delete

Located on the File toolbar, it deletes the object that is currently active in the lower tab bar of the Details window.

Help

Located on the Help toolbar, it opens a web browser to display the context-sensitive help (page 7) of Data Manager.

New equipment class

In the Equipment mode of Data Manager, located on the File toolbar, it creates a new equipment class, thus first opening the **New Equipment Class** dialog and afterwards a new tab in the lower tab bar of the Details window.

New equipment entity

In the Equipment mode of Data Manager, located on the File toolbar, it creates a new equipment entity, thus first opening the **New Equipment Entity** dialog and afterwards a new tab in the lower tab bar of the Details window.

New equipment graph

In the Equipment mode of Data Manager, located on the File toolbar, it creates a new equipment graph, thus first opening the **New Equipment Graph** dialog and afterwards a new tab in the lower tab bar of the Details window.

New property type

In the Equipment mode of Data Manager, located on the File toolbar, it creates a new property type, thus first opening the **New Property Type** dialog and afterwards a new tab in the lower tab bar of the Details window.

New station

In the Work Center mode of Data Manager, located on the File toolbar, it creates a new station, thus first opening the **New Station** dialog and afterwards a new tab in the lower tab bar of the Details window.

New template equipment entity

In the Equipment mode of Data Manager, located on the File toolbar, it creates a new template equipment entity, thus first opening the **New Template Equipment Entity** dialog and afterwards a new tab in the lower tab bar of the Details window.

New work center

In the Work Center mode of Data Manager, located on the File toolbar, it creates a new work center, thus first opening the **New Work Center** dialog and afterwards a new tab in the lower tab bar of the Details window.

Print barcode label

In the Equipment mode of Data Manager, only available for equipment entities.
In the Work Center mode of Data Manager, only available for stations.
Located on the File toolbar, it sends the barcode of the equipment entity or station that is currently active in the Details window for printing to a connected printer.

Refresh

Only available if there are no unsaved changes.
Refreshes the object that is currently active in the lower tab bar of the Details window by retrieving its data again from the database. Thus, it updates all runtime data, such as runtime properties, graph statuses or grouping situations, which may have changed due to processing activities on the shop floor.

Save

Located on the File toolbar, it saves all changes made to the object that is currently active in the lower tab bar of the Details window. This action saves all changes made on the object's attributes and properties tabs.

Zoom in

Located on the View toolbar, it zooms in on the currently active search result, switching the display to the next larger view, either tiles or cards.

Zoom out

Located on the View toolbar, it zooms out from the currently active search result, switching the display to the next smaller view, either tiles or mini tiles.

Action Buttons

Action buttons available on the panels of Data Manager:

Delete

- For equipment graphs, on their **Status/Trigger** tab, deletes the selected status or trigger.
- For equipment graphs, on their **Transition** tab, deletes the selected transition.

Display on card

In the Equipment mode of Data Manager, on the dynamic properties tabs of the Details window of equipment classes, toggle button to define if a property is displayed on the card view of the class itself and all entities it styles. You can select a maximum of four properties for display on a card view.

Filter for classes

In the Equipment mode of Data Manager, located on the search criteria panel (page 92), it toggles the display of equipment classes in the search results panel.

Filter for entities

In the Equipment mode of Data Manager, located on the search criteria panel (page 92), it toggles the display of equipment entities in the search results panel.

Filter for graphs

In the Equipment mode of Data Manager, located on the search criteria panel (page 92), it toggles the display of equipment graphs in the search results panel.

Filter for property types

In the Equipment mode of Data Manager, located on the search criteria panel (page 92), it toggles the display of property types in the search results panel.

Filter for stations

In the Work Center mode of Data Manager, located on the search criteria panel (page 92), it toggles the display of stations in the search results panel.

Filter for template entities

In the Equipment mode of Data Manager, located on the search criteria panel (page 92), it toggles the display of template equipment entities in the search results panel.

Filter for work centers

In the Work Center mode of Data Manager, located on the search criteria panel (page 92), it toggles the display of work centers in the search results panel.

Move down

In the Equipment mode of Data Manager

- on the dynamic properties tabs of the Details window, for sorting the list of properties. It moves the selected property one row down.
- for equipment classes and entities on their **Graph** tabs, for sorting the list of graphs. It moves the selected graph one row down.
- for equipment graphs on their **Transition** tabs, for sorting the sub-tables for actions. It moves the selected action one row down, thus changing the order in which the actions are executed.

Move up

In the Equipment mode of Data Manager

- on the dynamic properties tabs of the Details window, for sorting the list of properties. It moves the selected property one row up.
- for equipment classes and entities on their **Graph** tabs, for sorting the list of graphs. It moves the selected graph one row up.
- for equipment graphs on their **Transition** tabs, for sorting the sub-tables for actions. It moves the selected action one row up, thus changing the order in which the actions are executed.

Print equipment logbook report

In the Equipment mode of Data Manager, on **Logbook** tabs of the Details window, opens the equipment logbook report in a preview window, from which you can send it to a connected printer.

Open cell editor

Located in a focused table cell of the Details window, it opens a suitable cell editor for editing the content of a cell.

Refresh

For searching with the Search window and date range filtering in the **Change History** tabs of the Details window, synchronizes the database snapshot taken at the beginning of the search or filter operation with the central database.
In the Equipment mode of Data Manager, also available for **Logbook** and **Status History** tabs of the Details window.

Remove

- On the dynamic properties tabs of the Details window, removes the selected property from the list, thus revoking the assignment of the property type to the object.
- For equipment classes
  - on their **Graph** tabs in the Details window, removes the selected graph from the list, thus revoking the assignment of the graph to the class.
  - on their **Entity** tabs in the Details window, removes the selected equipment entity from the list, thus revoking the assignment of the entity to the class.
- For equipment entities
  - on their **Graph** tabs in the Details window, removes the selected graph from the list, thus revoking the assignment of the graph to the entity.
  - on their **Class** tabs in the Details window, removes the selected equipment class from the list, thus revoking the assignment of the class to the entity.
  - on their **Grouping** tabs in the Details window, removes the selected equipment entity from the list, thus revoking the assignment of the child entity to the entity.
- For equipment graphs on their **Property Type** tabs, removes the selected property type from the list, thus revoking the assignment of the property type to the graph.
- For work centers, on their **Station** tabs in the Details window, removes the selected station from the list, thus revoking the assignment of the station to the work center.

Trigger graph transition

On the **Graph** tab of an entity, opens the **Trigger Graph Transition** dialog to execute an available transition of the selected graph.

Marker Icons

Marker icon displayed in Data Manager:

Class
On the card or tile views of an object, indicates that the object is a class, holding a group of entities.

Displayed on card
On the dynamic properties tabs of the Details window, displayed as overlay on the usage/object type markers (page 77) for **Specification**, **Automation**, **Runtime**, **Historian** usage types, or the **Graph** object type to the left of a listed property or object. Indicates that the property or object data has been selected for display on the object's card view.

Entity
On the card or tile views of an object, indicates that the object is an entity.

Error
Displayed in the Details window of equipment graph objects, on the tables of the **Status/Trigger** and **Transition** tabs, it indicates that the marked table row contains an error. To open a tooltip that shows further error details, hover over the icon.

Graph
On the card or tile views of an object, indicates that the object is a graph.

Property type
On the card or tile views of an object, indicates that the object is a property type.

Station
On the card or tile views of an object, indicates that the object is a station.

Template entity
On the card or tile views of an object, indicates that the object is a template entity.

Usage/object type
On the dynamic properties tabs of the Details window, displayed to the left of a listed property or object. Indicates the usage type of the property or the object type, respectively:

Specification

Automation

Runtime

Historian

Graph

Work center
On the card or tile views of an object, indicates that the object is a work center.

Cursors

Cursors displayed while editing data objects:

Drop target allowed
Indicates that the object you are currently dragging can be dropped on this tab of the Details window.

Drop target unallowed
Indicates that the object you are currently dragging cannot be dropped on this tab of the Details window.

Select
Indicates that you can double-click to open the Details window of the component over which the cursor hovers.

Wait
Indicates that the system is currently busy.

Expression Editor Icons

Icons displayed in the Expression editor (page 29):

Attributes
Displayed as section heading in the auto-completion feature of the expression panel (page 32), it indicates that the listed options are attributes, such as the attributes of property types.

Constants
Displayed as section heading in the auto-completion feature of the expression panel (page 32), it indicates that the listed options are constants.

Commands
Displayed as section heading in the auto-completion feature of the expression panel (page 32), it lists the commands available for transition actions.

Error
Displayed in the expression panel (page 32), it indicates that the expression contains one or more errors. To open a tooltip that lists the errors, hover over the icon.

Functions
Displayed at the root node of the functions tree (page 48) or as section heading in the auto-completion feature of the expression panel (page 32), it indicates that the listed options are functions.

List options
Displayed as section heading in the auto-completion feature of the expression panel (page 32), it indicates the options of a list, such as the available purposes.

Operators
Displayed at the root node of the operators tree (page 42) or as section heading in the auto-completion feature of the expression panel (page 32), it indicates that the listed options are operators.

References
Displayed as section heading in the auto-completion feature of the expression panel (page 32), it indicates that the listed options are references, such as property types.

Separators
Displayed as section heading in the auto-completion feature of the expression panel (page 32), it indicates that the listed options are separators.

Keyboard Operation

Control by keyboard is primarily necessary for navigation and editing purposes.

Framework Navigation Shortcuts

Use the following keyboard shortcuts to navigate the general framework of Data Manager:

- ALT+F1
  Opens a web browser to display the start page of the help system (page 7) of Data Manager - Equipment or Data Manager - Work Center.

- ALT+F4
  Closes the application window.

- CTRL+MINUS
  Zooms out from the currently active search result, switching the display to the next smaller view, either tiles or mini tiles.
  In equipment management, in the Details window, on the **Transition** tab of an equipment graph, collapses all action sub-tables.

- CTRL+N
  Opens a new Search window to define another set of search criteria and process its search results.

- CTRL+PLUS
  Zooms in on the currently active search result, switching the display to the next larger view, either tiles or cards.
  In equipment management, in the Details window, on the **Transition** tab of an equipment graph, expands all action sub-tables.

- CTRL+SHIFT+A
  Toggles the display of objects in the **Archived** status in the results panel.

- CTRL+T
  Toggles the display of the card view as tooltip when you hover over a tile or mini tile.

- F1
  Opens a web browser to display the context-sensitive help (page 7) of Data Manager - Equipment or Data Manager - Work Center.

Action Shortcuts

Use the following keyboard shortcuts to handle the actions available for objects in Data Manager:

- CTRL+B
  Only available for equipment entities.
  Sends the barcode of the equipment entity that is currently active in the Details window for printing to a connected printer.

- CTRL+F4
  Closes the object that is currently active in the lower tab bar of the Details window.

- CTRL+F12
  Opens the **Duplicate <Object Type>** dialog to create a copy of the object that is currently active in the lower tab bar of the Details window. The new object opens as new tab in the lower tab bar of the Details window.

- CTRL+R
  Only available if there are unsaved changes.
  Restores the object that is currently active in the lower tab bar of the Details window to its last saved state, thus undoing all changes made in the meantime.

- CTRL+S
  Saves all changes made to the object that is currently active in the lower tab bar of the Details window. This action saves all changes made on the object's attributes and properties tabs.

- CTRL+SHIFT+C
  Creates a new equipment class, thus first opening the **New Equipment Class** dialog and afterwards a new tab in the lower tab bar of the Details window.

- CTRL+SHIFT+E
  Creates a new equipment entity, thus first opening the **New Equipment Entity** dialog and afterwards a new tab in the lower tab bar of the Details window.

- CTRL+SHIFT+F4
  Closes all objects that are currently open in the lower tab bar of the Details window.

- CTRL+SHIFT+F12
  Only available for template equipment entities or equipment entities.
  - For template equipment entities
    Opens the **Duplicate Template Equipment Entity** dialog to create a copy of the template entity. The new template entity opens as new tab in the lower tab bar of the Details window.
  - For equipment entities
    Opens the **Duplicate Equipment Entity** dialog to create a copy of the entity. The new entity opens as new tab in the lower tab bar of the Details window.

  The **Duplicate with classes** function copies all attributes and properties of the original including all of its assignments.

- CTRL+SHIFT+G
  Creates a new equipment graph, thus first opening the **New Equipment Graph** dialog and afterwards a new tab in the lower tab bar of the Details window.

- CTRL+SHIFT+H
  Only available for status-controlled objects.
  Opens the **Change Status** dialog to perform a status change (page 126) on the object that is currently active in the lower tab bar of the Details window.

- CTRL+SHIFT+P
  Creates a new property type, thus first opening the **New Property Type** dialog and afterwards a new tab in the lower tab bar of the Details window.

- CTRL+SHIFT+S
  Saves all changes made to all object that are currently active in the lower tab bar of the Details window. This action saves all changes made on the objects' properties tabs.

- CTRL+SHIFT+T
  - In the Equipment modes of Data Manager
    Creates a new template equipment entity, thus first opening the **New Template Equipment Entity** dialog and afterwards a new tab in the lower tab bar of the Details window.
  - In the Work Center mode of Data Manager
    Creates a new station, thus first opening the **New Station** dialog and afterwards a new tab in the lower tab bar of the Details window.

- CTRL+SHIFT+W
  Creates a new work center, thus first opening the **New Work Center** dialog and afterwards a new tab in the lower tab bar of the Details window.

- CTRL+T
  On the **Graph** tab of an entity, opens the **Trigger Graph Transition** dialog to execute an available transition of the selected graph.

- CTRL+UP/DOWN ARROW
  On the dynamic properties tabs of the Details window, moves a selected list row up or down in the list.

- DEL
  On the dynamic properties tabs of the Details window, removes a selected assigned object (property, graph, entity, or class) from the list, thus revoking its assignment.
  On the **Status/Trigger** and **Transition** tabs of the Details window for graphs, deletes a selected graph component.

- F5
  Only available if there are no unsaved changes.
  Refreshes the object that is currently active in the lower tab bar of the Details window by retrieving its data again from the database. Thus, it updates all runtime data, such as runtime properties, graph statuses or grouping situations, which may have changed due to processing activities on the shop floor.

Expression Editor Shortcuts

Use the following keyboard shortcuts to manipulate text in the expression panel:

- ALT+F3
  In the expression panel, opens the quick search component.

- BACKSPACE
  In the expression panel, deletes the character to the left of the cursor.

- CTRL+A
  In the expression panel, selects all characters.

- CTRL+B
  In the expression panel, selects the current expression segment between the nearest brackets.

- CTRL+BACKSPACE
  In the expression panel, deletes the characters to the left of the cursor up to the next word.

- CTRL+C
  In the expression panel, copies the selected characters to the clipboard.

- CTRL+D
  In the expression panel, duplicates the selected characters.

- CTRL+DEL
  In the expression panel, deletes the characters to the right of the cursor up to the next word.

- CTRL+END
  In the expression panel, moves the cursor to the end of the text.

- CTRL+ENTER
  In the expression panel, inserts a line break after the current cursor position.

- CTRL+F
  In the expression panel, opens the **Find Text** dialog.

- CTRL+G
  In the expression panel, opens the **Go to line number** dialog to move the cursor to a specific line.

- CTRL+H
  In the expression panel, opens the **Replace Text** dialog.

- CTRL+HOME
  In the expression panel, moves the cursor to the beginning of the text.

- CTRL+LEFT ARROW
  In the expression panel, moves the cursor to the preceding word to the left.

- CTRL+MINUS
  In the expression panel, collapses all sections.

- CTRL+PLUS
  In the expression panel, expands all sections.

- CTRL+R
  In the expression panel, deletes the line in which the cursor is currently positioned.

- CTRL+RIGHT ARROW
  In the expression panel, moves the cursor to the next word to the right.
- CTRL+SHIFT+END
  In the expression panel, selects all characters from the current cursor position to the end of the text.
- CTRL+SHIFT+HOME
  In the expression panel, selects all characters from the current cursor position to the beginning of the text.
- CTRL+SHIFT+J
  In the expression panel, joins the current line with the following line.
- CTRL+SHIFT+LEFT ARROW
  In the expression panel, selects the word preceding the cursor position to the left.
- CTRL+SHIFT MINUS
  In the expression panel, collapses the section in which the cursor is currently positioned.
- CTRL+SHIFT+PLUS
  In the expression panel, expands the section in which the cursor is currently positioned.
- CTRL+SHIFT+RIGHT ARROW
  In the expression panel, selects the word following the cursor position to the right.
- CTRL+SHIFT+U
  In the expression panel, toggles the case of the currently selected letters to all uppercase or all lowercase.
- CTRL+SHIFT+V
  In the expression panel, opens a dialog to select one of the previous clipboards.
- CTRL+SPACEBAR
  In the expression panel, opens the intelligent auto-completion feature to list potential references or operators to insert.
- CTRL+V
  In the expression panel, pastes characters from the clipboard.
- CTRL+W
  In the expression panel, selects the word at the current cursor position.

- CTRL+X
  In the expression panel, cuts the selected characters and writes them to the clipboard.

- CTRL+Y
  In the expression panel, redoes the last action you have revoked with CTRL+Z. You can redo up to 100 actions, thus you can step by step redo the last 100 actions you have revoked.

- CTRL+Z
  In the expression panel, revokes the last action you have performed. You can undo up to 100 actions, thus you can step by step revoke the last 100 action you have performed.

- DEL
  In the expression panel, deletes the character to the right of the cursor.

- DOWN
  In the expression panel, moves the cursor down to the next lower line.

- END
  In the expression panel, moves the cursor to the end of the line.

- ENTER
  In the expression panel, inserts a line break.

- F3
  In the expression panel, searches the next occurrence of the text specified in the **Find Text** dialog.

- HOME
  In the expression panel, moves the cursor to the beginning of the line.

- INSERT
  In the expression panel, toggles the insert/overwrite input modes.

- LEFT ARROW
  In the expression panel, moves the cursor to the preceding character to the left.

- PAGE DOWN
  In the expression panel, moves the cursor one page down.

- PAGE UP
  In the expression panel, moves the cursor one page up.

- RIGHT ARROW
  In the expression panel, moves the cursor to the next character to the right.

- SHIFT+DOWN
  In the expression panel, selects all characters from the current cursor position to the next line.

- SHIFT+END
  In the expression panel, selects all characters from the current cursor position to the end of the line.

- SHIFT+ENTER
  In the expression panel, inserts a new line after the current line.

- SHIFT+F3
  In the expression panel, searches the previous occurrence of the text specified in the **Find Text** dialog.

- SHIFT+HOME
  In the expression panel, selects all characters from the current cursor position to the beginning of the line.

- SHIFT+LEFT ARROW
  In the expression panel, selects the character preceding the current cursor position to the left.

- SHIFT+PAGE DOWN
  In the expression panel, selects all characters from the current cursor position one page down.

- SHIFT+PAGE UP
  In the expression panel, selects all characters from the current cursor position one page up.

- SHIFT+RIGHT ARROW
  In the expression panel, selects the character following the current cursor position to the right.

- SHIFT+TAB
  In the expression panel, resets the indent of indented lines.

- SHIFT+UP
  In the expression panel, selects all characters from the current cursor position to the preceding line.

- TAB
  In the expression panel, inserts a tab character. The system provides tab stops at approximately every fifth character.
  If you have text selected, the entire line or lines containing the selected text is/are indented.

- UP
  In the expression panel, moves the cursor up to the next higher line.

Basic Operations

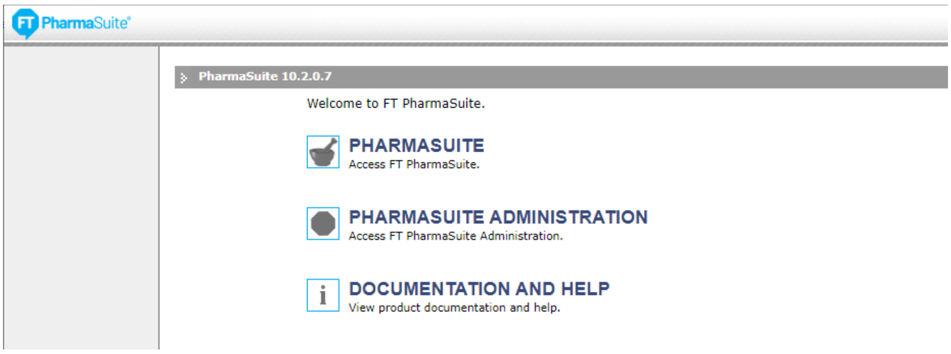
The following sections describe basic and recurring operations and functions in Data Manager.

Start, Login, Logout, and Password Change

Before you can start working with PharmaSuite your system administrator must have created a user account for you. The PharmaSuite administrator will inform you of your login name and initial password.
Depending on your company policy you may be forced to change your password when you log in for the first time. In this case, the system will display a message that indicates that your password has expired. Then the system will prompt you to change it (page 90).

START PHARMASUITE

To start PharmaSuite double-click the respective icon on the user interface or select it from the start menu. The system displays the webstart page in a browser window, from which you can select to start an application or view either the help system or the documentation.



Welcome to FT PharmaSuite.

PHARMASUITE
Access FT PharmaSuite.

PHARMASUITE ADMINISTRATION
Access FT PharmaSuite Administration.

DOCUMENTATION AND HELP
View product documentation and help.

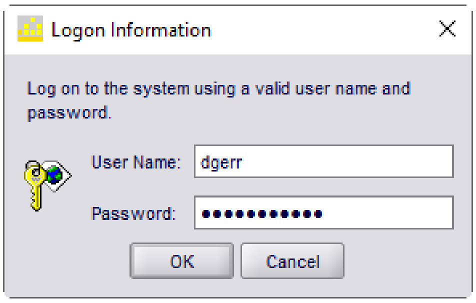
Figure 68: PharmaSuite webstart page

Login

When you select to start PharmaSuite it runs through an initialization phase in the course of which you will also see the splash screen of Shop Operations, which is the internal platform of PharmaSuite. As soon as the initialization phase has been completed, the login form for user login appears.

The login form contains two mandatory fields, one for the login name and one for the password. Your login name and your password are unique for all PharmaSuite applications and are linked to your role and user privileges.

Type your login name and password in the respective boxes. Please note that your password is masked by asterisks (*). Click the **OK** button to complete the login procedure. If your login attempt is not successful, a message appears, and you have to repeat the procedure.



Logon Information

Log on to the system using a valid user name and password.

User Name: dgerr

Password: ●●●●●●●●●●●

Figure 69: Login form

After you have successfully logged in, the system displays the PharmaSuite welcome page. From here you can start the Production Execution, Production Execution Viewer, Production Management, Data Manager, Recipe and Workflow Designer, and Production Responses applications, change your password and work station, or access the system documentation and help.

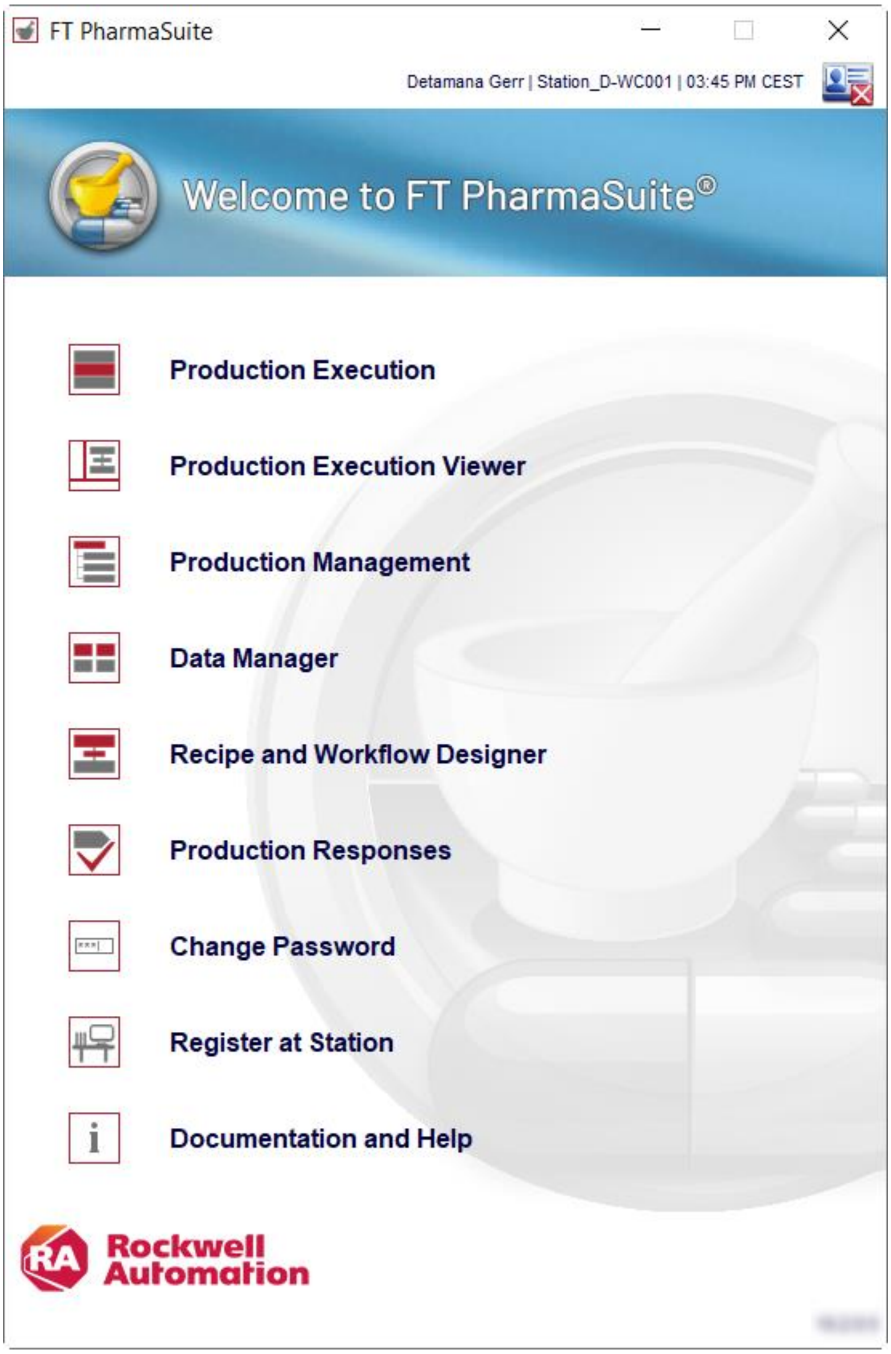


Figure 70: PharmaSuite welcome page

TIP

Please note that logins can be linked to access rights, which means that you can only start an application if your system administrator has assigned the suitable access privileges to you.
Some logins, especially in the production execution environment, are directly connected to an application and work station. This means that the welcome page will be skipped, and the application will start directly after you have successfully logged in.

Logout

In Data Manager, from the **File** menu, select the **Exit** function or use the ALT+F4 keyboard shortcut to quit the application.

On the PharmaSuite welcome page, the **Logout** button is also located in the top right corner. Click it to return to the webstart page.

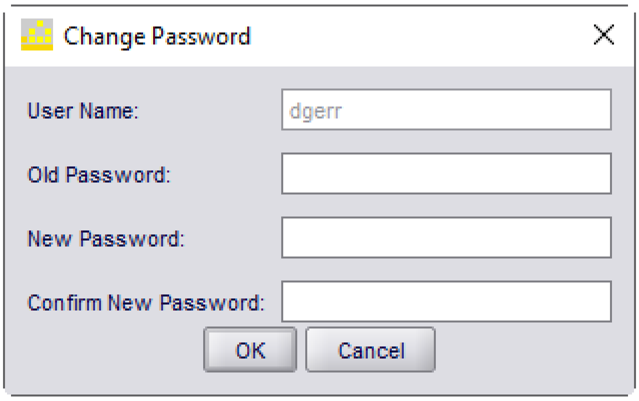
If you decide to log out from a running application, the system will request you to confirm the decision and also warn you if there is any unsaved data you may want to save before you log out.

Password Change

You can access the function for changing your password from the PharmaSuite welcome page.

1. Click the **Change Password** link to open the **Change Password** form.
   When your password expires the system will open the form automatically. This can also happen when you log in for the first time to force you to change the initial password, which your system administrator defined for you.
2. On the **Change Password** form, the **User Name** box is output-only and contains your login name.
3. Type your current password in the **Old Password** box.
4. Type your new password first in the **New Password** box and then in the **Confirm New Password** box.
   For security reasons, passwords are masked by asterisks (*).

5. Click the **OK** button to close the form.
   From now on, use the new password to log in.



Change Password

User Name: dgerr

Old Password:

New Password:

Confirm New Password:

Figure 71: Change password

Required Servers

For providing its full functional scope, PharmaSuite relies on the following servers that are responsible for communication to external systems or between its applications.

- Electronic Batch Recording (EBR) server
  It controls the execution of EBR recipes and workflows and can process incoming messages from a Distributed Control System.
- Triggered Operation Management (TOM) server
  It manages event-triggered operations.
- Operation Execution (OE) server
  It controls the execution of server-run operations.
- Automation Integration (AI) server
  It controls the communication with automation-related systems.
- Transition server
  It performs automatic, system-triggered status changes on objects, such as master recipes, master workflows, batches, orders, workflows, or equipment entities and can process incoming messages from external systems, such as a Quality Management System or Warehouse Management.

PharmaSuite runs a heartbeat check on the servers to monitor their availability. To see if there are any issues, open the **About PharmaSuite** dialog (page 8), which shows the status of the EBR server. For information on the other servers, open the **Details** dialog and refer to the section that indicates servers with heartbeat issues.

Searching

The Search window provides you with a fast and efficient way to locate objects or groups of objects that match your search criteria and display them in the results panel.
In Data Manager - Equipment, the system provides two modes, Smart Search (page 96) and Basic Search (page 98), that differ with respect to their search spaces and their search refinement capabilities.
In Data Manager - Work Center, only the Smart Search (page 96) mode is available.

You can have several search tabs open simultaneously and thus switch between different search results.

- To open a new search tab, click the + button on the empty tab located to the right of your existing search tabs.
- To rename a search tab, double-click its title on the tab.
- To close a search tab, it must be active. Then hover over the right tab margin to make the **Close** button appear.

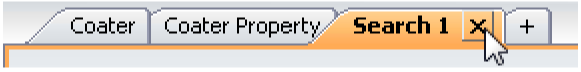
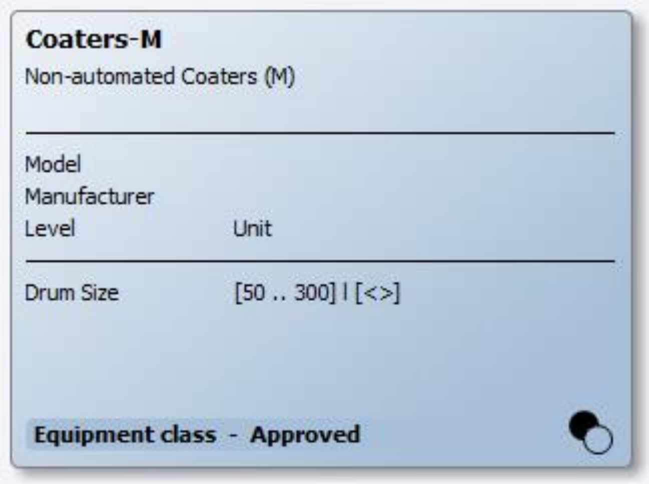


Figure 72: Search tabs with Close button

The **search results panel** displays all objects that match your search criteria. Objects are also listed as match when their assigned objects match the search criteria. The objects are ordered by object type. In the Equipment modes, the order is equipment classes first, then template equipment entities, equipment entities, equipment graphs, and finally property types. In the Work Center mode, the order is work centers first, stations second.
The system displays the number of hits found by your search criteria and the total number of objects in a semi-transparent overlay banner at the bottom of the results panel.

There are three zoom levels available, the card view, the tile view, and the mini tile view. The card view represents an overview of the most basic attributes of an object, such as its identifier, short description, icon (if available), and its object type with object status (if available) and marker icon. In the Equipment modes, you can configure the card view to display up to four of an object's other properties, while in the Work Center mode, the display attributes are not configurable.



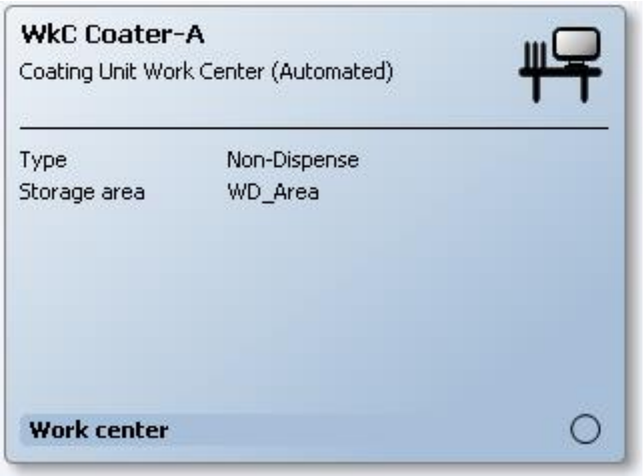
Coaters-M
Non-automated Coaters (M)

Model
Manufacturer
Level Unit

Drum Size [50 .. 300] l [<>]

Equipment class - Approved

Figure 73: Card of equipment class



WkC Coater-A
Coating Unit Work Center (Automated)

Type Non-Dispense
Storage area WD_Area

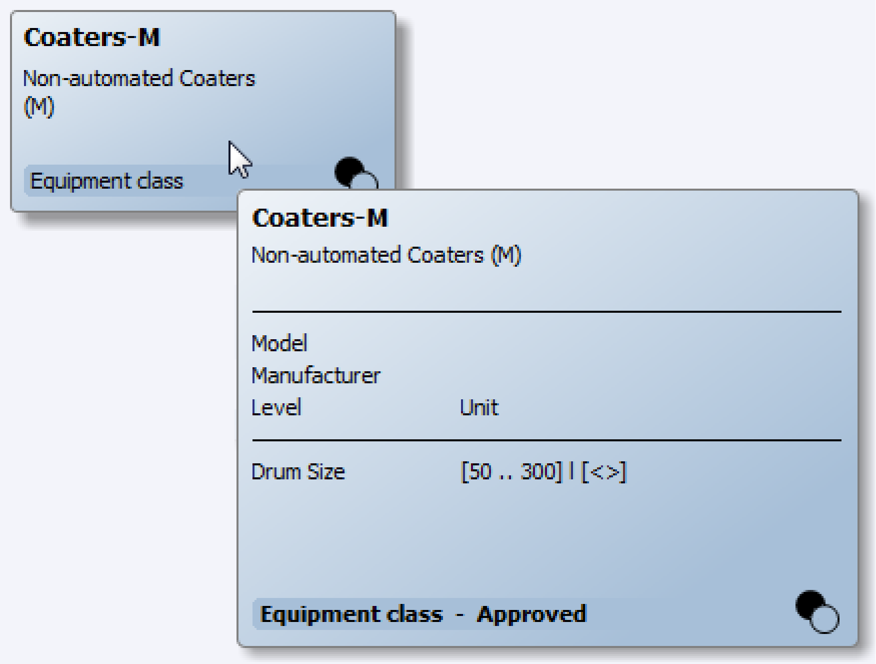
Work center

Figure 74: Card of work center

The tile view represents a zoomed-out version of the card view and displays the object's identifier, short description, icon (if available), and object type with marker icon. Hovering over a tile displays the card view as tooltip.

TIP

Please note that you can choose if the system displays the card tooltips. To switch the function off or on, open the **View** menu and unselect or select **Card tooltips** (CTRL+T).



Coaters-M

Non-automated Coaters (M)

Equipment class

Coaters-M

Non-automated Coaters (M)

Model

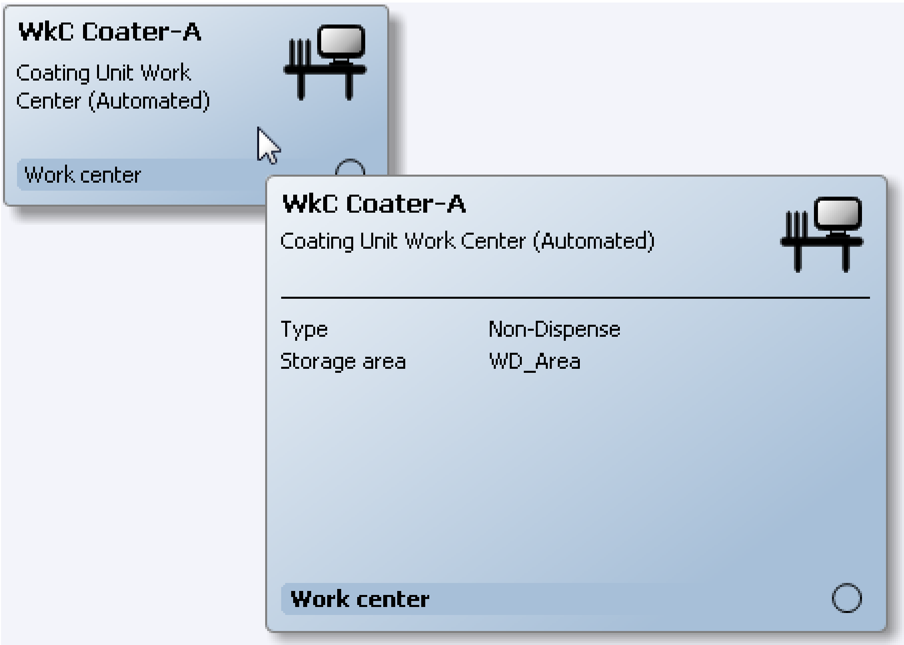
Manufacturer

Level Unit

Drum Size [50 .. 300] l [<>]

Equipment class - Approved

Figure 75: Tile of equipment class with tooltip



WkC Coater-A

Coating Unit Work Center (Automated)

Work center

WkC Coater-A

Coating Unit Work Center (Automated)

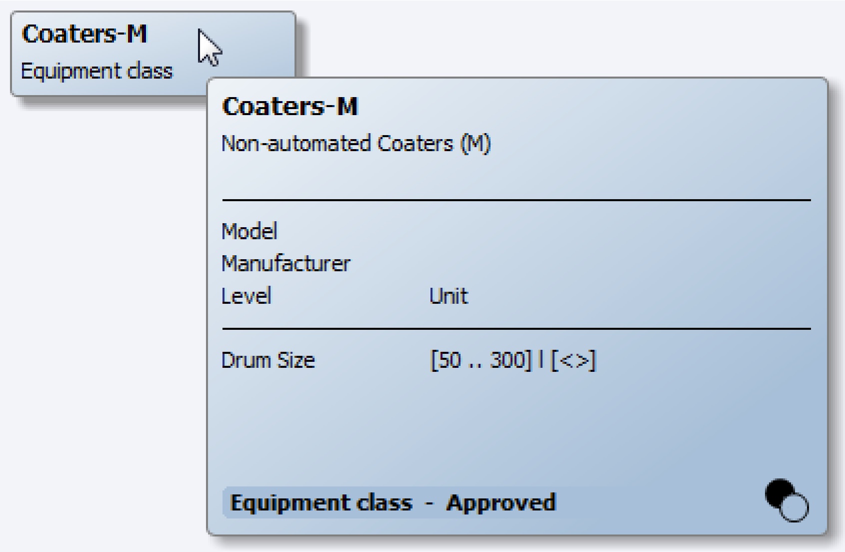
Type Non-Dispense

Storage area WD_Area

Work center

Figure 76: Tile of work center with tooltip

The mini tile view is further zoomed out to provide the best possible overview of the available objects and displays the object's identifier, icon, and type.



Coaters-M
Equipment class

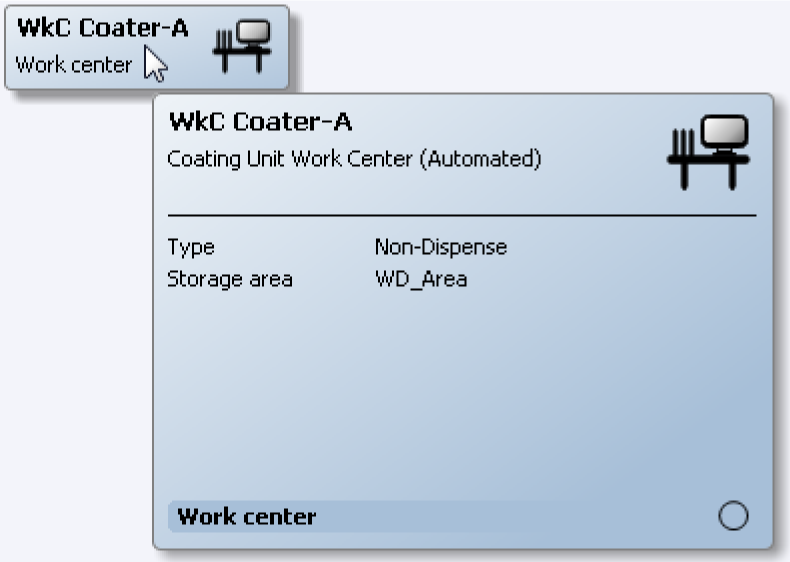
Coaters-M
Non-automated Coaters (M)

Model
Manufacturer
Level Unit

Drum Size [50 .. 300] l [<>]

Equipment class - Approved

Figure 77: Mini tile of equipment class with tooltip



WkC Coater-A
Work center

WkC Coater-A
Coating Unit Work Center (Automated)

Type Non-Dispense
Storage area WD_Area

Work center

Figure 78: Mini tile of work center with tooltip

To view and edit all attributes and properties of an object, double-click its card or tile. The system opens the object in the Details window (page 100).

Smart Search

The Search window provides you with a fast and efficient way to locate objects or groups of objects that match your search criteria and display them in the results panel. In Data Manager - Equipment, the system provides two modes, Smart Search and Basic Search, that differ with respect to their search spaces and their search refinement capabilities.

The search space of Smart Search consists of

- the values of the **Basic** attributes of all objects
- the identifiers of properties of the objects
- the values of those properties that are selected for display on the card views of the objects
- the identifiers of objects assigned to other objects
- the identifiers of objects that are part of the grouping structure of other objects.

In Smart Search, the system starts up with all data objects loaded and displayed in the results panel, sorted by object type. By refining your search, you can reduce the number of objects shown as result.

> **TIP**
>
> Please note that you can choose if the system displays **Archived** objects in the results panel. In the Smart Search mode, **Archived** objects are hidden by default. To switch the function off or on, open the **View** menu and unselect or select **Hide Archived objects** (CTRL+T).

To define your search, the **search criteria panel** supports you with a multi-functional input box, the capability to filter by object type, and a **Refresh** button:

- In the input box, type the string of characters you are looking for. The search becomes effective immediately with the first character you type.
- To clear the input box, click the **Cancel** icon that appears when you have typed the first character in the box.
- To filter your search results by object type, the system provides toggle buttons that allow you to define which types of objects you wish to display as search results:
  - For the Equipment mode: equipment classes, template equipment entities, equipment entities, equipment graphs, or property types.
  - For the Work Center mode: work centers or stations.
- To update your search space from the database, click the **Refresh** button. All objects that have been added or modified by another user since you started your search are then included.

TIP

By default, the search is not case-sensitive and returns results that contain all character strings given in the search box (AND mode) and match anywhere within the text strings contained in the search space.
Click the magnifying glass icon to open an option list for refining your search criteria:

- AND mode:
  All search strings are contained in the result objects.
- OR mode:
  Any of the search strings is contained in the result objects.
  Searches of this mode may return a large number of result objects.
- Phrase mode:
  The terms typed in the search box are treated as one search string, so the search returns only those objects as result that contain the exact string in the given order and with blanks.
  Searches of this mode allow you to search for objects that contain a specific phrase.
- Case-sensitive
- Case-insensitive
- Match from start:
  Result objects have a property whose value starts with the search string.
- Match exactly:
  Result objects have a property that is identical with the search string.
- Match anywhere:
  Result objects have a property whose value contains the search string.

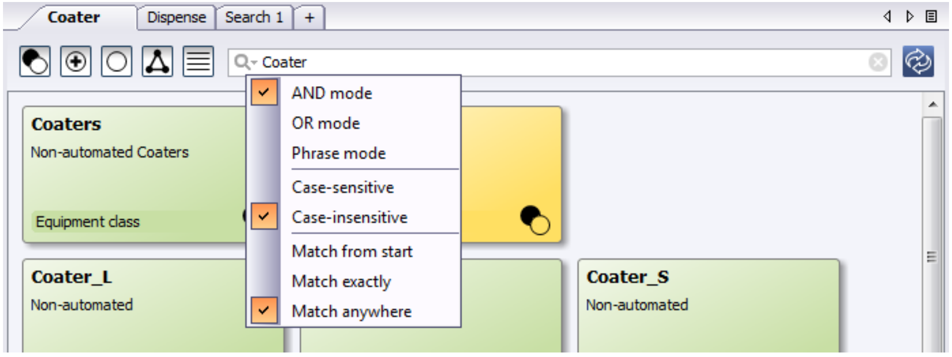


Figure 79: Search criteria panel - Equipment - Smart Search

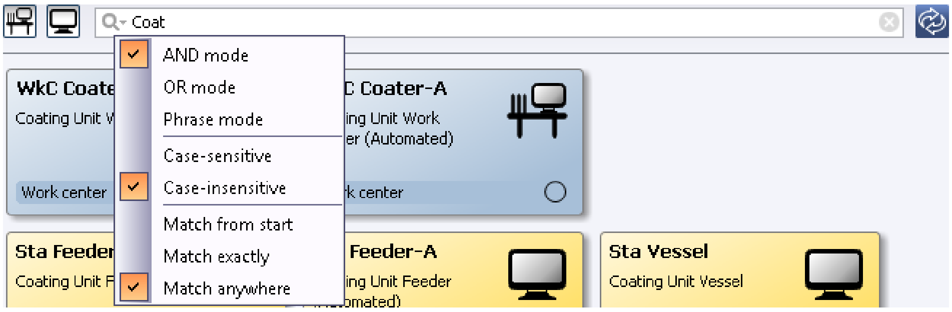


Figure 80: Search criteria panel - Work Center

Basic Search

The search space of Basic Search only considers object identifiers and starts up with an empty results panel. After entering search criteria, you need to trigger the search explicitly by pressing the RETURN key or clicking the **Refresh** button.

> **TIP**
>
> Please note that you can choose if the system displays **Archived** objects in the results panel. In the Basic Search mode, **Archived** objects are displayed by default. To switch the function off or on, open the **View** menu and unselect or select **Hide Archived objects** (CTRL+T).

To define your search, the **search criteria panel** supports you with a multi-functional input box, the capability to filter by object type, and a **Refresh** button:

- In the input box, type the string of characters you are looking for. To start the search, you need to press the RETURN key or click the **Refresh** button.
- To clear the input box, click the **Cancel** icon that appears when you have typed the first character in the box.
- To filter your search results by object type, the system provides toggle buttons that allow you to define which types of objects you wish to display as search results (equipment classes, template equipment entities, equipment entities, equipment graphs, or property types).
- To update your search space from the database, click the **Refresh** button. All objects that have been added or modified by another user since you started your search are then included.

TIP

By default, the search is not case-sensitive and returns results that contain all character strings given in the search box (AND mode) and match anywhere within the text strings contained in the search space.
Click the magnifying glass icon to open an option list for refining your search criteria:

- AND mode:
  All search strings are contained in the result objects.
- OR mode:
  Any of the search strings is contained in the result objects.
  Searches of this mode may return a large number of result objects.
- Case-sensitive
- Case-insensitive
- Match from start:
  Result objects have a property whose value starts with the search string.
- Match exactly:
  Result objects have a property that is identical with the search string.
- Match anywhere:
  Result objects have a property whose value contains the search string.

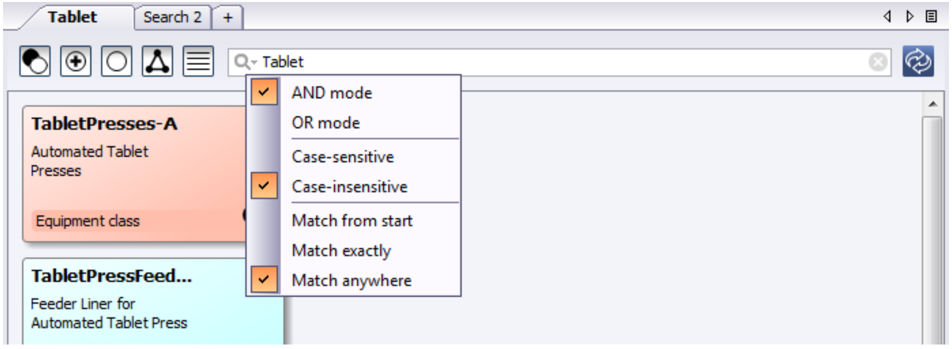


Figure 81: Search criteria panel - Equipment - Basic Search

Editing

The Details window contains all attributes and properties of an object. You can have the detail data of several objects open.
In Data Manager - Equipment - Smart Search and in Data Manager - Work Center, all objects that are open in the Details window have their tile or card frames highlighted in dark blue in the search results panel, while the currently active object has red highlighting.
In Data Manager - Equipment - Basic Search, however, only the currently active object is indicated with red highlighting. The search results panel does not display highlighting for other objects that are open in the Details window.

TIP

Please note that editing objects may be unavailable because

- data objects may be under status control and only editable in the **Draft** status, such as equipment classes (page 127) and equipment entities (page 129).
- the user rights linked to your login name control if you can access Data Manager only for viewing data objects, for editing only specific data objects, or for editing all data objects.
  If you can only view data objects, the system displays **[Read-only]** in the application title bar. **[Restricted Access]** indicates that the system allows you to perform a restricted set of functions.

For each object you choose to view or open, the Details window displays a tab in the tab bar at the bottom of the window. When you open an object for editing or create a new object, the system adapts the title of the Details window and opens a new tab in the lower tab bar to display:

- the object's identifier
- the object's type
- a status marker, if applicable.
  An asterisk (*) indicates that the object has unsaved changes, while objects that are not editable are marked by an (R).

TIP
Please note that objects that are in a read-only status can contain runtime data, which reflects activities performed on the shop floor. Since it is also possible to edit runtime data in Data Manager to resolve processing exceptions by administrative means, an object may display both the (R) and the (*) markers.

For each object, the Details window holds an upper tab bar with a **Basic** tab for the object's static attributes and may hold several other tabs for its object-specific properties.

- To open the detail data of an object, double-click its tile or card in the search results panel.
  The system opens a new tab in the lower tab bar.
- To assign another object to the object whose data you are editing in the Details window:
  1. In the search results panel, locate the object you wish to assign.
  2. Drag the object to the property tab on which you wish to place it.

     The system indicates by mouse cursor (┼ or ⊘) if the tab is suitable as target for the assignment.
     To switch to another tab than the one that is currently open, drag the object to the title of the other tab.
  3. Drop the object on the tab.
- To close the detail data of an object:
  - Click the **Close** button in the title bar of the object's Details window.
  - Click anywhere on the object's detail data tab to make it the currently active one.
    From the **File** menu, select the **Close <object identifier>** function.

Signature Requests

When performing safety-sensitive or GxP-relevant functions the system may request you to enter an electronic signature, for example during a status change. Signatures are linked to user groups and access privileges, which means that the system will only accept the signature of a user who is qualified to perform the task in question. Unless the required signature data has been entered correctly, subsequent functions cannot be executed.

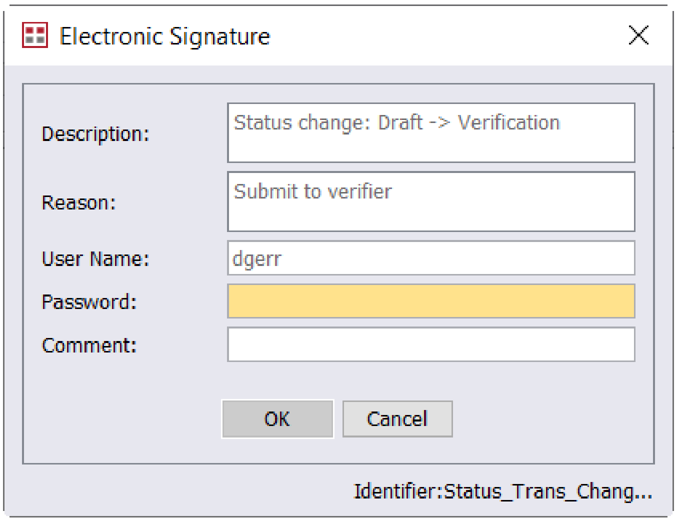
For situations requiring a witness, the system will ask not only for a single but for a double signature. In these cases, two different users, typically with different qualifications, have to complete the signature form before task processing can continue.

TIP

Please note that the system registers each signature with the timestamp when it has been verified successfully. Thus, the signature timestamps of a double signature will usually be different.

To perform an electronic signature, type your login name and password and click the **OK** button. Free-text comments can be optional or mandatory and may consist of up to 255 characters.

If a list of pre-defined comment texts is configured for a signature, the system displays an option list, which shows the headers of the available text options. Once an option has been selected, the system shows its full text in the read-only text box below the option list. If a pre-defined comment text is configured, it is mandatory. Later on, the pre-defined comment text is added as exception comment to the related exception.



Electronic Signature

Description: Status change: Draft -> Verification

Reason: Submit to verifier

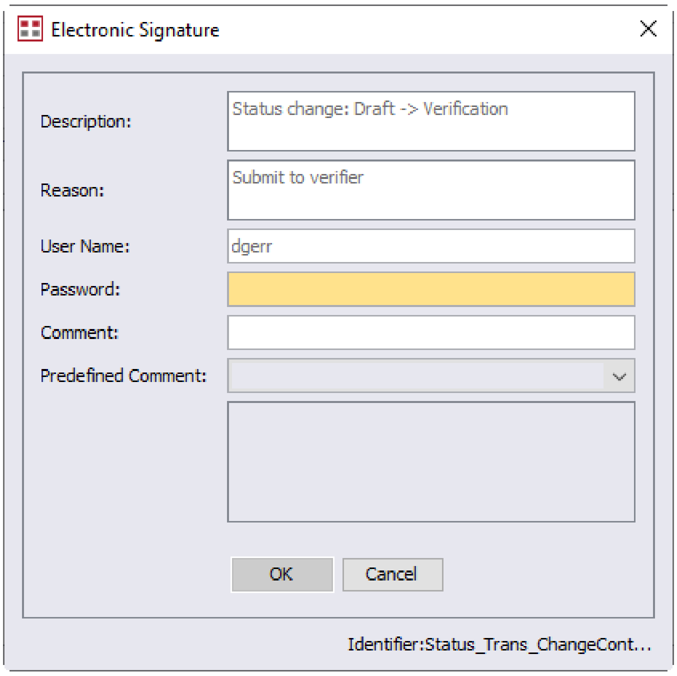
User Name: dgerr

Password:

Comment:

Identifier:Status_Trans_Chang...

Figure 82: Single electronic signature



Electronic Signature

Description: Status change: Draft -> Verification

Reason: Submit to verifier

User Name: dgerr

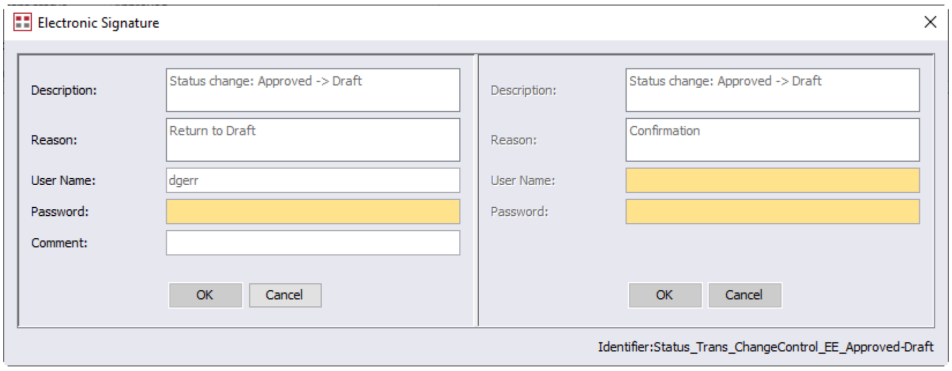
Password:

Comment:

Predefined Comment:

Identifier:Status_Trans_ChangeCont...

Figure 83: Single electronic signature with pre-defined comment



Description: Status change: Approved -> Draft
Reason: Return to Draft
User Name: dgerr
Password:
Comment:

Description: Status change: Approved -> Draft
Reason: Confirmation
User Name:
Password:

Identifier:Status_Trans_ChangeControl_EE_Approved-Draft

Figure 84: Double electronic signature to support witness role

Exporting and Importing Data

Data Manager provides several functions to cover the aspects of exporting equipment data from one system and later importing the data into another system:

- With the **Export** function (page 103), you can export a ZIP archive containing entire data objects, such as equipment entities, with all their relevant referenced objects.
- With the **Export tags/points** function (page 107), you can export the tags and points defined with the **Automation** and **Historian** properties of entities. The system creates an editable XML file for the exported data.
- With the **Import** function (page 110), you can import a ZIP archive created with the **Export** function. The system makes sure that the exported data objects have not been changed.
- With the **Import tags/points** function (page 115), you can import an XML file whose structure corresponds to that of a file exported with the **Export tags/points** function.
- With the **Verify tags/points** function (page 118), you can collectively verify all tag paths and point names defined for an equipment entity or for the tag paths and point names defined for all the entities of a class.

Exporting Objects

The **Export** function allows you to export equipment data objects (property types, graphs, entities, template entities, or classes) along with their relevant referenced objects.

- For **classes**, the system exports:
  - basic attributes with values except for the **Status**
  - style definitions made for the class
  - **Specification**, **Automation**, and **Historian** properties, including their content
  - **Runtime** properties, without content

  - property types on which the class' properties are based
  - template entities assigned to the class (including their assigned objects)
  - entities assigned to the class (including their assigned objects)
  - graphs assigned to the class (including their assigned objects)

> TIP
> When exporting classes, you can select to also export the archived template entities and/or the generated entities of the class. By default, the options are not selected and the objects are excluded from the export.

- For **template entities**, the system exports:
  - basic attributes with values except for the **Status**
  - class assignment information, but not the assigned class objects
  - style class setting, but not the style class object itself
  - **Specification**, **Automation**, and **Historian** properties, including their content
  - **Runtime** properties, without content
  - graphs assigned to the template entity (including their assigned objects)
  - property types on which the template entity's properties are based
- For **entities**, the system exports:
  - basic attributes with values except for the **Status** and the **Template used**
  - class assignment information, but not the assigned class objects
  - style class setting, but not the style class object itself
  - **Specification**, **Automation**, and **Historian** properties, including their content
  - **Runtime** properties, without content
  - graphs assigned to the entity (including their assigned objects)
  - property types on which the entity's properties are based
- For **graphs**, the system exports:
  - basic attributes with values except for the **Status**
  - style settings of the graph
  - statuses, triggers, and transitions of the graph
  - property types assigned to the graph
- For **property types**, the system exports:
  - basic attributes with values

TIPS

Please note that the **Status**, **Status History**, and **Change History** data of classes, template entities, entities, and graphs as well as the **Logbook** data of entities are not included in the export.

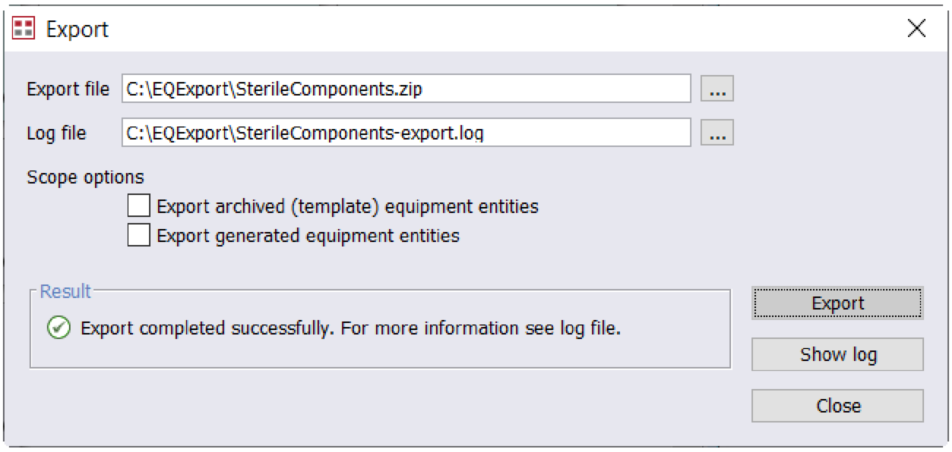
Usually, the most efficient export strategy is to export classes first, since class export packages include all of the class' template entities, entities, graphs, and property types. Only if the objects to be exported are neither assigned directly to a class nor to a (template) entity within the class, is it necessary to export them individually.

The system creates a ZIP archive that contains one BML file per exported object and an HTM file for meta data of the export package. Additionally, it creates a log file that lists the individual actions performed during the export.

To perform an export, proceed as follows:

1. In the Search window, locate the object you wish to export.
   Right-click it to open the object's shortcut menu and select the **Export <object identifier>** function or double-click to open the object and then, from the **File** menu, select **Export <object identifier>**.
   The system displays the **Export** dialog.
2. Click the ellipsis button (**...**) to select a location and name for the ZIP file the system will create.
   The system pre-fills both the **Export file** and **Log file** boxes with a default directory. For both files, it suggests the object's identifier as file name, but for the log file it also appends **-export**. The log file has **.log** as file extension. You can change the target locations and file names of both files, if required.
   When exporting a class, you can define if you wish to include archived (template) entities and/or generated entities in the scope of the export.
3. Click the **Export** button to run the export process. The system displays the overall result of the export in the **Result** box.
   In addition, to view the complete log that lists all steps performed during the export, click the **Show log** button.

4. To close the dialog, click the **Close** button.



Export

Export file: C:\EQExport\SterileComponents.zip

Log file: C:\EQExport\SterileComponents-export.log

Scope options

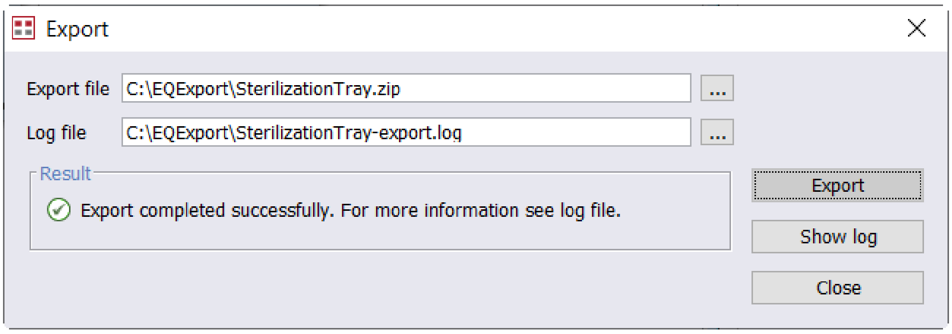
☐ Export archived (template) equipment entities

☐ Export generated equipment entities

Result

✓ Export completed successfully. For more information see log file.

Figure 85: Export dialog for class



Export

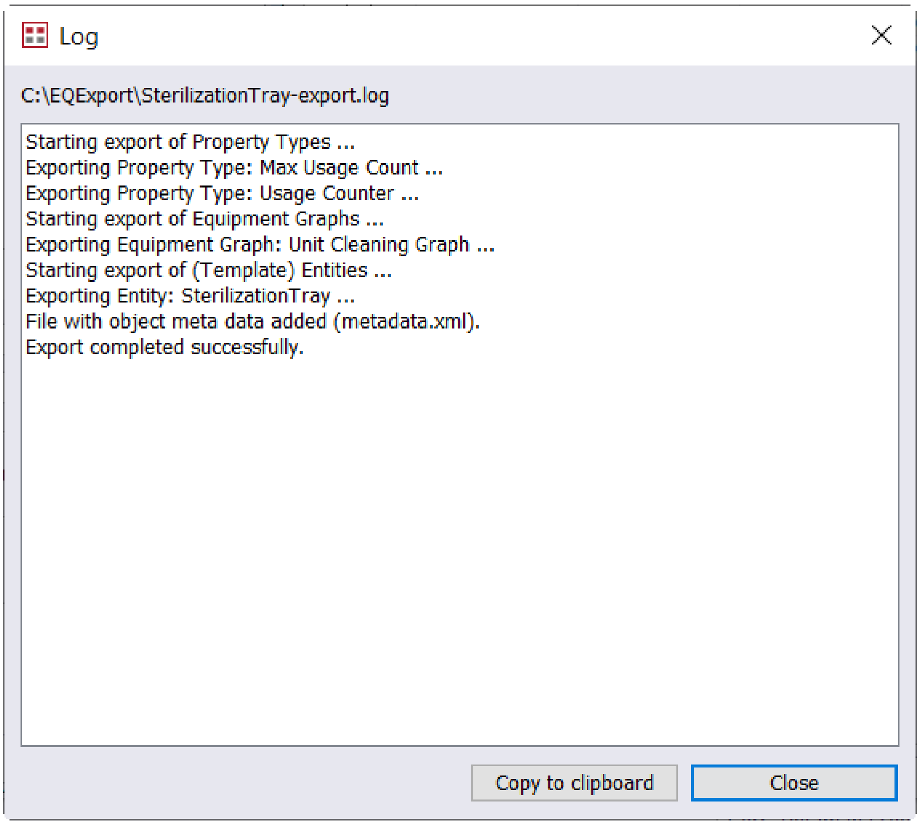
Export file: C:\EQExport\SterilizationTray.zip

Log file: C:\EQExport\SterilizationTray-export.log

Result

✓ Export completed successfully. For more information see log file.

Figure 86: Export dialog for entity



C:\EQExport\SterilizationTray-export.log

```
Starting export of Property Types ...
Exporting Property Type: Max Usage Count ...
Exporting Property Type: Usage Counter ...
Starting export of Equipment Graphs ...
Exporting Equipment Graph: Unit Cleaning Graph ...
Starting export of (Template) Entities ...
Exporting Entity: SterilizationTray ...
File with object meta data added (metadata.xml).
Export completed successfully.
```

Figure 87: Log dialog of class export

TIP

If you wish to retain the information listed in the **Log** dialog, click the **Copy to clipboard** button to copy the list and then paste it into a system-external text editor.

Exporting Tags/Points

The **Export tags/points** function allows you to export all tag and point data defined with the **Automation** and **Historian** properties of entities.

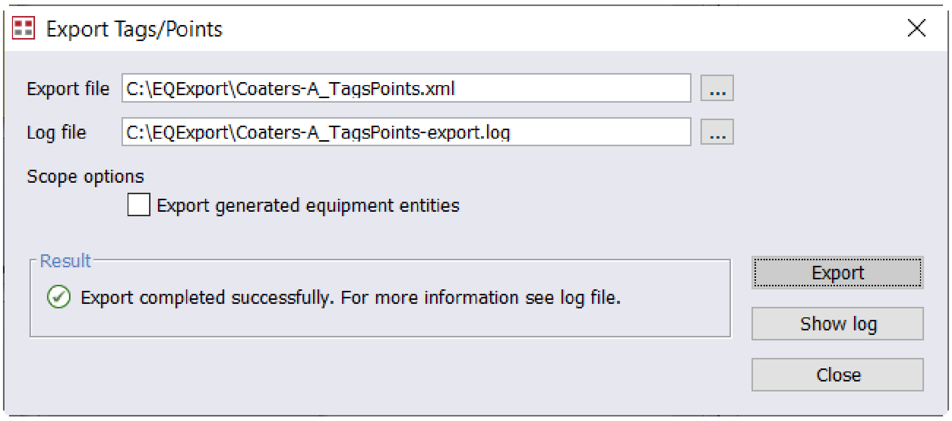
TIP

When exporting tags/points for classes, which exports the tags/points of the entities of the class, you can select to also export the tags/points of the generated entities of the class. By default, the option is not selected and the tags/points of generated entities are excluded from the export.

The system creates an XML file that contains the tag and point data of all entities included in the export. Additionally, it creates a log file that lists which entities are included in the export.

To perform an export, proceed as follows:

1. In the Search window, locate the object you wish to export.
   Right-click it to open the object's shortcut menu and select the **Export tags/points of <object identifier>** function or double-click to open the object and then, from the **File** menu, select **Export tags/points of <object identifier>**.
   The system displays the **Export Tags/Points** dialog.
2. Click the ellipsis button (**...**) to select a location and name for the XML file the system will create.
   The system pre-fills both the **Export file** and **Log file** boxes with a default directory. For both files, it suggests the object's identifier as file name, extended by **_TagsPoints**. For the log file, it also appends **-export**. The log file has **.log** as file extension. You can change the target locations and file names of both files, if required.
   When exporting the tags/points of a class, you can define if you wish to include the tags/points of generated entities in the scope of the export.
3. Click the **Export** button to run the export process. The system displays the overall result of the export in the **Result** box.
   In addition, to view the complete log that lists all steps performed during the export, click the **Show log** button.
4. To close the dialog, click the **Close** button.


Export Tags/Points

Export file C:\EQExport\Coaters-A_TagsPoints.xml

Log file C:\EQExport\Coaters-A_TagsPoints-export.log

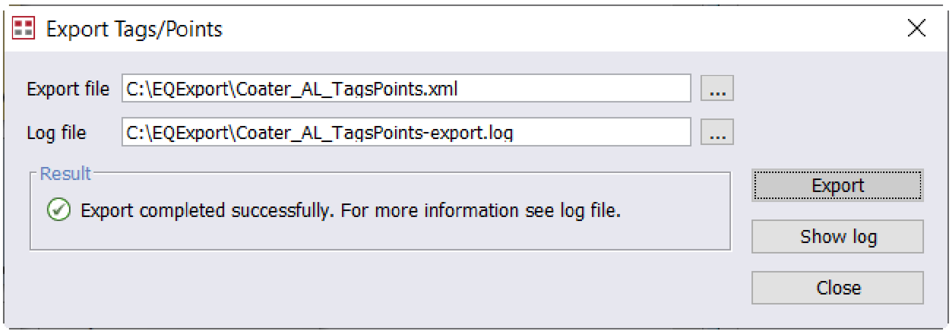
Scope options

□ Export generated equipment entities

Result

✓ Export completed successfully. For more information see log file.

Figure 88: Export Automation Tags dialog for class



Export Tags/Points

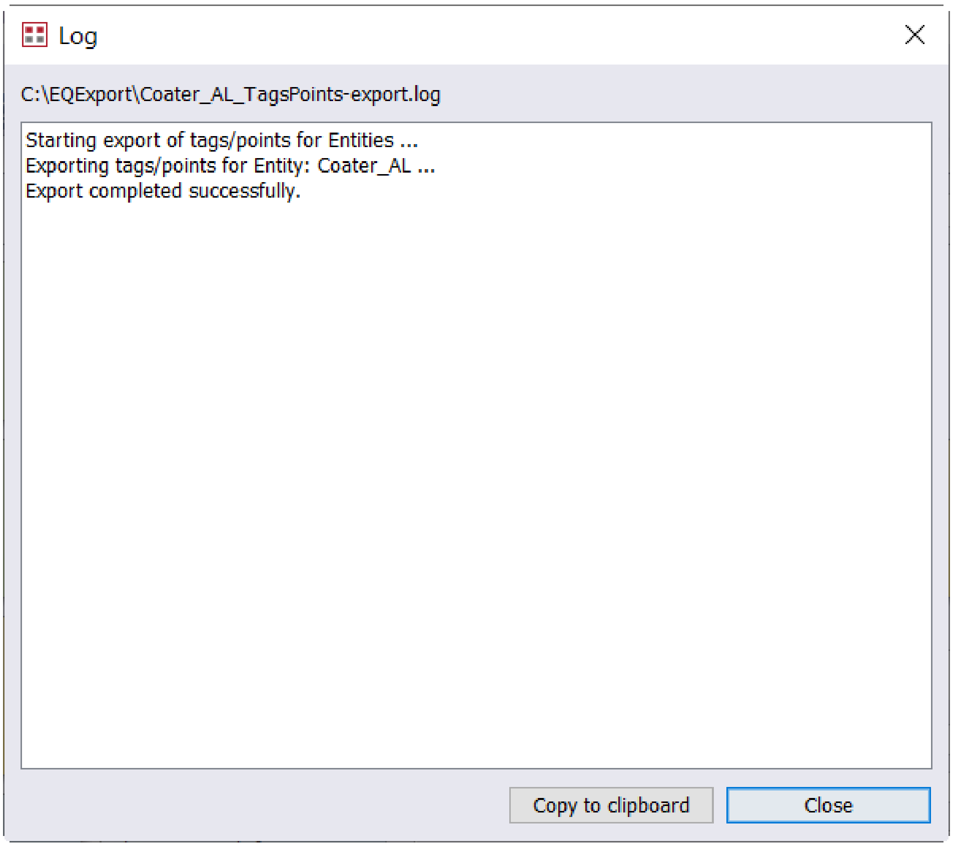
Export file C:\EQExport\Coater_AL_TagsPoints.xml

Log file C:\EQExport\Coater_AL_TagsPoints-export.log

Result

✓ Export completed successfully. For more information see log file.

Figure 89: Export Automation Tags dialog for entity



Log

C:\EQExport\Coater_AL_TagsPoints-export.log

```
Starting export of tags/points for Entities ...
Exporting tags/points for Entity: Coater_AL ...
Export completed successfully.
```

Figure 90: Log dialog of automation tag export

TIP

If you wish to retain the information listed in the **Log** dialog, click the **Copy to clipboard** button to copy the list and then paste it into a system-external text editor.

Importing Objects

The **Import** function allows you to import equipment data objects (property types, graphs, entities, or classes) along with their relevant referenced objects that were previously exported from another PharmaSuite system with the **Export** function (page 103) of Data Manager. It checks the objects to be imported against the objects that are already available in your system to determine potential data conflicts and resolves them by skipping objects or updating object data.

The system rejects an import file and aborts the import if

- the checksum of the files has changed, which happens when exported files are changed manually.
- the B2MML schema of the files to be imported differs from the schema expected by the system.

> **TIP**
>
> Please note that status-controlled objects (classes, template entities, entities, and graphs) are always imported in the **Draft** status, regardless of the status they had when they were exported.

The system checks on the basis of the objects' identifiers if an object to be imported already exists in the system. In this case, it does not import the object, but applies the following import strategy that depends on the type and status of the object.

- For **property types**:
  - The system aborts the import if the property type to be imported already exists but differs with respect to its attributes or content.
  - The system aborts the import if the property type to be imported references an FSM or a unit of measure that does not exist in the system.
  - The system skips the import if the property type to be imported already exists and has identical attributes and content.
- For **equipment graphs**:
  - The system aborts the import if the graph already exists in the system, is in a read-only status such as **Verification** or **Approved**, and differs from the graph to be imported.
  - The system updates an existing graph by overwriting its attributes, property types, statuses, triggers, and transitions with the imported data if the graph is in an editable status such as **Draft** and differs with respect to its data.
  - The system skips the import if the graph to be imported already exists and has identical attributes, property types, statuses, triggers, and transitions.

- For **equipment entities**:
  - The system aborts the import if the entity already exists in the system, is in a read-only status such as **Verification** or **Approved**, and differs from the entity to be imported.
  - The system aborts the import if the entity to be imported has a property type of the **WorkCenterAssignment** data type assigned to it, which references a work center that does not exist in the system.
  - The system aborts the import if one of the entity's assigned objects (property type, graph) already exists in the system, is in a read-only status such as **Verification** or **Approved**, and differs from the object to be imported.
  - The system updates an existing entity by overwriting its attributes and properties with the imported data if the entity is in an editable status such as **Draft** and differs with respect to its data.

  > TIP
  > Please note that the system does not update the runtime property values and the equipment graph-controlled status and expiry date of an existing entity.

  - The system skips the import if the entity to be imported already exists and has identical attributes and properties.
  - If an entity to be imported does not yet exist in the system and holds an assignment to a graph, the system sets it to the initial status of the graph.
  - If an entity to be imported holds an assignment to a class that does not exist in the system, the assignment cannot be established and is consequently removed from the entity. The import results log contains a warning to this effect.
- For **template equipment entities**:
  - The system aborts the import if the template entity already exists in the system, is in a read-only status such as **Verification** or **Approved**, and differs from the template entity to be imported.
  - The system aborts the import if the template entity to be imported has a property type of the **WorkCenterAssignment** data type assigned to it, which references a work center that does not exist in the system.
  - The system aborts the import if one of the template entity's assigned objects (property type, graph) already exists in the system, is in a read-only status such as **Verification** or **Approved**, and differs from the object to be imported.
  - The system updates an existing template entity by overwriting its attributes and properties with the imported data if the template entity is in an editable status such as **Draft** and differs with respect to its data.

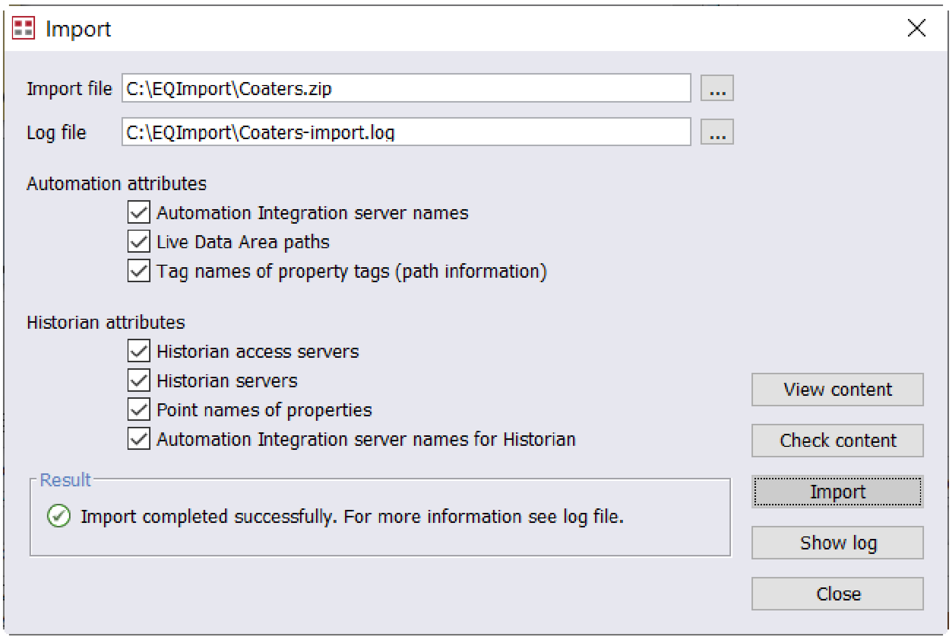
    - The system skips the import if the template entity to be imported already exists and has identical attributes and properties.
    - If a template entity to be imported does not yet exist in the system and holds an assignment to a graph, the system sets it to the initial status of the graph.
    - If a template entity to be imported holds an assignment to a class that does not exist in the system, the assignment cannot be established and is consequently removed from the template entity. The import results log contains a warning to this effect.
    - If a template entity to be imported holds an assignment to a label layout that does not exist in the system, the template entity is imported without the label layout assignment. The import results log contains a warning to this effect.
- For **equipment classes**:
    - The system aborts the import if the class already exists in the system, is in a read-only status such as **Verification** or **Approved**, and differs from the class to be imported.
    - The system aborts the import if one of its assigned objects (property type, graph, or entity) already exists in the system, is in a read-only status such as **Verification** or **Approved**, and differs from the object to be imported.
    - The system updates an existing class by overwriting its attributes and properties with the imported data if the class is in an editable status such as **Draft** and differs with respect to its data.

The system expects a ZIP archive as import file, which contains one BML file per object to be imported and an HTM file for meta data of the import package. It creates a log file that lists the individual actions performed during the import.

To perform an import, proceed as follows:

1. From the **File** menu, select **Import**.
   The system displays the **Import** dialog.
2. Click the ellipsis button (**...**) to select the ZIP file of the object you wish to import.
   Once you have selected an import file, the system automatically pre-fills the log file box with the same directory and file name but appends **-import** to the file name. The log file has **.log** as file extension.
3. If the import you are about to make includes entities that hold **Automation** or **Historian** properties, you can define to which extent their data is to be imported. Select the corresponding options to import data such as server names, Live Data area paths, tag paths, or point names defined for the individual properties.
4. To display the **Import Content** dialog that contains a list of the identifiers of the objects contained in the import file, sorted by their object types, click the **View content** button.

5. To perform suitability checks on the objects contained in the import file and view the check results in the **Import Checks** dialog, click the **Check content** button.
6. When you have verified that you can import the objects without conflicts, click the **Import** button to run the import process. The system displays the overall result of the import in the **Result** box.
   In addition, to view the complete log that lists all steps performed during the import, click the **Show log** button.
7. To close the dialog, click the **Close** button.



Import file: C:\EQImport\Coaters.zip

Log file: C:\EQImport\Coaters-import.log

Automation attributes

☑ Automation Integration server names
☑ Live Data Area paths
☑ Tag names of property tags (path information)

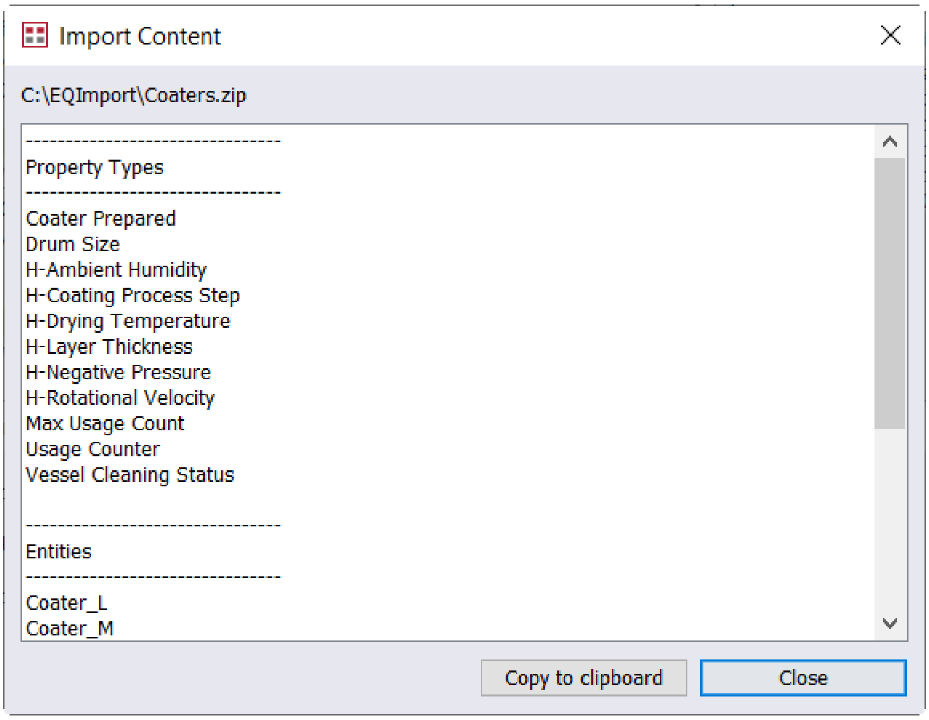
Historian attributes

☑ Historian access servers
☑ Historian servers
☑ Point names of properties
☑ Automation Integration server names for Historian

Result

✓ Import completed successfully. For more information see log file.

Figure 91: Import dialog

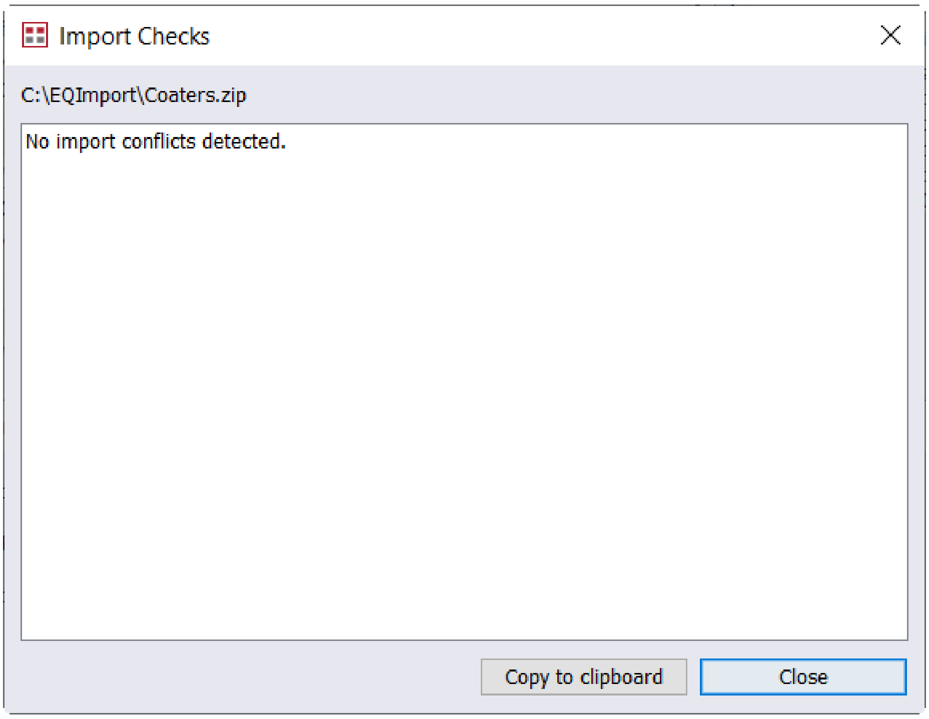


C:\EQImport\Coaters.zip

```
--------------------------------
Property Types
--------------------------------
Coater Prepared
Drum Size
H-Ambient Humidity
H-Coating Process Step
H-Drying Temperature
H-Layer Thickness
H-Negative Pressure
H-Rotational Velocity
Max Usage Count
Usage Counter
Vessel Cleaning Status

--------------------------------
Entities
--------------------------------
Coater_L
Coater_M
```

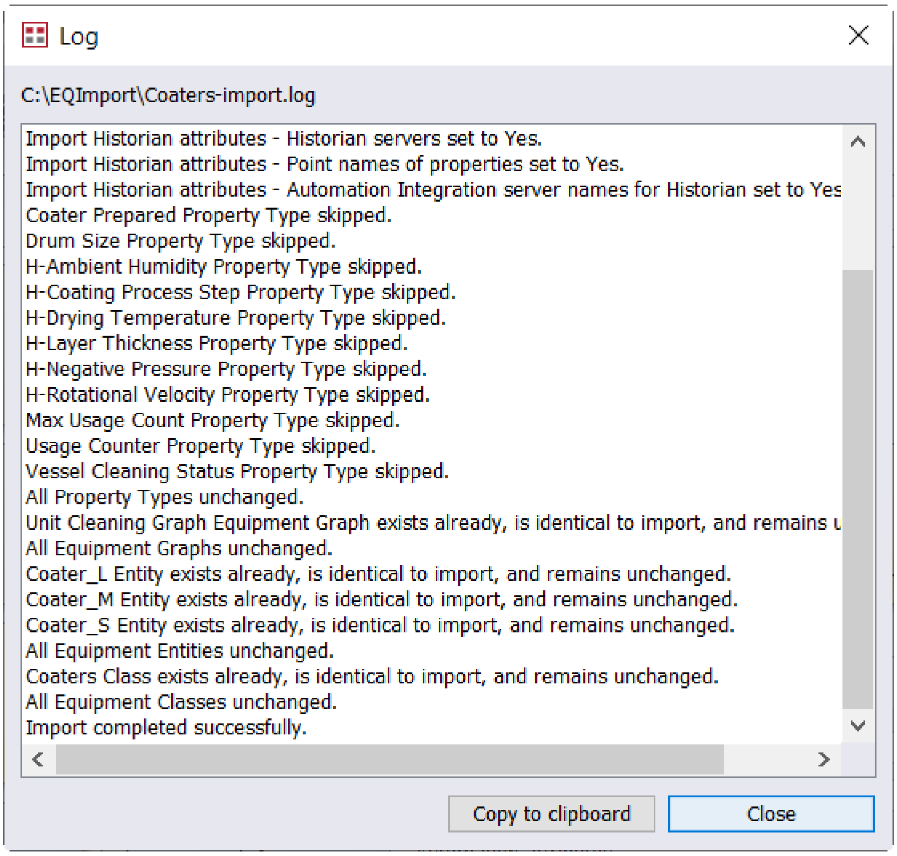
Figure 92: Import Content dialog



C:\EQImport\Coaters.zip

```
No import conflicts detected.
```

Figure 93: Import Checks dialog



```
Import Historian attributes - Historian servers set to Yes.
Import Historian attributes - Point names of properties set to Yes.
Import Historian attributes - Automation Integration server names for Historian set to Yes
Coater Prepared Property Type skipped.
Drum Size Property Type skipped.
H-Ambient Humidity Property Type skipped.
H-Coating Process Step Property Type skipped.
H-Drying Temperature Property Type skipped.
H-Layer Thickness Property Type skipped.
H-Negative Pressure Property Type skipped.
H-Rotational Velocity Property Type skipped.
Max Usage Count Property Type skipped.
Usage Counter Property Type skipped.
Vessel Cleaning Status Property Type skipped.
All Property Types unchanged.
Unit Cleaning Graph Equipment Graph exists already, is identical to import, and remains u
All Equipment Graphs unchanged.
Coater_L Entity exists already, is identical to import, and remains unchanged.
Coater_M Entity exists already, is identical to import, and remains unchanged.
Coater_S Entity exists already, is identical to import, and remains unchanged.
All Equipment Entities unchanged.
Coaters Class exists already, is identical to import, and remains unchanged.
All Equipment Classes unchanged.
Import completed successfully.
```

Figure 94: Log dialog of import

TIP

If you wish to retain the information listed in the **Import Content**, **Import Checks**, or **Log** dialogs, click the **Copy to clipboard** button to copy the lists and then paste them into a system-external text editor.

IMPORTING TAGS/POINTS

The **Import tags/points** function allows you to import tag and point data for **Automation** and **Historian** properties that were either previously exported from another PharmaSuite system with the **Export tags/points** function (page 107) of Data Manager or have an equivalent format. It checks the tag/point data to be imported against the data that is already available in your system to determine potential data conflicts and resolves them by skipping or updating tag/point data.

The system skips tag/point data of

- entities that do not exist in the target system and of
- properties that are not defined for a given entity in the target system.

The system updates all non-skipped tag/point data by overwriting the existing data with the imported data. Identical data remains unchanged.

The system expects an XML file as import file that contains tag/point data for entities included in the import. It creates a log file that lists the individual actions performed during the import. A summary at the end of the log file, indicates the import results for each entity.

To perform an import, proceed as follows:

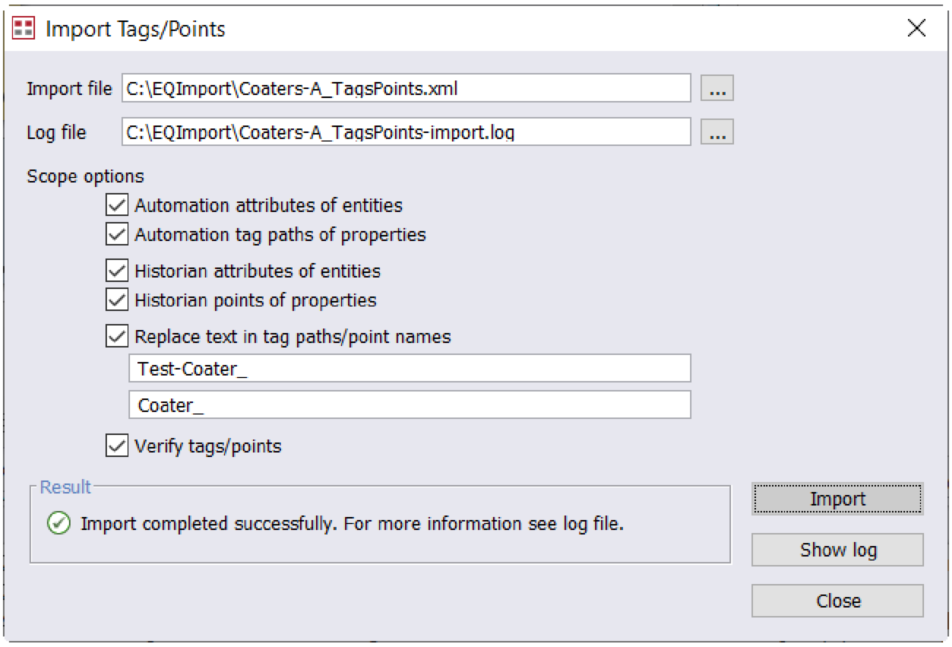
1. From the **File** menu, select **Import tags/points**.
   The system displays the **Import Tags/Points** dialog.
2. Click the ellipsis button (**...**) to select the XML file you wish to import.
   Once you have selected an import file, the system automatically pre-fills the log file box with the same directory and file name but appends **-import** to the file name. The log file has **.log** as file extension.
3. You can define to which extent the **Automation** and **Historian** data is to be imported. Select the corresponding options to import
   - the **Automation** attributes defined on the **Basic** tab of an entity,
   - the tags defined with the properties of an entity
   - and **Historian** attributes defined on the **Basic** tab of an entity,
   - the points defined with the properties of an entity.
4. If you need to update the individual tag paths and point names as you import data from a test system into a production system, you can define characters to be replaced in the imported tag paths and point names. If you just enter a string in the **Search for** box and leave the **Replace with** box blank, the system deletes the search string from the tag paths and point names.

   TIP
   Please note that the system performs the replace operation on all occurrences of the given search string in all selected tag paths and point names. So, if you need to differentiate the replace strings between **Automation** and **Historian** data or attributes and tag paths/point names, you can run the import several times, and only select the respective options for each run.
5. Select the **Verify tags/points** option to verify all imported tag paths and point names and see the verification result in the log file.

   TIP
   If you know that the tag path/point name verification cannot be successful, since the data on the Automation Integration or the Historian server has not been updated yet, uncheck the option and perform the verification later with the **Verify tags/points** function (page 118) from the File menu.

6. When you have defined the scope options, click the **Import** button to run the import process. The system displays the overall result of the import in the **Result** box.
   In addition, to view the complete log that lists all steps performed during the import, click the **Show log** button.

7. To close the dialog, click the **Close** button.



Import Tags/Points

Import file C:\EQImport\Coaters-A_TagsPoints.xml

Log file C:\EQImport\Coaters-A_TagsPoints-import.log

Scope options

☑ Automation attributes of entities
☑ Automation tag paths of properties
☑ Historian attributes of entities
☑ Historian points of properties
☑ Replace text in tag paths/point names

Test-Coater_

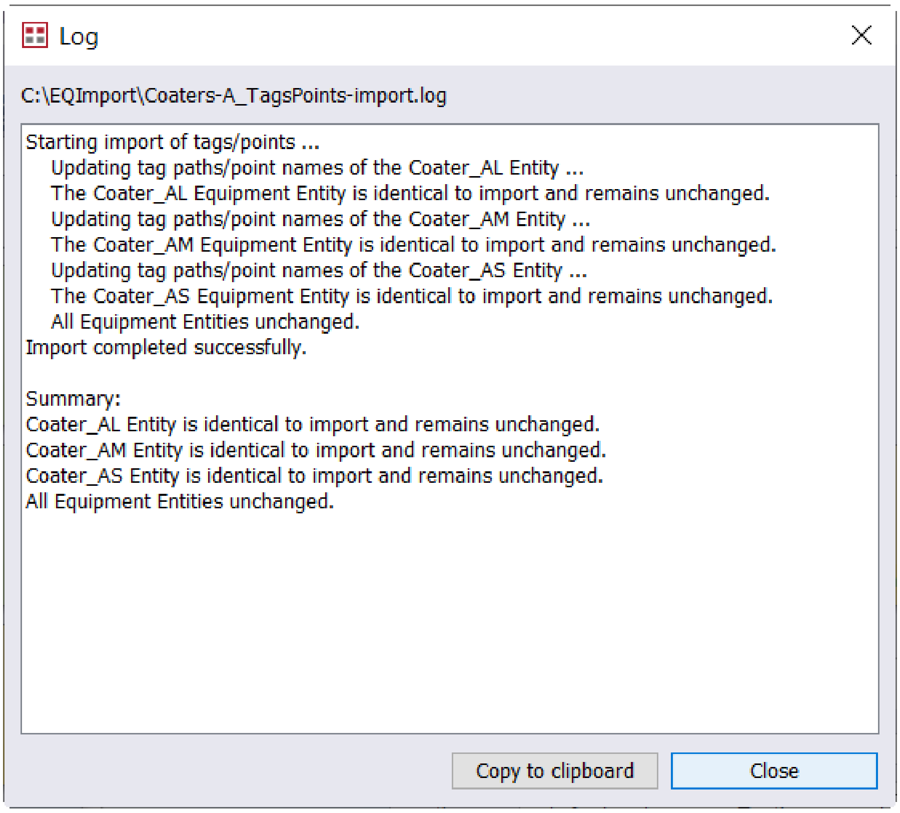
Coater_

☑ Verify tags/points

Result

✓ Import completed successfully. For more information see log file.

Figure 95: Import Automation Tags dialog



C:\EQImport\Coaters-A_TagsPoints-import.log

```
Starting import of tags/points ...
    Updating tag paths/point names of the Coater_AL Entity ...
    The Coater_AL Equipment Entity is identical to import and remains unchanged.
    Updating tag paths/point names of the Coater_AM Entity ...
    The Coater_AM Equipment Entity is identical to import and remains unchanged.
    Updating tag paths/point names of the Coater_AS Entity ...
    The Coater_AS Equipment Entity is identical to import and remains unchanged.
    All Equipment Entities unchanged.
Import completed successfully.

Summary:
Coater_AL Entity is identical to import and remains unchanged.
Coater_AM Entity is identical to import and remains unchanged.
Coater_AS Entity is identical to import and remains unchanged.
All Equipment Entities unchanged.
```

Figure 96: Log dialog of automation tag import

TIP

If you wish to retain the information listed in the **Log** dialog, click the **Copy to clipboard** button to copy the list and then paste it into a system-external text editor.

Verifying Tags/Points

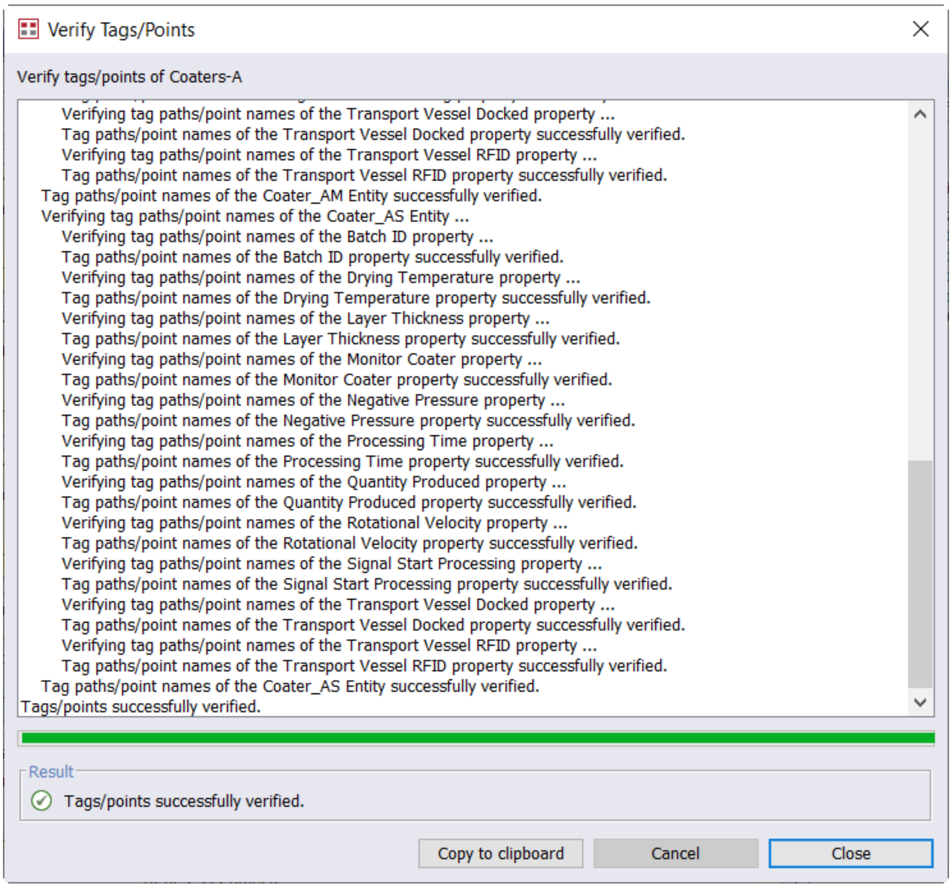
The **Verify tags/points** function allows you to verify the tag and point data for all **Automation** and **Historian** properties of an entity or of all entities of a class instead of having to run the verification individually for each property from its respective editor.

To perform a verification, proceed as follows:

1. In the Search window, locate the object whose tags and points you wish to verify. Right-click it to open the object's shortcut menu and select the **Verify tags/points of <object identifier>** function or double-click to open the object and then, from the **File** menu, select **Verify tags/points of <object identifier>**.
   The system displays the **Verify Tags/Points** dialog and immediately runs the verification.

2. While running the verification, the system lists all steps it performs. At the same time, the progress bar below the list indicates the status of the verification. Once the verification run is complete, the last line of the list displays the overall result.

3. To close the dialog, click the **Close** button.



```
        Verifying tag paths/point names of the Transport Vessel Docked property ...
        Tag paths/point names of the Transport Vessel Docked property successfully verified.
        Verifying tag paths/point names of the Transport Vessel RFID property ...
        Tag paths/point names of the Transport Vessel RFID property successfully verified.
    Tag paths/point names of the Coater_AM Entity successfully verified.
    Verifying tag paths/point names of the Coater_AS Entity ...
        Verifying tag paths/point names of the Batch ID property ...
        Tag paths/point names of the Batch ID property successfully verified.
        Verifying tag paths/point names of the Drying Temperature property ...
        Tag paths/point names of the Drying Temperature property successfully verified.
        Verifying tag paths/point names of the Layer Thickness property ...
        Tag paths/point names of the Layer Thickness property successfully verified.
        Verifying tag paths/point names of the Monitor Coater property ...
        Tag paths/point names of the Monitor Coater property successfully verified.
        Verifying tag paths/point names of the Negative Pressure property ...
        Tag paths/point names of the Negative Pressure property successfully verified.
        Verifying tag paths/point names of the Processing Time property ...
        Tag paths/point names of the Processing Time property successfully verified.
        Verifying tag paths/point names of the Quantity Produced property ...
        Tag paths/point names of the Quantity Produced property successfully verified.
        Verifying tag paths/point names of the Rotational Velocity property ...
        Tag paths/point names of the Rotational Velocity property successfully verified.
        Verifying tag paths/point names of the Signal Start Processing property ...
        Tag paths/point names of the Signal Start Processing property successfully verified.
        Verifying tag paths/point names of the Transport Vessel Docked property ...
        Tag paths/point names of the Transport Vessel Docked property successfully verified.
        Verifying tag paths/point names of the Transport Vessel RFID property ...
        Tag paths/point names of the Transport Vessel RFID property successfully verified.
    Tag paths/point names of the Coater_AS Entity successfully verified.
Tags/points successfully verified.
```

Result

✓ Tags/points successfully verified.

Figure 97: Verify Automation Tags dialog

TIP

If you wish to retain the listed verification information, click the **Copy to clipboard** button to copy the list and then paste it into a system-external text editor.

Chapter 2

Managing Equipment

Data Manager provides a comprehensive framework for creating, configuring, and managing all data you need to maintain the equipment in use at your site and to define the equipment requirements for master recipes and workflows.

Data Manager - Equipment can be used in two different modes. Data Manager - Equipment - Smart Search is the recommended mode for most usage scenarios and provides the full potential of the system's flexible search mechanisms. In this mode and depending on its configuration, the system ignores entities that have been generated on the shop floor and are in the **Archived** status (page 129) or all generated entities regardless of their status.
Data Manager - Equipment - Basic Search is recommended for usage scenarios when specific objects whose identifiers are known need to be located within a very large number of equipment objects. A scenario of this kind can evolve when large numbers of equipment entities are generated for one-time use on the shop floor, followed by their immediate archival.

In both modes, the system supports you with providing a set of default objects (classes (page 157), graphs (page 189), and property types (page 199) in Smart Search and classes (page 221), graphs (page 251), and property types (page 262) in Basic Search).

What Is an Equipment Requirement?

An **equipment requirement** specifies the needs an equipment entity must provide for a specific process step.

Equipment requirements in PharmaSuite are defined and maintained in Recipe and Workflow Designer during recipe and workflow configuration.

What Is an Equipment Class?

An **equipment class** provides a means to bundle and address equipment entities that share a common set of properties and capabilities.

Equipment classes are maintained as data objects (page 156) in Data Manager.

What Is a Template Equipment Entity?

A **template equipment entity** is the data representation of a piece of equipment that is used during processing as a master to generate new equipment entities and required for pieces of equipment such as filter liners, hoses, or plastic bags, which are used as equipment during processing and thus need to be tracked. The template entity can be assigned to a material in order to specify the properties that generated equipment entities of the material will have. Template entities without a material reference can be used for components that are not inventoried.

Template entities to be used for generating material-related equipment entities during production require specific object assignments:

Object type	Usage	Data type Content	Purpose
Property type	Specification	EquipmentType Hybrid (RS)	---
Property type	Runtime	String	Current Sublot (RS)
Property type	Runtime	String	Base Sublot (RS)

When a phase building block generates an entity during identification of a sublot, the system determines the suitable template entity on account of the template entity assigned to the sublot's material.
When the identified sublot is split, but continues to exist, the system fills the new entity's **Current Sublot (RS)** runtime property with the identifier of the sublot that is generated along with the entity. The **Base Sublot (RS)** runtime property is filled with the identifier of the sublot that has been split in order to generate the sublot of the **Current Sublot (RS)** runtime property.
When the identified sublot is consumed immediately and thus ceases to exist, its identifier is shown in the **Base Sublot (RS)** runtime property of the generated entity. The **Current Sublot (RS)** runtime property of the entity remains blank in this case.

What Is an Equipment Entity?

An **equipment entity** is the data representation of a physical piece of equipment and its properties. Equipment entities can be derived from an equipment class, a template equipment entity, or from another equipment entity. They can be assigned to all classes whose properties they share.
Additionally, equipment entities can be collected in groups to facilitate scenarios such as a common treatment of otherwise unconnected entities, e.g. sterilization, or a complex piece of equipment that consists of a base entity and several other attached entities, e.g. an assembled machine. A group of entities consists of a parent entity and one or more child entities, which themselves can also be parent to an entity group.

The following types of equipment (page 205) entities require a specific set of properties to be fully configured and suitable for use with the Weighing and Dispense phases of PharmaSuite:

- Containers
  Container is a generic term to cover the various types of equipment for holding materials between processing steps. Typical containers used in pharmaceutical production environments are bins or IBCs.
  A container equipment entity needs to have specific objects assigned to it:

Object type	Usage	Data type Content	Purpose
Property type, such as Container Type (RS)	Specification	EquipmentType Container (RS)	---
Property type, such as Container Tare (RS)	Runtime	MeasuredValue	Current Tare (RS)
Property type, such as Container Sublot (RS)	Runtime	String, used in container cleaning graph	Current Sublot (RS)
Property type, such as Container Clean Shelflife (RS)	Specification	Duration, used in container cleaning graph	---
Equipment graph, such as ContainerCleaning_RS_1	---	---	Container Cleaning (RS)

TIP

To define a reference tare for a container that can be accessed when taring the container during Dispense or Output Weighing, create a MeasuredValue runtime property type of the Reference Tare (RS) purpose.

- Rooms

  A **room** is a physical location, at which a step of an order execution is performed, such as dispensing, mixing, packaging, etc.

  A room equipment entity needs to have specific objects assigned to it:

Object type	Usage	Data type Content	Purpose
Property type, such as Room Type (RS)	Specification	EquipmentType Room (RS)	---
Property type, such as Room Cleaning Rules (RS)	Runtime	RoomCleaningRules <room-specific>	---
Property type, such as Work Center Assignment (RS)	Specification	WorkCenterAssignment <room-specific>	---
Property type, such as General Clean Shelflife (RS)	Specification	Duration, used in room cleaning graph	---
Equipment graph, such as RoomCleaning_RS_1	---	---	Room Cleaning (RS)

- Scales

  A **scale** is a type of equipment used to measure the weight of a material or an object.

  A scale equipment entity needs to have specific objects assigned to it:

Object type	Usage	Data type Content	Purpose
Property type, such as Scale Type (RS)	Specification	EquipmentType Scale (RS)	---
Property type, such as Scale Ranges (RS)	Specification	Ranges <scale-specific>	ScaleRanges (RS)
Property type, such as Scale Configuration (RS)	Specification	ScaleConfiguration <scale-specific>	---
Property type, such as Scale Test and Calibration (RS)	Specification	ScaleTestAndCalibration <scale-specific>	---

Object type	Usage	Data type Content	Purpose
Property type, such as Work Center Assignment (RS)	Specification	WorkCenterAssignment <scale-specific>	---
Property type, such as Scale Test Shelflife (RS)	Specification	Duration, used in scale test graph	---
Property type, such as Scale Calibration Shelflife (RS)	Specification	Duration, used in scale calibration graph	---
Property type, such as Scale Load (RS)	Runtime	String, required for handling of loaded scales	Current Load (RS)
Equipment graph, such as ScaleTest_RS_1	---	---	Scale Test (RS)
Equipment graph, such as ScaleCalibration_RS_1	---	---	Scale Calibration (RS)

TIP

Unless an entity has all required objects assigned, it cannot be moved to a read-only status (page 129), such as **Verification** or **Approved**.

If PharmaSuite is configured to communicate with Warehouse Management, equipment entities that are movable and hold a sublot are typically represented as a logistic unit in Warehouse Management.
Since equipment entities in PharmaSuite and logistic units in Warehouse Management are different objects, the following prerequisites need to be considered when data is transferred between both applications:

- A logistic unit to be used in PharmaSuite as equipment entity must be created as permanent logistic unit in Warehouse Management.
- The identifiers of an equipment entity in PharmaSuite and its corresponding logistic unit in Warehouse Management must be identical. The same applies to the barcodes of both objects.
- A logistic unit that corresponds to a PharmaSuite-managed container must be created as a logistic unit of the **Transport only** category in Warehouse Management. Warehouse Management will not trigger graph transitions on the container expected by PharmaSuite and will not set equipment properties when performing load or unload operations.

- In Warehouse Management, the **Load control** category of logistic units must be used to trigger PharmaSuite graph transitions on the corresponding equipment entities at first load and total unload operations.
- A storage location in PharmaSuite that holds sublots must be created in Warehouse Management as a storage unit. The identifier of the storage location and the name of the storage unit must be identical.

A logistic unit of the **Load control** category expects an equipment graph with the following purpose and triggers to be assigned to its corresponding equipment entity:

- Purpose: Warehouse Load Check (RS)
- Trigger (key): FIRST_LOAD, TOTAL_UNLOAD

What Is an Equipment Graph?

An **equipment graph** specifies the statuses an equipment entity can assume while it is used in execution, the triggers that affect the entity's status, as well as the transitions between the statuses, their conditions, and actions.
A typical example of an equipment graph is a cleaning graph with statuses such as "Clean" and "Unclean" and triggers such as "Entity is bound to process" and "Entity is unbound from process".

Equipment graphs are maintained as data objects (page 188) in Data Manager.

What Is a Property Type?

A **property type** specifies equipment properties and equipment property requirements.

Property types are maintained as data objects (page 199) in Data Manager.

Status Handling of Equipment

Equipment classes, template entities, entities, and graphs are maintained in Data Manager throughout their entire life cycle and follow an inherent status graph that determines if they are approved for use.

In addition to the inherent status graph, which states that an equipment object's attributes and properties are complete and correct, there may be further dynamic statuses that affect the equipment object's suitability for use, such as a cleaning status or a maintenance status.
Statuses of this kind are defined and maintained in equipment graph (page 188) objects, which you can assign to classes and entities. Equipment graph objects themselves are also controlled by an inherent status graph.

Equipment Class Statuses

During their life cycle, equipment classes can assume the following statuses:

- **Draft**: The equipment class has been created and saved. It can be edited.

  > TIP
  >
  > When you create an assignment (page 100) between an object in **Draft** and another object that is already in a read-only status such as **Verification**, **Approved**, or even **Archived**, only one assignment direction is available, namely assigning to the object in **Draft**. The **Assign...** function of an object is disabled when its assignment target is in **Verification** or higher.

- **Verification**: The equipment class is ready for review. It is not editable anymore. If updates are required, its status needs to be changed back to **Draft**.
  Under specific circumstances, such as equipment qualification procedures on the shop floor, it is possible to allow the usage of classes that are in the **Verification** status.
- **Approved**: The equipment class is ready for use as equipment requirement. Those of its assigned entities that are also **Approved** (or have another suitable status) can be identified and used for production on the shop floor. It is not editable but can return to **Draft** if it requires to be updated.
- **Archived**: The equipment class is not intended to be used as equipment requirement anymore. For this reason, it is no longer available in the Universe of Recipe and Workflow Designer. This is the final status of an equipment class.
  It is, however, still possible to use equipment entities of an **Archived** class for production on the shop floor, provided the entities themselves have a suitable status.

  > TIP
  >
  > Please note that once an equipment class has been used as requirement in a master recipe or master workflow it cannot be deleted anymore.

Depending on the system's configuration, a status change can require a user to enter a single or double electronic signature as proof of authorization.

Template Equipment Entity Statuses

During their life cycle, template equipment entities can assume the following statuses:

- **Draft**: The template equipment entity has been created and saved. It can be edited and assigned to an equipment class.

  TIP
  When you create an assignment (page 100) between an object in **Draft** and another object that is already in a read-only status such as **Verification**, **Approved**, or even **Archived**, only one assignment direction is available, namely assigning to the object in **Draft**. The **Assign...** function of an object is disabled when its assignment target is in **Verification** or higher.

- **Verification**: The template equipment entity is ready for review. It is not editable anymore. If updates are required, its status needs to be changed back to **Draft**. A template entity can only be moved to **Verification**, if all of its assigned graphs are also at least in **Verification**.

  TIP
  Under specific circumstances, such as equipment qualification procedures, a template equipment entity in a status lower than **Approved**, like **Draft** or **Verification**, may be used for generating equipment entities on the shop floor. The entities are then also not **Approved** but are generated in the same status as their template entity.

- **Approved**: The template equipment entity is ready for generating equipment entities during production on the shop floor. It is not editable but can return to **Draft** if it requires to be updated.

  TIP
  An equipment entity generated on the shop floor from an **Approved** template entity is also **Approved** and immediately ready for unrestricted use.

- **Archived**: The template equipment entity has been retired and is no longer available for generating entities. This is the final status of a template equipment entity.

  TIP
  Please note that once a template equipment entity has been used for generating entities on the shop floor, it cannot be deleted anymore.

Depending on the system's configuration, a status change can require a user to enter a single or double electronic signature as proof of authorization.

Equipment Entity Statuses

During their life cycle, equipment entities can assume the following statuses:

- **Draft**: The equipment entity has been created and saved. It can be edited and assigned to an equipment class.

  TIP
  When you create an assignment (page 100) between an object in **Draft** and another object that is already in a read-only status such as **Verification**, **Approved**, or even **Archived**, only one assignment direction is available, namely assigning to the object in **Draft**. The **Assign...** function of an object is disabled when its assignment target is in **Verification** or higher.

- **Verification**: The equipment entity is ready for review. It is not editable anymore. If updates are required, its status needs to be changed back to **Draft**.
  This, however, does not apply to updates of runtime data, which reflects the actual activities that are performed on the shop floor and can change whenever an equipment entity is bound and used for processing. In order to resolve processing exceptions by administrative means, it is possible to update runtime data in this status in Data Manager as well.
  An entity can only be moved to **Verification**, if all of its assigned graphs are also at least in **Verification**.

  TIP
  Under specific circumstances, such as equipment qualification procedures on the shop floor, it is possible to allow the identification and use of entities that are in the **Verification** status. Even though it would be possible to move such entities back to **Draft** while they are being used and are bound to an order or workflow step, this change could compromise the quality of the process and should thus be strictly avoided.

- **Approved**: The equipment entity is ready for use during production on the shop floor, provided its equipment class required by the master recipe or master workflow is **Approved** (or has another suitable status). It is not editable but can return to **Draft** if it requires to be updated.
  This, however, does not apply to updates of runtime data, which reflects the actual activities that are performed on the shop floor and can change whenever an equipment entity is bound and used for processing. In order to resolve processing exceptions by administrative means, it is possible to update runtime data in this status in Data Manager as well.

- **Archived**: The equipment entity has been retired and is no longer available for regular use. This is the final status of an equipment entity.

  TIP
  Please note that once an equipment entity has been used for processing and has been bound to an order or workflow, it cannot be deleted anymore.

Depending on the system's configuration, a status change can require a user to enter a single or double electronic signature as proof of authorization.

Equipment Graph Statuses

During their life cycle, equipment graphs can assume the following statuses:

- **Draft**: The equipment graph has been created and saved. It can be edited.

  TIP
  When you create an assignment (page 100) between an object in **Draft** and another object that is already in a read-only status such as **Verification**, **Approved**, or even **Archived**, only one assignment direction is available, namely assigning to the object in **Draft**. The **Assign...** function of an object is disabled when its assignment target is in **Verification** or higher.

- **Verification**: The equipment graph is ready for review. It is not editable anymore. If updates are required, its status needs to be changed back to **Draft**.
  A graph can only be moved back from **Verification** to **Draft**, if all entities to which it is assigned are also in **Draft**.
- **Approved**: The equipment graph is ready to control the statuses of equipment entities on the shop floor.
- **Archived**: The equipment graph is not intended to be used for status control of equipment entities any longer. This is the final status of an equipment graph.

Depending on the system's configuration, a status change can require a user to enter a single or double electronic signature as proof of authorization.

Changing the Status of Equipment

To change the status of an equipment class, template entity, entity, or graph, proceed as follows:

1. From the **File** menu, select **Change status of <object identifier>,** click the **Change status** button (page 71) in the File toolbar, or right-click an equipment class or entity to open its shortcut menu and select **Change status of <object identifier>**.
   The system opens the **Change status** dialog for the equipment class or entity. It displays the identifier, status, and short description of the equipment.

2. From the option list, select the **Action** you wish to perform on the equipment. The status to which the action will move the class, template entity, entity, or graph is indicated in parentheses.
   The system displays mismatch issues that exist for the object in the **Mismatch Details**. Issues that cannot be resolved anymore after the status change has been performed are listed as errors and prevent the status change. Other issues are listed as warnings, and you can perform the status change.

   > TIP
   > Please be aware that even if a mismatch issue is listed as warning, you need to make sure that performing the status change will not prevent resolving the issues at the object where the resolution should take place.

3. Click the **OK** button.
   The system displays the **Electronic Signature** dialog. The **Description** box displays the status change you are about to perform.
4. Execute the electronic signature (page 101).
   When the status change has completed successfully, the **Change Status** dialog closes automatically.
   To view all status changes performed on the equipment, open the **Status History** tab, which is available in an object's Details window in both modes of Data Manager - Equipment, **Smart Search** (classes (page 162), template entities (page 171), entities (page 187), and graphs (page 198)) and **Basic Search** (classes (page 227), template entities (page 235), entities (page 250), and graphs (page 261)).

   > TIP
   > Please note that you cannot change that status of equipment that is currently bound by processing on the shop floor.

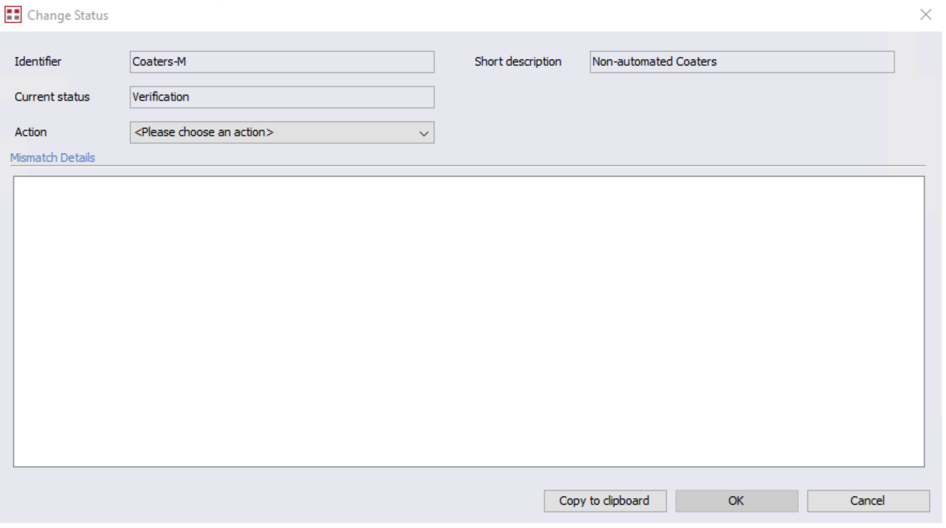
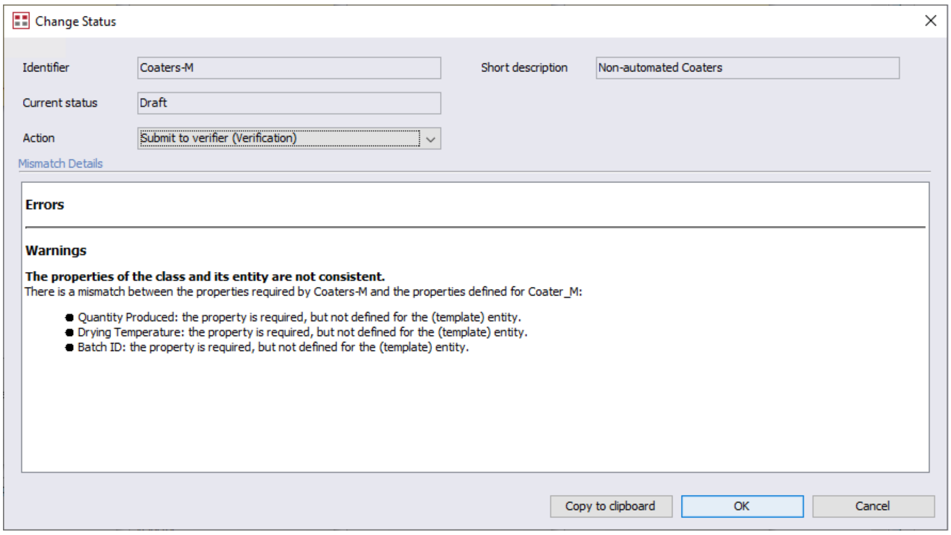


Figure 98: Change Status dialog



Identifier: Coaters-M

Short description: Non-automated Coaters

Current status: Draft

Action: Submit to verifier (Verification)

Mismatch Details

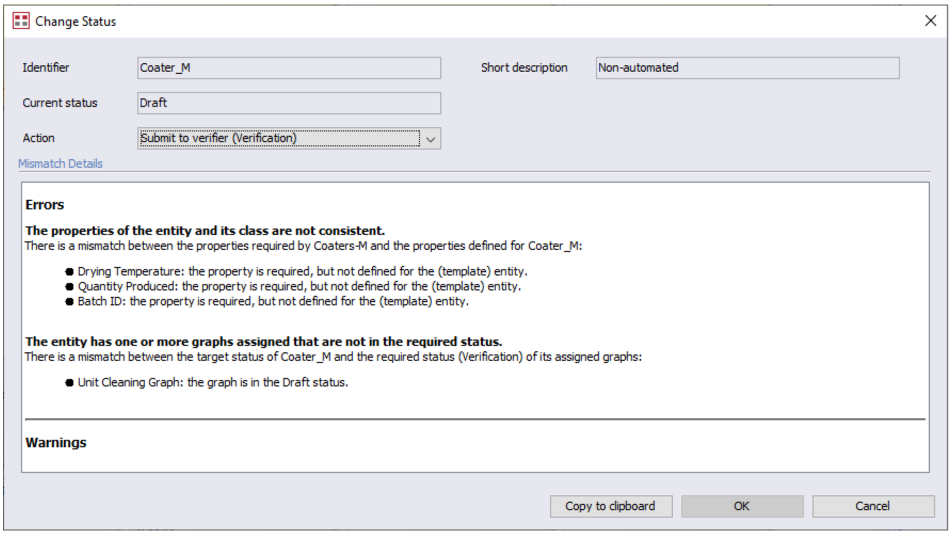
Errors

Warnings

The properties of the class and its entity are not consistent.
There is a mismatch between the properties required by Coaters-M and the properties defined for Coater_M:

- Quantity Produced: the property is required, but not defined for the (template) entity.
- Drying Temperature: the property is required, but not defined for the (template) entity.
- Batch ID: the property is required, but not defined for the (template) entity.

Figure 99: Change Status dialog with warnings



Identifier: Coater_M

Short description: Non-automated

Current status: Draft

Action: Submit to verifier (Verification)

Mismatch Details

Errors

The properties of the entity and its class are not consistent.
There is a mismatch between the properties required by Coaters-M and the properties defined for Coater_M:

- Drying Temperature: the property is required, but not defined for the (template) entity.
- Quantity Produced: the property is required, but not defined for the (template) entity.
- Batch ID: the property is required, but not defined for the (template) entity.

The entity has one or more graphs assigned that are not in the required status.
There is a mismatch between the target status of Coater_M and the required status (Verification) of its assigned graphs:

- Unit Cleaning Graph: the graph is in the Draft status.

Warnings

Figure 100: Change Status dialog with errors

Mass Change Operations on (Template) Entities

In the course of its life cycle, an equipment entity can change its characteristics, which are defined by its properties or its equipment graphs. Changes to an entity's characteristics can be caused by circumstances such as

- a sensor is added to an entity and requires a new property type to be assigned to the entity.
- an additional counter or expiry date needs to be maintained for the entity and requires a new status graph with corresponding property types to be assigned to the entity.

If the required changes do not only affect a single entity but all entities of an equipment class, you can perform the required changes on the class and select the **Update (template) entities** function from the **File** menu or the shortcut menu of the class.

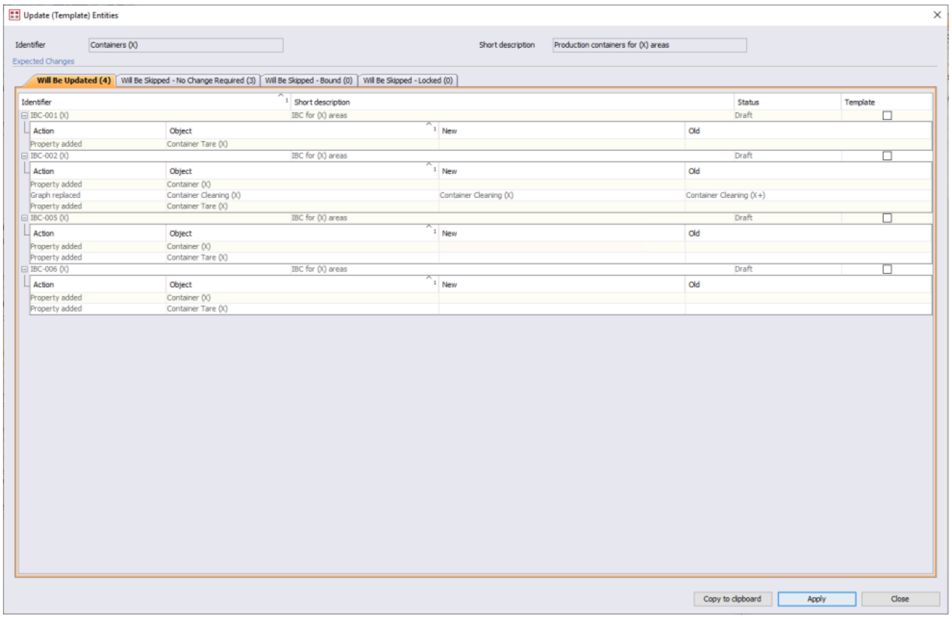
TIP

Please note that the **Update (template) entities** function applies changes to all (template) entities that are in a status other than **Archived**. For **Approved** entities, the system performs the updates directly on the entity, which means it overrides the read-only restriction of the **Approved** status and does not perform any status changes. To support you with assessing the impact of changes to the (template) entities of an **Approved** class, the system provides the **Compile usage list** function (page 141) for equipment classes.

Performing a mass change operation on the template entities and entities of a class happens in the following two steps. In the first step, the system calculates the expected changes and presents them for reviewing. In the second step a suitably authorized user triggers the system to perform the updates on the (template) entities.

CALCULATING EXPECTED CHANGES

From the **File** menu, select **Update (template) entities of <class identifier>** or right-click an equipment class to open its shortcut menu and select **Update (template) entities of <class identifier>**.
The system opens the **Update (Template) Entities** dialog for the equipment class and its entities. It displays the identifier and short description of the equipment class. While calculating the expected changes, it shows a progress bar. The calculated expected changes are then listed in four tabs.


Update (Template) Entities
Will Be Updated (4)
Will Be Skipped - No Change Required (3)
Will Be Skipped - Bound (0)
Will Be Skipped - Locked (0)
Copy to clipboard
Apply
Close

Identifier: Containers (X)

Short description: Production containers for (X) areas

Expected Changes

Identifier	Short description	Status	Template
IBC-001 (X)	IBC for (X) areas	Draft	□

Action	Object	New	Old
Property added	Container Tare (X)		

Identifier	Short description	Status	Template
IBC-002 (X)	IBC for (X) areas	Draft	□

Action	Object	New	Old
Property added	Container (X)		
Graph replaced	Container Cleaning (X)	Container Cleaning (X)	Container Cleaning (X+)
Property added	Container Tare (X)		

Identifier	Short description	Status	Template
IBC-005 (X)	IBC for (X) areas	Draft	□

Action	Object	New	Old
Property added	Container (X)		
Property added	Container Tare (X)		

Identifier	Short description	Status	Template
IBC-006 (X)	IBC for (X) areas	Draft	□

Action	Object	New	Old
Property added	Container (X)		
Property added	Container Tare (X)		

Figure 101: Update (Template) Entities - Expected Changes

To provide an overview of the expected changes, the header of each tab displays the number of listed (template) entities:

- The **Will Be Updated** tab lists all entities and template entities that the system will try to update.
  Each (template) entity is displayed as a nested table, with the common attributes (**Identifier**, **Short description**, **Status**, and **Template** indicator checkbox) as main table and in a collapsible sub-table each (template) entity's expected changes:
  - **Action**
    indicates which change needs to be performed.
  - **Object**
    displays the identifier of the affected object. In the case of replace actions, this is the old component.
  - **New**
    displays the new component (for replace actions) or the new value (for property value changes).
  - **Old**
    displays the old component (for replace actions) or the old value (for property value changes).

- The **Will Be Skipped - No Change Required** tab lists all entities and template entities that already contain all intended changes and thus do not require to be updated.
  Each (template) entity is displayed with its attributes (**Identifier**, **Short description**, **Status**, and **Template** indicator checkbox).

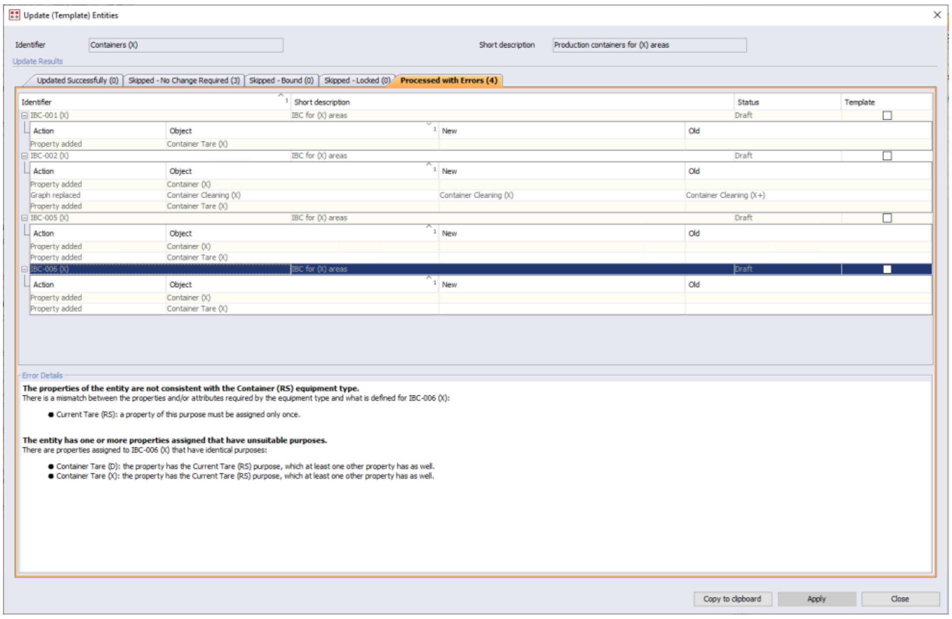
- The **Will Be Skipped - Bound** tab lists all entities that cannot be updated, since they are currently in use for processing.
  Each entity is displayed with its attributes (**Identifier**, **Short description**, **Status**, and **Template** indicator checkbox).
  Selecting an entity in the list displays further information in the **Details** section at the bottom of the dialog. It indicates the work center where the entity is currently being used and in which phase of the order or workflow that is being executed.

- The **Will Be Skipped - Locked** tab lists all entities and template entities that cannot be updated, since they are currently being edited by another user or have unsaved changes.
  Each (template) entity is displayed with its attributes (**Identifier**, **Short description**, **Status**, and **Template** indicator checkbox).
  Selecting an entity or template entity in the list displays further information in the **Details** section at the bottom of the dialog. If in use in another instance of Data Manager or another application, it indicates the login name of the user who is currently editing the (template) entity, the ID of the device where it is being edited, and the application that is used for editing. If opened with unsaved changes in your current instance of Data Manager, it indicates this situation.

TIP
To view the information of all tabs, click the **Copy to clipboard** button and paste the text into a system-external text editor.

UPDATING (TEMPLATE) ENTITIES

1. Click the **Apply** button to start the mass change.
   The system displays the **Electronic Signature** dialog.
2. Execute the electronic signature (page 101).
   The system shows a progress bar indicating the number of processed (template) entities. After the update run, it displays the update results in five tabs.



Identifier: Containers (X)
Short description: Production containers for (X) areas

Identifier	Short description	Status	Template
IBC-001 (X)	IBC for (X) areas	Draft	□
IBC-002 (X)	IBC for (X) areas	Draft	□
IBC-005 (X)	IBC for (X) areas	Draft	□
IBC-006 (X)	IBC for (X) areas	Draft	□

IBC-001 (X):

Action	Object	New	Old
Property added	Container Tare (X)		

IBC-002 (X):

Action	Object	New	Old
Property added	Container (X)		
Graph replaced	Container Cleaning (X)	Container Cleaning (X)	Container Cleaning (X+)
Property added	Container Tare (X)		

IBC-005 (X):

Action	Object	New	Old
Property added	Container (X)		
Property added	Container Tare (X)		

IBC-006 (X):

Action	Object	New	Old
Property added	Container (X)		
Property added	Container Tare (X)		

The properties of the entity are not consistent with the Container (RS) equipment type.
There is a mismatch between the properties and/or attributes required by the equipment type and what is defined for IBC-006 (X):

- Current Tare (RS): a property of this purpose must be assigned only once.

The entity has one or more properties assigned that have unsuitable purposes.
There are properties assigned to IBC-006 (X) that have identical purposes:

- Container Tare (D): the property has the Current Tare (RS) purpose, which at least one other property has as well.
- Container Tare (X): the property has the Current Tare (RS) purpose, which at least one other property has as well.

Figure 102: Update (Template) Entities - Update Results

To provide an overview of the update results, the header of each tab displays the number of listed (template) entities:

- The **Updated Successfully** tab lists all entities and template entities that the system has updated.
  Each (template) entity is displayed as a nested table, with the common attributes (**Identifier**, **Short description**, **Status**, and **Template** indicator checkbox) as main table and in a collapsible sub-table each (template) entity's performed changes:
  - **Action**
    indicates which change was performed.
  - **Object**
    displays the identifier of the affected object. In the case of replace actions, this is the old component.
  - **New**
    displays the new component (for replace actions) or the new value (for property value changes).

  - **Old**
    displays the old component (for replace actions) or the old value (for property value changes).

- The **Skipped - No Change Required** tab lists all entities and template entities that already contain all intended changes and thus did not require to be updated. Each (template) entity is displayed with its attributes (**Identifier**, **Short description**, **Status**, and **Template** indicator checkbox).

- The **Skipped - Bound** tab lists all entities that could not be updated, since they were in use for processing.
  Each entity is displayed with its attributes (**Identifier**, **Short description**, **Status**, and **Template** indicator checkbox).
  Selecting an entity in the list displays further information in the **Details** section at the bottom of the dialog. It indicates the work center where the entity was being used and in which phase of the order or workflow that was being executed.

  > TIP
  > Please note that the number of bound entities may differ between the expected changes and the update result as an entity can have been bound when the changes were calculated but was unbound before the update run started or an entity was still unbound at calculation time but identified and bound before the update run started.

- The **Skipped - Locked** tab lists all entities and template entities that could be updated, since they were being edited by another user or had unsaved changes. Each (template) entity is displayed with its attributes (**Identifier**, **Short description**, **Status**, and **Template** indicator checkbox).
  Selecting an entity or template entity in the list displays further information in the **Details** section at the bottom of the dialog. If in use in another instance of Data Manager or another application, it indicates the login name of the user who was editing the (template) entity, the ID of the device where it was being edited, and the application that was used for editing. If opened with unsaved changes in your current instance of Data Manager, it indicates this situation.

  > TIP
  > Please note that the number of locked entities may differ between the expected changes and the update result as a (template) entity can have been locked when the changes were calculated but was unlocked before the update run started or a (template) entity was not locked at calculation time but opened for editing before the update run started.

- The **Processed with Errors** tab lists all entities and template entities that could not be updated due to errors.

TIP
Please note that (template) entities that are listed on the **Processed with Errors** tab remain fully unchanged even if only a sub-set of the attempted update actions could not be performed.

Each (template) entity is displayed as a nested table, with the common attributes (**Identifier**, **Short description**, **Status**, and **Template** indicator checkbox) as main table and in a collapsible sub-table each (template) entity's expected changes:

- **Action**
  indicates which change should have been performed.
- **Object**
  displays the identifier of the affected object. In the case of replace actions, this is the old component.
- **New**
  displays the new component (for replace actions) or the new value (for property value changes).
- **Old**
  displays the old component (for replace actions) or the old value (for property value changes).

Selecting an entity or template entity in the main table displays further information in the **Error Details** section at the bottom of the dialog.

TIP
To view the information of all tabs, click the **Copy to clipboard** button and paste the text into a system-external text editor.

Click the **Close** button to close the dialog.

TIP

You can perform the **Update (template) entities** function multiple times for a class. This allows you to resolve issues listed on the **Processed with Errors** tab, wait for bound entities to finish processing, or make sure a locked entity is available for updating. The you can re-run the function until all (template) entities have been changed successfully or are listed in the **Skipped - No Changes Required** tab as they were updated successfully in an earlier run of the function.

Graph Transition Triggering on Entities

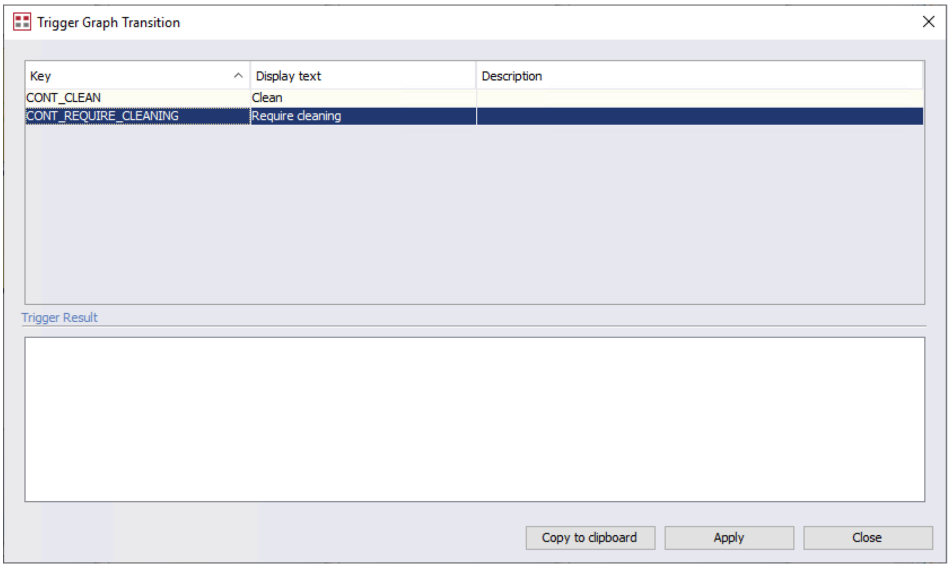
Graph transitions typically happen when an equipment entity is used during execution on the shop floor or when its status expires. To resolve exception situations during processing it may be necessary to perform a status transition with all related actions and conditions on an equipment entity or group by administrative means in Data Manager.

The **Trigger graph transition** function allows to perform a specific transition either on a single equipment entity or on an entire group of entities, but only if the entity is the parent of the group. If the entity is a child entity in a group or a sub-parent in a hierarchical structure of several groups, the transition is only performed on the entity itself, not on any other entities in the group hierarchy.

To trigger a transition on an entity, proceed as follows:

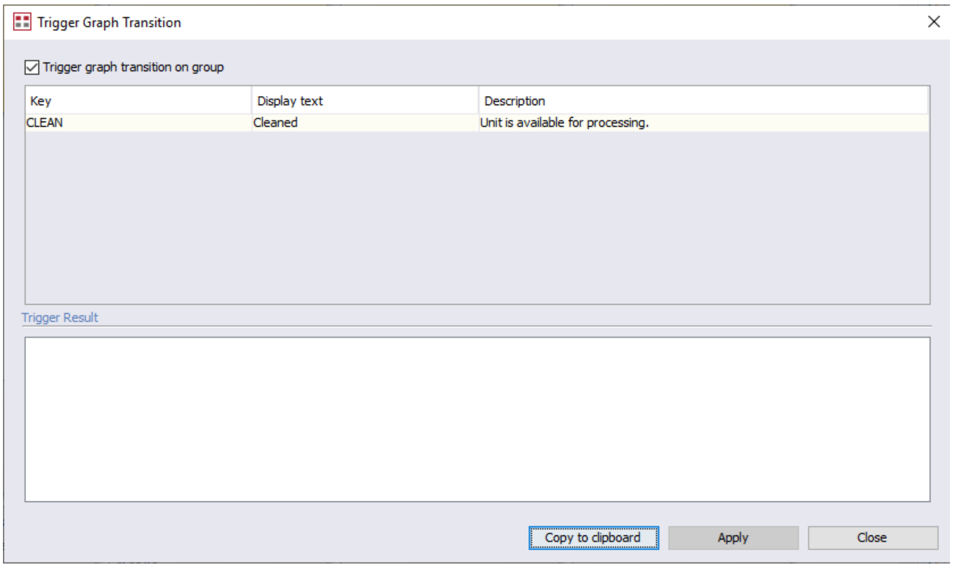
1. In the Search window, locate the entity.
   Double-click it to open it in the Details window.
2. Navigate to the **Graph** tab and select the graph that holds the transition you wish to perform. Then click the **Trigger graph transition** button (page 75).
   The system displays the **Trigger Graph Transition** dialog that lists all triggers available in the current status of the entity.
3. If your entity is the parent of an entity group and you wish to perform the transition on all entities of the group that are also controlled by the selected graph, select the **Trigger graph transition on group** option.
4. Select the trigger you wish to execute and click the **Apply** button.
   The system performs the transition on the entity or entity group and displays either a success message or a list of issues as **Trigger Result**.

5. To copy the complete list to the clipboard for subsequent pasting into a spreadsheet application or a text editor, click the **Copy to clipboard** button. To retain its basic formatting in the application into which it is pasted, the list holds blanks and tabs as separator characters.
   To close the dialog, click the **Close** button.



Key	Display text	Description
CONT_CLEAN	Clean	
CONT_REQUIRE_CLEANING	Require cleaning	

Figure 103: Trigger Graph Transition dialog for single entity



☑ Trigger graph transition on group

Key	Display text	Description
CLEAN	Cleaned	Unit is available for processing.

Figure 104: Trigger Graph Transition dialog for parent entity of group

Usage Lists of Classes

The usage list of a class provides an overview of where the class is referenced in recipes, workflows, or building blocks. This information can support you with assessing the impact a change to the class will have on data that is relevant to processing. Typically, a class's usage list is compiled as first step before class data is changed and the **Update (template) entities** function (page 133) is used to pass on the changes to all of its template entities and entities.

The usage list differentiates between the following types of an equipment class's usages:

- Usage as equipment requirement
  The class is referenced as equipment requirement parameter in a phase.
- Usage within an expression that contains the **Equipment is member of class** function
  The function can occur in equipment requirement rules, transition conditions, or process parameter attributes. The equipment class identifier is the second argument of the function and is given as string. A class can only be detected for the usage list if the parameter contains the equipment class identifier as single string constant and not as a concatenation of strings or variables.

> TIP
> Please note that the system may not be able to determine if an expression uses an equipment class when the expression contains syntax errors.

To compile the usage list of a class, proceed as follows:

1. In the Search window, locate the class.
   Right-click it to open the class's shortcut menu and select the **Compile usage list** function or double-click to open the object and then, from the **File** menu, select **Compile usage list**.
   The system displays the **Usage List** dialog.
2. The **Usage List** dialog provides the **Status category** search options to refine your search.
   Available options: **All**, **Edit**, **Verification**, **Valid**, **Retired**.
   Determines which statuses of master recipes, master workflows, and building blocks that have occurrences of the equipment class in question are considered when the system compiles the usage list.

TIP

Please note that the **Edit** category includes the **Edit** status of master recipes and master workflows as well as the **Draft** status of building blocks. Similarly, the **Valid** category covers **Scheduled** and **Valid** master recipes and workflows and **Approved** building blocks. The **Retired** category includes **Archived** and **Obsolete** master recipes and workflows and **Archived** building blocks.

3. Click the **Run** button to compile the usage list.

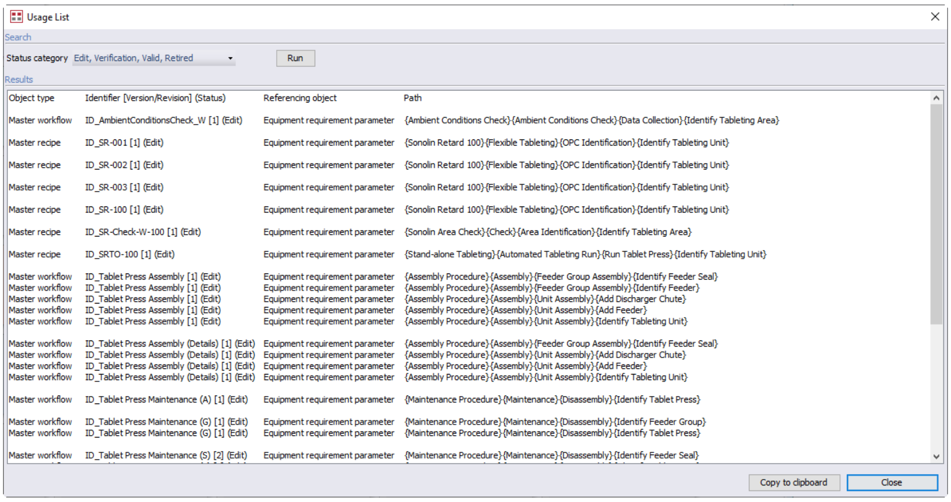
TIPS

Please note that each time you compile a usage list for a class, the system refreshes the data on which it performs the search. For this purpose, it has to access all relevant objects (master recipes, master workflows, and building blocks) on the database to determine if they have been modified since the last time a usage list was compiled.

If there are other users who work with the system on the same database, some objects may be locked as they are currently being edited and are thus not accessible for refreshing. In this case, the usage list compiles with the potentially obsolete data it has retrieved before its previous run, but lists all objects it could not refresh in a specific section.

4. To copy the complete list to the clipboard for subsequent pasting into a spreadsheet application or a text editor, click the **Copy to clipboard** button. To retain its basic formatting in the application into which it is pasted, the list holds blanks and tabs as separator characters.

   To close the dialog, click the **Close** button.



Status category: Edit, Verification, Valid, Retired

Object type	Identifier [Version/Revision] (Status)	Referencing object	Path
Master workflow	ID_AmbientConditionsCheck_W [1] (Edit)	Equipment requirement parameter	{Ambient Conditions Check}{Ambient Conditions Check}{Data Collection}{Identify Tableting Area}
Master recipe	ID_SR-001 [1] (Edit)	Equipment requirement parameter	{Sonolin Retard 100}{Flexible Tableting}{OPC Identification}{Identify Tableting Unit}
Master recipe	ID_SR-002 [1] (Edit)	Equipment requirement parameter	{Sonolin Retard 100}{Flexible Tableting}{OPC Identification}{Identify Tableting Unit}
Master recipe	ID_SR-003 [1] (Edit)	Equipment requirement parameter	{Sonolin Retard 100}{Flexible Tableting}{OPC Identification}{Identify Tableting Unit}
Master recipe	ID_SR-100 [1] (Edit)	Equipment requirement parameter	{Sonolin Retard 100}{Flexible Tableting}{OPC Identification}{Identify Tableting Unit}
Master recipe	ID_SR-Check-W-100 [1] (Edit)	Equipment requirement parameter	{Sonolin Area Check}{Check}{Area Identification}{Identify Tableting Area}
Master recipe	ID_SRTO-100 [1] (Edit)	Equipment requirement parameter	{Stand-alone Tableting}{Automated Tableting Run}{Run Tablet Press}{Identify Tableting Unit}
Master workflow	ID_Tablet Press Assembly [1] (Edit)	Equipment requirement parameter	{Assembly Procedure}{Assembly}{Feeder Group Assembly}{Identify Feeder Seal}
Master workflow	ID_Tablet Press Assembly [1] (Edit)	Equipment requirement parameter	{Assembly Procedure}{Assembly}{Feeder Group Assembly}{Identify Feeder}
Master workflow	ID_Tablet Press Assembly [1] (Edit)	Equipment requirement parameter	{Assembly Procedure}{Assembly}{Unit Assembly}{Add Discharger Chute}
Master workflow	ID_Tablet Press Assembly [1] (Edit)	Equipment requirement parameter	{Assembly Procedure}{Assembly}{Unit Assembly}{Add Feeder}
Master workflow	ID_Tablet Press Assembly [1] (Edit)	Equipment requirement parameter	{Assembly Procedure}{Assembly}{Unit Assembly}{Identify Tableting Unit}
Master workflow	ID_Tablet Press Assembly (Details) [1] (Edit)	Equipment requirement parameter	{Assembly Procedure}{Assembly}{Feeder Group Assembly}{Identify Feeder Seal}
Master workflow	ID_Tablet Press Assembly (Details) [1] (Edit)	Equipment requirement parameter	{Assembly Procedure}{Assembly}{Unit Assembly}{Add Discharger Chute}
Master workflow	ID_Tablet Press Assembly (Details) [1] (Edit)	Equipment requirement parameter	{Assembly Procedure}{Assembly}{Unit Assembly}{Add Feeder}
Master workflow	ID_Tablet Press Assembly (Details) [1] (Edit)	Equipment requirement parameter	{Assembly Procedure}{Assembly}{Unit Assembly}{Identify Tableting Unit}
Master workflow	ID_Tablet Press Maintenance (A) [1] (Edit)	Equipment requirement parameter	{Maintenance Procedure}{Maintenance}{Disassembly}{Identify Tablet Press}
Master workflow	ID_Tablet Press Maintenance (G) [1] (Edit)	Equipment requirement parameter	{Maintenance Procedure}{Maintenance}{Disassembly}{Identify Feeder Group}
Master workflow	ID_Tablet Press Maintenance (G) [1] (Edit)	Equipment requirement parameter	{Maintenance Procedure}{Maintenance}{Disassembly}{Identify Tablet Press}
Master workflow	ID_Tablet Press Maintenance (S) [2] (Edit)	Equipment requirement parameter	{Maintenance Procedure}{Maintenance}{Disassembly}{Identify Feeder Seal}

Figure 105: Usage List dialog

Managing Equipment - Smart Search

Data Manager - Equipment - Smart Search is the recommended mode for most usage scenarios and provides the full potential of the system's flexible search mechanisms. In this mode and depending on its configuration, the system ignores entities that have been generated on the shop floor and are in the **Archived** status (page 129) or all generated entities regardless of their status.

Menus and Toolbars

You can access all relevant functions either from the menu bar in the main Data Manager - Equipment - Smart Search window, from context-sensitive shortcut menus, or from toolbars provided for quick access to frequently used functions.

MENUS

Data Manager - Equipment - Smart Search provides a main menu bar (page 143) with all relevant functions as well as context-sensitive shortcut menus (page 149) for quick access to specific functions.

MAIN MENU BAR

The **main menu bar** offers the following menus and functions:

File

- New equipment class (CTRL+SHIFT+C)
  Creates a new equipment class, thus first opening the **New Equipment Class** dialog and afterwards a new tab in the lower tab bar of the Details window.
- New template equipment entity (CTRL+SHIFT+T)
  Creates a new template equipment entity, thus first opening the **New Template Equipment Entity** dialog and afterwards a new tab in the lower tab bar of the Details window.
- New equipment entity (CTRL+SHIFT+E)
  Creates a new equipment entity, thus first opening the **New Equipment Entity** dialog and afterwards a new tab in the lower tab bar of the Details window.
- New equipment graph (CTRL+SHIFT+G)
  Creates a new equipment graph, thus first opening the **New Equipment Graph** dialog and afterwards a new tab in the lower tab bar of the Details window.
- New property type (CTRL+SHIFT+P)
  Creates a new property type, thus first opening the **New Property Type** dialog and afterwards a new tab in the lower tab bar of the Details window.

- Import
  Starts the process of importing (page 110) an equipment data object with all referenced objects by selecting, checking, and importing a PharmaSuite import ZIP archive.
- Import tags/points
  Starts the process of importing (page 115) the tags and points of equipment entities by selecting, checking, and importing a PharmaSuite import XML file.
- Export <object identifier>
  Starts the process of exporting (page 103) an equipment object with all referenced objects by selecting a target location for the PharmaSuite export ZIP archive.
- Export tags/points of <object identifier>
  Starts the process of exporting (page 107) the tags and points of equipment entities by selecting a target location for the PharmaSuite export XML file.
- Verify tags/points of <object identifier>
  Starts the process of verifying (page 118) the tag paths and point names of equipment entities.
- Save <object identifier> (CTRL+S)
  Saves all changes made to the object that is currently active in the lower tab bar of the Details window. This action saves all changes made on the object's attributes and properties tabs.
- Save all (CTRL+SHIFT+S)
  Saves all changes made to all object that are currently active in the lower tab bar of the Details window. This action saves all changes made on the objects' properties tabs.

  TIP
  Please note that you can only save an object if none of its assigned objects have unsaved changes.

- Duplicate <equipment class identifier> without entities (CTRL+F12)
  Only available for equipment classes.
  Opens the **Duplicate Equipment Class** dialog to create a copy of the class. The new class opens as new tab in the lower tab bar of the Details window.
  The **Duplicate** function copies all attributes and properties of the original class except for its template entity and entity assignments.
- Duplicate <(template) equipment entity identifier> without classes (CTRL+F12)
  Only available for template equipment entities or equipment entities.
  - For template equipment entities
    Opens the **Duplicate Template Equipment Entity** dialog to create a copy of the template entity. The new template entity opens as new tab in the lower tab bar of the Details window.

  - For equipment entities
    Opens the **Duplicate Equipment Entity** dialog to create a copy of the entity. The new entity opens as new tab in the lower tab bar of the Details window.

  The **Duplicate** function copies all attributes and properties of the original including all of its assignments.

- Duplicate <(template) equipment entity identifier> with classes (CTRL+SHIFT+F12)
  Only available for template equipment entities or equipment entities.

  - For template equipment entities
    Opens the **Duplicate Template Equipment Entity** dialog to create a copy of the template entity. The new template entity opens as new tab in the lower tab bar of the Details window.

  - For equipment entities
    Opens the **Duplicate Equipment Entity** dialog to create a copy of the entity. The new entity opens as new tab in the lower tab bar of the Details window.

  The **Duplicate** function copies all attributes and properties of the original including all of its assignments.

  TIP
  Please note that tag names defined with automation properties are suffixed with "_COPY" when they are duplicated.

  Duplicate <equipment graph identifier> (CTRL+F12)
  Only available for equipment graphs.
  Opens the **Duplicate Equipment Graph** dialog to create a copy of the graph. The new graph opens as new tab in the lower tab bar of the Details window.

- Duplicate <property type identifier> (CTRL+F12)
  Only available for property types.
  Opens the **Duplicate Property Type** dialog to create a copy of the property type. The new property type opens as new tab in the lower tab bar of the Details window.

- Change status of <object identifier> (CTRL+SHIFT+H)
  Opens the **Change Status** dialog to perform a status change (page 130) on the equipment class, template entity, or entity that is currently active in the lower tab bar of the Details window.

- Force unbinding of <equipment entity identifier>
  Only available for equipment entities that are currently bound for processing in PharmaSuite for Production Execution.
  Opens an **Electronic Signature** dialog (page 101) to perform and sign for unbinding of the equipment entity that is currently active in the lower tab bar of the Details window.

TIP
This function should only be used as a last resort for scenarios such as cleaning up after processing has met with unresolvable issues in PharmaSuite for Production Execution.

- Update (template) entities of <class identifier>
  Only available for equipment classes that hold template entities or entities.
  Opens the **Update (Template) Entities** dialog to perform a mass change operation on all entities and template entities assigned to the class.

- Compile usage list of <class identifier>
  Opens the **Usage List** dialog to compile a list of recipes, workflows, and building blocks that reference the equipment class as requirement or in an expression.

- Close <object identifier> (CTRL+F4)
  Closes the object that is currently active in the lower tab bar of the Details window.

- Close all (CTRL+SHIFT+F4)
  Closes all objects that are currently open in the lower tab bar of the Details window.

- Restore <object identifier> (CTRL+R)
  Only available if there are unsaved changes.
  Restores the object that is currently active in the lower tab bar of the Details window to its last saved state, thus undoing all changes made in the meantime.

- Refresh <object identifier> (F5)
  Only available if there are no unsaved changes.
  Refreshes the object that is currently active in the lower tab bar of the Details window by retrieving its data again from the database. Thus, it updates all runtime data, such as runtime properties, graph statuses or grouping situations, which may have changed due to processing activities on the shop floor.

- Delete <object identifier>
  Deletes the object that is currently active in the lower tab bar of the Details window.

- Print barcode label for <object identifier> (CTRL+B)
  Only available for equipment entities.
  Sends the barcode of the equipment entity that is currently active in the Details window for printing to a connected printer.

- Exit (ALT+F4)
  Closes the application window.

View

- New search (CTRL+N)
  Opens a new Search window to define another set of search criteria and process its search results.
- Zoom in (CTRL+PLUS)
  Zooms in on the currently active search result, switching the display to the next larger view, either tiles or cards.
- Zoom out (CTRL+MINUS)
  Zooms out from the currently active search result, switching the display to the next smaller view, either tiles or mini tiles.
- Hide Archived objects (CTRL+SHIFT+A)
  Toggles the display of objects in the **Archived** status in the results panel.
- Card tooltips (CTRL+T)
  Toggles the display of the card view as tooltip when you hover over a tile or mini tile.

Window

- Equipment Manager - Smart Search
  Switches to the Equipment - Smart Search mode of Data Manager. Before switching, the system closes all open objects, prompting you to save if you have unsaved changes.
  A checkmark to the left of the selected menu option indicates which mode is currently active.
  Only available if you have the user rights necessary for accessing Data Manager - Equipment - Smart Search.
- Equipment Manager - Basic Search
  Switches to the Equipment - Basic Search mode of Data Manager. Before switching, the system closes all open objects, prompting you to save if you have unsaved changes.
  A checkmark to the left of the selected menu option indicates which mode is currently active.
  Only available if you have the user rights necessary for accessing Data Manager - Equipment - Basic Search.

- Work Center Manager
  Switches to the Work Center mode of Data Manager. Before switching, the system closes all open objects, prompting you to save if you have unsaved changes.
  A checkmark to the left of the selected menu option indicates which mode is currently active.
  Only available if you have the user rights necessary for accessing Data Manager - Work Center.
- Undo layout change
  Revokes the last layout change you have performed. You can undo up to 100 actions, thus you can step by step revoke the last 100 layout changes you have performed. Once you have performed a layout change, the menu function changes to a more precise description of the undo action, such as **Undo resizing** or **Undo dragging**.
- Redo layout change
  Redoes the last layout change you have revoked with the **Undo layout change** function. You can redo up to 100 actions, thus you can step by step redo the last 100 layout changes you have revoked. Once you have performed a layout change, the menu function changes to a more precise description of the undo action, such as **Redo resizing** or **Redo dragging**.
- Save user layout
  Saves the current window layout and overwrites the layout that was last saved by you on this computer.
- Load user layout
  Loads the last layout you have saved on this computer with the **Save user layout** function.
- Reset layout
  Resets the window layout to the system-defined default layout. This function does not affect the saved user layout, which can be restored with the **Load user layout** function.

Help

- Data Manager Help (ALT+F1)
  Opens a web browser to display the start page of the help system (page 7).
- About PharmaSuite
  Opens the **About PharmaSuite** dialog (page 8).

SHORTCUT MENUS

The card and tile views make object type-specific **shortcut menus** available, which you can access by right-clicking a card or tile in the search results panel.

Equipment class

- Open <equipment class identifier> in Details window
  Opens the detail view of the equipment class as new tab in the Details window.
- Duplicate <equipment class identifier> without (template) entities
  Opens the **Duplicate Equipment Class** dialog to create a copy of the class. The new class opens as new tab in the lower tab bar of the Details window.
  The **Duplicate** function copies all attributes and properties of the original class except for its template entity and entity assignments.
- New equipment class
  Creates a new equipment class, thus first opening the **New Equipment Class** dialog and afterwards a new tab in the lower tab bar of the Details window.
- Change status of <equipment class identifier>
  Opens the **Change Status** dialog to perform a status change (page 130) on the equipment class.
- Update (template) entities of <class identifier>
  Only available for equipment classes that hold template entities or entities.
  Opens the **Update (Template) Entities** dialog to perform a mass change operation on all entities and template entities assigned to the class.
- Restore <equipment class identifier>
  Only available if there are unsaved changes.
  Restores the equipment class to its last saved state, thus undoing all changes made in the meantime.
- Refresh <equipment class identifier>
  Only available if there are no unsaved changes.
  Refreshes the equipment class by retrieving its data again from the database.
- Delete <equipment class identifier>
  Only available if the class is in an editable status, such as **Draft**.
  Deletes the equipment class.
- Export <equipment class identifier>
  Starts the process of exporting (page 103) an equipment class with all referenced template entities, entities, graphs, and property types by selecting a target location for the PharmaSuite export ZIP archive.

- Export tags/points of <equipment class identifier>
  Starts the process of exporting (page 107) the tags and points of the entities of an equipment class by selecting a target location for the PharmaSuite export XML file.
- Verify tags/points of <equipment class identifier>
  Starts the process of verifying (page 118) the tag paths and point names of the equipment entities of a class.
- Compile usage list of <class identifier>
  Opens the **Usage List** dialog to compile a list of recipes, workflows, and building blocks that reference the equipment class as requirement or in an expression.
- Assign <equipment class identifier> to <equipment entity identifier> entity
  Only available if the active tab of the Details window holds a template equipment entity or an equipment entity.
  Assigns the class to the template entity or entity that is displayed in the Details window.
- Create entity from <equipment class identifier> class
  Creates a new equipment entity on the basis of the data defined for the equipment class. The function passes all attributes and properties defined with the class on to the new entity. Moreover, the class is assigned to the entity as its style class.
- Create template entity from <equipment class identifier> class
  Creates a new template equipment entity on the basis of the data defined for the equipment class. The function passes all attributes and properties defined with the class on to the new template entity. Moreover, the class is assigned to the template entity as its style class.

Template equipment entity

- Open <template equipment entity identifier> in Details window
  Opens the detail view of the template equipment entity as new tab in the Details window.
- Duplicate <template equipment entity identifier> without classes
  Opens the **Duplicate Template Equipment Entity** dialog to create a copy of the template entity. The new template entity opens as new tab in the lower tab bar of the Details window.
  The **Duplicate** function copies all attributes and properties of the template entity except for its class assignments.
- Duplicate <template equipment entity identifier> with classes
  Opens the **Duplicate Template Equipment Entity** dialog to create a copy of the template entity. The new template entity opens as new tab in the lower tab bar of the Details window.
  The **Duplicate with classes** function copies all attributes and properties of the original including all of its assignments.

- New template equipment entity (CTRL+SHIFT+T)
Creates a new template equipment entity, thus first opening the **New Template Equipment Entity** dialog and afterwards a new tab in the lower tab bar of the Details window.
- Change status of <template equipment entity identifier>
Opens the **Change Status** dialog to perform a status change (page 130) on the template equipment entity.
- Restore <template equipment entity identifier>
Only available if there are unsaved changes.
Restores the template equipment entity to its last saved state, thus undoing all changes made in the meantime.
- Refresh <template equipment entity identifier>
Only available if there are no unsaved changes.
Refreshes the template equipment entity by retrieving its data again from the database.
- Delete <template equipment entity identifier>
Only available if the template entity is in an editable status, such as **Draft**, and has not been used as template for generating equipment entities during execution.
Deletes the template equipment entity.
- Export <template equipment entity identifier>
Starts the process of exporting (page 103) a template equipment entity with all referenced graphs and property types by selecting a target location for the PharmaSuite export ZIP archive.
- Assign <template equipment entity identifier> to <equipment class identifier> class
Only available if the active tab of the Details window holds an equipment class.
Assigns the template entity to the equipment class that is displayed in the Details window.
- Create class from <template equipment entity identifier> template equipment entity
Creates a new equipment class on the basis of the data defined for the template equipment entity. The function passes all attributes and properties defined with the template entity on to the new class. Thus, the data on the **Style** tab of the new class is not defined.
- Create entity from <template equipment entity identifier> template equipment entity
Creates a new equipment entity on the basis of the data defined for the template equipment entity. The function passes all attributes, properties, and assignments defined with the template entity on to the new entity.

Equipment entity

- Open <equipment entity identifier> in Details window
  Opens the detail view of the equipment entity as new tab in the Details window.
- Duplicate <equipment entity identifier> without classes
  Opens the **Duplicate Equipment Entity** dialog to create a copy of the entity. The new entity opens as new tab in the lower tab bar of the Details window.
  The **Duplicate** function copies all attributes and properties of the original except for its class assignments.
- Duplicate <equipment entity identifier> with classes
  Opens the **Duplicate Equipment Entity** dialog to create a copy of the entity. The new entity opens as new tab in the lower tab bar of the Details window.
  The **Duplicate with classes** function copies all attributes and properties of the original including its assignments.
- New equipment entity
  Creates a new equipment entity, thus first opening the **New Equipment Entity** dialog and afterwards a new tab in the lower tab bar of the Details window.
- Change status of <equipment entity identifier>
  Opens the **Change Status** dialog to perform a status change (page 130) on the equipment entity.
- Force unbinding of <equipment entity identifier>
  Only available for equipment entities that are currently bound for processing in PharmaSuite for Production Execution.
  Opens an **Electronic Signature** dialog (page 101) to perform and sign for unbinding of the equipment entity that is currently active in the lower tab bar of the Details window.

  TIP
  This function should only be used as a last resort for scenarios such as cleaning up after processing has met with unresolvable issues in PharmaSuite for Production Execution.
- Restore <equipment entity identifier>
  Only available if there are unsaved changes.
  Restores the equipment entity to its last saved state, thus undoing all changes made in the meantime.
- Refresh <equipment entity identifier>
  Only available if there are no unsaved changes.
  Refreshes the equipment entity by retrieving its data again from the database. Thus, it updates all runtime data, such as runtime properties, graph statuses or grouping situations, which may have changed due to processing activities on the shop floor.

- Delete <equipment entity identifier>
  Only available if the entity is in an editable status, such as **Draft**, and has not been used for processing yet.
  Deletes the equipment entity.
- Export <equipment entity identifier>
  Starts the process of exporting (page 103) an equipment entity with all referenced graphs and property types by selecting a target location for the PharmaSuite export ZIP archive.
- Export tags/points of <equipment entity identifier>
  Starts the process of exporting (page 107) the tags and points of an equipment entity by selecting a target location for the PharmaSuite export XML file.
- Verify tags/points of <equipment entity identifier>
  Starts the process of verifying (page 118) the tag paths and point names of an equipment entity.
- Assign <equipment entity identifier> to <equipment class identifier> class
  Only available if the active tab of the Details window holds an equipment class.
  Assigns the entity to the equipment class that is displayed in the Details window.
- Create class from <equipment entity identifier> equipment entity
  Creates a new equipment class on the basis of the data defined for the equipment entity. The function passes all attributes and properties defined with the entity on to the new class. Thus, the data on the **Style** tab of the new class is not defined.

Equipment graph

- Open <equipment graph identifier> in Details window
  Opens the detail view of the equipment graph as new tab in the Details window.
- Duplicate <equipment graph identifier>
  Opens the **Duplicate Equipment Graph** dialog to create a copy of the graph. The new graph opens as new tab in the lower tab bar of the Details window.
- New equipment graph
  Creates a new equipment graph, thus first opening the **New Equipment Graph** dialog and afterwards a new tab in the lower tab bar of the Details window.
- Change status of <equipment graph identifier>
  Opens the **Change Status** dialog to perform a status change (page 130) on the equipment graph.
- Restore <equipment graph identifier>
  Only available if there are unsaved changes.
  Restores the equipment graph to its last saved state, thus undoing all changes made in the meantime.

- Refresh <equipment graph identifier>
  Only available if there are no unsaved changes.
  Refreshes the equipment graph by retrieving its data again from the database.
- Delete <equipment graph identifier>
  Deletes the equipment graph.
  Only available if the graph is not assigned to another object, such as a class or an entity.
- Export <equipment graph identifier>
  Starts the process of exporting (page 103) an equipment graph with all referenced property types by selecting a target location for the PharmaSuite export ZIP archive.
- Assign <equipment graph identifier> to <object type identifier> <object type>
  Only available if the active tab of the Details window holds an equipment class, template entity, or entity and the graph has not been assigned already to the class, template entity, or entity.
  Assigns the graph to the equipment class, template entity, or entity that is displayed in the Details window.

Property type

- Open <property type identifier> in Details window
  Opens the detail view of the property type as new tab in the Details window.
- Duplicate <property type identifier>
  Opens the **Duplicate Property Type** dialog to create a copy of the property type. The new property type opens as new tab in the lower tab bar of the Details window.
- New property type
  Creates a new property type, thus first opening the **New Property Type** dialog and afterwards a new tab in the lower tab bar of the Details window.
- Restore <property type identifier>
  Only available if there are unsaved changes.
  Restores the property type to its last saved state, thus undoing all changes made in the meantime.
- Refresh <property type identifier>
  Only available if there are no unsaved changes.
  Refreshes the property type by retrieving its data again from the database.
- Delete <property type identifier>
  Only available if the property type is not assigned to another object, such as a class, an entity, or a graph.
  Deletes the property type.

- Export <property type identifier>
  Starts the process of exporting (page 103) a property type by selecting a target location for the PharmaSuite export ZIP archive.
- Assign <property type identifier> to <object type identifier> <object type>
  Only available if the active tab of the Details window holds a suitable tab of an equipment class, template entity, entity, or graph and the property type has not been assigned already to the class, template entity, entity, or graph.
  Assigns the property type to the property tab (**Specification**, **Process**, **Engineering**) of the equipment class, template entity, or entity or to the **Property Type** tab of the equipment graph that is displayed in the Details window.

TOOLBARS

Data Manager provides several toolbars for context-sensitive quick access to specific functions.

MAIN TOOLBARS

The **main toolbars** offer shortcuts to the more frequently required menu functions:

- File toolbar with
  - New equipment class (page 71)
  - New template equipment entity (page 72)
  - New equipment entity (page 71)
  - New equipment graph (page 71)
  - New property type (page 72)
  - Refresh (page 72)
  - Save (page 72)
  - Close (page 71)
  - Delete (page 71)
  - Change status (page 71)
  - Print barcode label (page 72)
- View toolbar with
  - Zoom in (page 73)
  - Zoom out (page 73)
- Help toolbar with
  - Help (page 71)

PROPERTY TAB TOOLBAR

The **property tab toolbar** provides the following functions for sorting and managing the property types assigned to an equipment class, entity, or graph:

- Move down (page 74)
- Move up (page 74)
- Display on card (page 73)
  Only available for equipment classes and graphs.
- Remove (page 75)

Detail Data

From the upper tab bar in the Details window, you can access the attributes, properties, assignments, and usage data of the following equipment-related objects:

- Equipment classes (page 156)
- Template equipment entities (page 163)
- Equipment entities (page 172)
- Equipment graphs (page 188)
- Property types (page 199)

CLASS

The Details window of a class contains the following data tabs:

- Basic (page 157)
- Specification (page 158)
- Process (page 159)
- Engineering (page 159)
- Graph (page 160)
- Entity (page 160)
- Style (page 161)
- Change History (page 161)
- Status History (page 162)

Basic

The **Basic** tab lists the Asset Attributes of the equipment class.

Click an attribute value to edit it. The system displays a text cursor or a button to access a suitable editor (page 10) for the required value type. Any asset attribute values you define for an equipment class are copied to an equipment entity created from the class. The attribute values copied to the equipment entity can, however, be changed on the equipment entity without affecting the values defined on the class.

> **TIP**
>
> Please note that once you have created an equipment class, you can no longer modify its **Identifier**.

Default classes

PharmaSuite provides the following equipment classes with any default installation of the system. They are intended as templates to create entities that contain all required graphs and property types:

- Container_RS_1
  The default class for containers includes the following default objects:
  - Graph:
    ContainerCleaning_RS_1
  - Property types:
    - Container Clean Shelflife (RS)
    - Container Sublot (RS)
    - Container Tare (RS)
    - Container Type (RS)
- Room_RS_1
  The default class for rooms includes the following default objects:
  - Graph
    RoomCleaning_RS_1
  - Property types:
    - General Clean Shelflife (RS)
    - Room Cleaning Rules (RS)
    - Room Type (RS)
    - Work Center Assignment (RS)

- Scale_RS_1
  The default class for scales includes the following default objects:
  - Graphs:
    - ScaleCalibration_RS_1
    - ScaleTest_RS_1
  - Property types:
    - Scale Calibration Shelflife (RS)
    - Scale Configuration (RS)
    - Scale Load (RS)
    - Scale Ranges (RS)
    - Scale Test and Calibration (RS)
    - Scale Test Shelflife (RS)
    - Scale Type (RS)
    - Work Center Assignment (RS)

The **Level** attribute refers to the physical model of the S88 standard and provides the following options:

- Control module
- Equipment module
- Unit
- Process cell

The **Disposable** attribute allows you to choose (**Yes**, **No**, **N/A**) if the class groups (template) equipment entities that are intended for single use.

The **Logbook enabled** attribute allows you to choose (**Yes**, **No**, **N/A**) if the class groups equipment entities whose life cycle needs to be tracked in the logbook.

SPECIFICATION

The **Specification** tab lists the property types assigned as specification properties to the equipment class.

- Click a property value to edit it. The system displays a text cursor or a button to access a suitable editor (page 10) for the required value type.
- To add a property to the list, drag and drop (page 100) its property type on the specification tab.

- To change a property's position in the list, click the **Move up** or **Move down** buttons (page 74).
- To display a property on the card view of the equipment class, click the **Display on card** button (page 73).
- To remove a property from the list, click the **Remove property** button (page 75).

PROCESS

The **Process** tab lists the property types assigned as process properties to the equipment class.

- Click a property value to edit it. The system displays a text cursor or a button to access a suitable editor (page 10) for the required value type.
- To add a property to the list, drag and drop (page 100) its property type on the process tab.
- To change a property's position in the list, click the **Move up** or **Move down** buttons (page 74).
- To display a property on the card view of the equipment class, click the **Display on card** button (page 73).
- To remove a property from the list, click the **Remove property** button (page 75).

ENGINEERING

The **Engineering** tab lists the property types assigned as engineering properties to the equipment class.

- Click a property value to edit it. The system displays a text cursor or a button to access a suitable editor (page 10) for the required value type.
- To add a property to the list, drag and drop (page 100) its property type on the engineering tab.
- To change a property's position in the list, click the **Move up** or **Move down** buttons (page 74).
- To display a property on the card view of the equipment class, click the **Display on card** button (page 73).
- To remove a property from the list, click the **Remove property** button (page 75).

GRAPH

The **Graph** tab lists the equipment graphs assigned to the equipment class.

- To add a graph to the list, drag and drop (page 100) its card or tile on the Details window of the class to which you wish to assign it.

> TIP
> Please note that you can only assign graphs that do not have unsaved changes and contain at least one status.

- To change a graph's position in the list, click the **Move up** or **Move down** buttons (page 74).
- To display a graph with its short description on the card view of the equipment class, click the **Display on card** button (page 73).
- To remove a graph from the list, click the **Remove graph** button (page 75).

ENTITY

The **Entity** tab lists the equipment entities and template equipment entities assigned to the equipment class. The checkbox in the **Style class** column indicates which of the assigned (template) entities take their styling (color, icon, properties displayed on card view) from the class. The checkbox in the **Template** column indicates which of the assigned objects are plain entities and which of them are template entities.

- To add a (template) entity to the list, drag and drop (page 100) its card or tile on the Details window of the class to which you wish to assign it.

> TIP
> Please note that the system only checks if the properties of the assigned (template) entity correspond to those of your class, but not on which tabs (**Specification**, **Process**, **Engineering**) the properties are located. Thus, the system does not rearrange the (template) entity's properties to match the properties' positioning on the class.

- To remove a (template) entity from the list, select or multi-select it and click the **Remove equipment entity** button (page 75) or press the DEL key.
  Removing a (template) entity from the list of entities also removes the class from the list of classes shown on the **Class** tab of the (template) entity's detail view.

Style

The **Style** tab provides tools for defining the visual appearance of the equipment classes' tiles and cards:

- Background color
- Font color
- Icon

TIP

Please note that (template) equipment entities assigned to a class can have the class set as their style class and thus inherit its visual appearance.

Change History

The **Change History** tab provides a list of all changes that have been performed on the equipment class, which also includes indirect assignment changes that occur when an entity's assignment change also involves the class.
For each change event that affected the class, the table displays one row.

To filter the change event entries by date, select the date range you wish to consider. The system only displays the entries that have been registered between the two dates you have selected.
The date range starts at 12:00:00 AM on the date you have selected and ends at 23:59:59 PM on the date you have selected.
When you open the change history, the date range is preset to start with the date when the last change event occurred and to end with the current date. If the system has no change events registered for the object, the date range is preset to the current date.
To refresh the list of change events, click the **Refresh** button (page 75). All change events involving the class that have occurred since you opened the change history are then included.

The following table columns are available and are filled if relevant to the respective change event:

- Timestamp
  indicates the date and time when the change event occurred.
- Action
  indicates the change event that triggered the entry in the change history.

- Affected object
  indicates the object that is affected by the change event. So, when you update the basic data of your class, the change history shows the identifier of the attribute you have modified and when you assign a template entity, entity, graph, or property type to your class, the change history shows the identifier of the assigned object.
- Old content/New content
  for data change events on your class, displays the changed values.
- Signature
  displays two items per signature (page 101):
  - User and login names of the user who performed the signature.
  - Timestamp when the signature was recorded.
- Comment
  displays the comment recorded for each signature. For double signatures, each comment is prefixed with **Comment 1** or **Comment 2** to indicate with which signature the comment was recorded.

  > TIP
  > Signatures and their comments are only recorded for mass change operations.
- Logged-in user
  login name of the user under whose login the change event occurred.

Status History

The **Status History** tab provides a list of all status changes that have been performed on the equipment class.
For each status change that affected the class, the table displays one row.

To filter the status change entries by date, select the date range you wish to consider. The system only displays the entries that have been registered between the two dates you have selected.
The date range starts at 12:00:00 AM on the date you have selected and ends at 23:59:59 PM on the date you have selected.
When you open the status history, the date range is preset to start with the date when the last status change occurred and to end with the current date. If the system has no status changes registered for the object, the date range is preset to the current date.
To refresh the list of status changes, click the **Refresh** button (page 75). All status changes involving the class that have occurred since you opened the status history are then included.

The following table columns are available and are filled if relevant to the respective status change:

- Timestamp
  indicates the date and time when the status change occurred.
- Action
  indicates the status change that triggered the entry in the status history.
- To status
  indicates the status to which the class was moved.
- Signature
  displays two items per signature (page 101):
  - User and login names of the user who performed the signature.
  - Timestamp when the signature was recorded.
- Comment
  displays the comment recorded for each signature. For double signatures, each comment is prefixed with **Comment 1** or **Comment 2** to indicate with which signature the comment was recorded.

TEMPLATE ENTITY

The Details window of an entity contains the following data tabs:

- Basic (page 164)
- Specification (page 166)
- Process (page 168)
- Engineering (page 168)
- Graph (page 169)
- Class (page 169)
- Material (page 169)
- Change History (page 170)
- Status History (page 171)

BASIC

The **Basic** tab lists the Asset, Automation, Historian, and Entity Barcode attributes of the template equipment entity.

Click an attribute value to edit it. The system displays a text cursor or a button to access a suitable editor (page 10) for the required value type.
Asset, Automation, and Historian attribute values defined for a template entity are copied to all equipment entities created from the template in Data Manager or generated from the template on the shop floor.
Asset attribute values you define for a template equipment entity are copied to an equipment class created from the entity.
The attribute values copied to the new object can, however, be changed on the new object without affecting the values defined on the template entity.

TIP

Please note that once you have created a template equipment entity, you can no longer modify its **Identifier**.

The **Level** attribute refers to the physical model of the S88 standard and provides the following options:

- Control module
- Equipment module
- Unit
- Process cell

The **Disposable** attribute allows you to choose (**Yes**, **No**) if the entities generated from the template are intended for single use.

The **Logbook enabled** attribute allows you to choose (**Yes**, **No**) if the life cycles of all generated entities need to be tracked in the logbook.

The **Label layout (default)** attribute shows which label layout is pre-configured by the system and defines the layout that will be used for printing entity barcode labels when the template entity is used for generating entities on the shop floor.
With the **Label layout** attribute, you can override the default setting and define a different label layout to be used. For selecting a suitable label layout, the system provides the **Label Layout Selection** editor (page 58).
If you leave the box blank, the system uses the pre-configured default label design displayed in the **Label layout (default)** attribute.

When a new entity is generated from the template entity during execution, the **Entity Barcode** attributes determine what the identifier and barcode of the new entity will look like.
The identifier of a generated entity consists of a prefix, which is a configurable sequence of characters and a count, whose number of digits is also configurable.

- The **Barcode prefix character (default)** is read-only and shows which character (if any) is pre-configured as barcode prefix by the system.
- With the **Barcode prefix source** attribute, you can define if the barcode will be prefixed with the default character (**From configuration key**) or with a character specifically defined with the template (**Template-specific**).
- The **Character** attribute is only available when the **Barcode prefix source** is set to **Template-specific**. It must be no longer than one character but can also be left blank.
- The **Identifier prefix (default)** is read-only and shows the sequence of characters that is pre-configured by the system to precede the count of a generated entity.
- With the **Identifier prefix** you can override the system default with another sequence of characters. It can have a length of up to 30 characters but can also remain blank.
- The **Sequencer length (default)** is read-only and shows the pre-configured number of digits to be used for the count of generated entities.
- With the **Sequencer length** you can override the system default with another number of digits for the count. The system uses leading zeros to fill up the digits if the actual count would require fewer digits.

Examples:

Barcode prefix source	Character	Identifier prefix	Sequencer length	Entity identifier Entity barcode
Template-specific	&	FL	5	FL00001 &FL00001
Template-specific		F-Liner	10	F-Liner0000000001 F-Liner0000000001

Barcode prefix source	Character	Identifier prefix	Sequencer length	Entity identifier Entity barcode
Template-specific	§		15	000000000000001 §000000000000001
Template-specific	$	FL-	8	FL-00000001 $FL-00000001

With the Automation attributes, you can establish the connection to the automation layer, by way of the **Automation Integration server name** and the **Live Data Area path**. The information is required if you wish to use automation phases in your workflows or master recipes. The system shows the pre-configured default paths, which are used unless you define another path, which then overrides the default.

The **Live Data Area path** specifies the path the system uses to access the tags on your generated equipment entity within the Factory Talk Directory. Together with the specific tag names you can configure when you edit the automation properties (page 10) of the entity, it forms the full path to a specific tag. The path consists of a protocol prefix (**RNA://$Global**), an application name, and one or more area names.

So the path to an equipment entity, located in a **Coating** sub-area of a **Building03** area within a company's **MES** application would be **RNA://$Global/MES/Building03/Coating**.

With the Historian attributes, you can access the data archived in a Historian application. Unless you can use the pre-defined defaults, you need to specify the **Provider** for interfacing to the **Historian server**, the **Historian access server**, and also the name of the **Automation Integration server** through which the data access is channeled. The information is required if your master recipe or workflow holds phases for retrieving historical data.

SPECIFICATION

The **Specification** tab lists the property types assigned as specification properties to the template equipment entity.

TIP

Automation properties indicate if a tag path has been specified for them.
(-) means that they have no tag path,
(+) means that at least one tag path has been defined.
Please note, however, that the indicator only refers to the existence of a tag path and not to its validity.

Template entities to be used for generating material-related equipment entities during production require specific object assignments:

Object type	Usage	Data type Content	Purpose
Property type	Specification	EquipmentType Hybrid (RS)	---
Property type	Runtime	String	Current Sublot (RS)
Property type	Runtime	String	Base Sublot (RS)

When a phase building block generates an entity during identification of a sublot, the system determines the suitable template entity on account of the template entity assigned to the sublot's material.

When the identified sublot is split, but continues to exist, the system fills the new entity's **Current Sublot (RS)** runtime property with the identifier of the sublot that is generated along with the entity. The **Base Sublot (RS)** runtime property is filled with the identifier of the sublot that has been split in order to generate the sublot of the **Current Sublot (RS)** runtime property.

When the identified sublot is consumed immediately and thus ceases to exist, its identifier is shown in the **Base Sublot (RS)** runtime property of the generated entity. The **Current Sublot (RS)** runtime property of the entity remains blank in this case.

- Click a property value to edit it. The system displays a text cursor or a button to access a suitable editor (page 10) for the required value type.
- To add a property to the list, drag and drop (page 100) its property type on the specification tab.
- To change a property's position in the list, click the **Move up** or **Move down** buttons (page 74).
- To remove a property from the list, click the **Remove property** button (page 75).

PROCESS

The **Process** tab lists the property types assigned as process properties to the template equipment entity.

TIP

Automation properties indicate if a tag path has been specified for them.
(-) means that they have no tag path,
(+) means that at least one tag path has been defined.
Please note, however, that the indicator only refers to the existence of a tag path and not to its validity.

- Click a property value to edit it. The system displays a text cursor or a button to access a suitable editor (page 10) for the required value type.
- To add a property to the list, drag and drop (page 100) its property type on the process tab.
- To change a property's position in the list, click the **Move up** or **Move down** buttons (page 74).
- To remove a property from the list, click the **Remove property** button (page 75).

ENGINEERING

The **Engineering** tab lists the property types assigned as engineering properties to the template equipment entity.

TIP

Automation properties indicate if a tag path has been specified for them.
(-) means that they have no tag path,
(+) means that at least one tag path has been defined.
Please note, however, that the indicator only refers to the existence of a tag path and not to its validity.

- Click a property value to edit it. The system displays a text cursor or a button to access a suitable editor (page 10) for the required value type.
- To add a property to the list, drag and drop (page 100) its property type on the engineering tab.
- To change a property's position in the list, click the **Move up** or **Move down** buttons (page 74).
- To remove a property from the list, click the **Remove property** button (page 75).

Graph

The **Graph** tab lists the equipment graphs assigned to the template equipment entity. Each graph shows with its initial status and expiry date (if set), both of which cannot be changed. The statuses controlled by the graph do not apply to the template entity itself, but to all entities generated from it.

- To add a graph to the list, drag and drop (page 100) its card or tile on the Details window of the template entity to which you wish to assign it.

> TIP
> Please note that you can only assign graphs that do not have unsaved changes and contain at least one status.

- To change a graph's position in the list, click the **Move up** or **Move down** buttons (page 74).
- To remove a graph from the list, click the **Remove graph** button (page 75).

Class

The **Class** tab lists the equipment classes assigned to the template equipment entity. A template equipment entity can belong to several equipment classes, but takes its styling (color, icon, properties displayed on card view) from the class set as its **Style class**.

- To add a class to the list, drag and drop (page 100) its card or tile on the Details window of the entity to which you wish to assign it.

> TIP
> Please note that the system only checks if the properties of the assigned class correspond to those of your template entity, but not on which tabs (**Specification**, **Process**, **Engineering**) the properties are located. Thus, the system does not rearrange the template entity's properties to match the properties' positioning on the class.

- To define the styling of the template entity by setting an assigned class as style class, select the **Style class** checkbox of the class whose styling the template entity is to adopt.
- To remove a class from the list, select or multi-select it and click the **Remove equipment class** button (page 75) or press the DEL key.
  Removing a class from the list of classes also removes the template entity from the list of entities shown on the **Entity** tab of the class' detail view.

Material

The **Material** tab lists the materials assigned to the template equipment entity. The assignments between materials and template entities are maintained in PharmaSuite for Production Management, where only those templates entities are available for assignment that have a specification property of the **Hybrid (RS)** equipment type.

CHANGE HISTORY

The **Change History** tab provides a list of all changes that have been performed on the template equipment entity, which also includes indirect assignment changes that occur when a class' assignment change also involves the template entity.
For each change event that affected the template entity, the table displays one row.

To filter the change event entries by date, select the date range you wish to consider. The system only displays the entries that have been registered between the two dates you have selected.
The date range starts at 12:00:00 AM on the date you have selected and ends at 23:59:59 PM on the date you have selected.
When you open the change history, the date range is preset to start with the date when the last change event occurred and to end with the current date. If the system has no change events registered for the object, the date range is preset to the current date.
To refresh the list of change events, click the **Refresh** button (page 75). All change events involving the template entity that have occurred since you opened the change history are then included.

The following table columns are available and are filled if relevant to the respective change event:

- Timestamp
  indicates the date and time when the change event occurred.
- Action
  indicates the change event that triggered the entry in the change history.
- Affected object
  indicates the object that is affected by the change event. So, when you update the basic data of your template entity, the change history shows the identifier of the attribute you have modified and when you assign a class, graph, or property type to your template entity, the change history shows the identifier of the assigned class, graph, or property type.
- Old content/New content
  for data change events on your template entity, displays the changed values.
- Information
  indicates additional details on the change, such as if it was performed in the context of a mass change or a tag or point import.
- Logged-in user
  login name of the user under whose login the change event occurred.

STATUS HISTORY

The **Status History** tab provides a list of all status changes that have been performed on the template equipment entity.
For each status change that affected the template entity, the table displays one row.

To filter the status change entries by date, select the date range you wish to consider. The system only displays the entries that have been registered between the two dates you have selected.
The date range starts at 12:00:00 AM on the date you have selected and ends at 23:59:59 PM on the date you have selected.
When you open the status history, the date range is preset to start with the date when the last status change occurred and to end with the current date. If the system has no status changes registered for the object, the date range is preset to the current date.
To refresh the list of status changes, click the **Refresh** button (page 75). All status changes involving the template entity that have occurred since you opened the status history are then included.

The following table columns are available and are filled if relevant to the respective status change:

- Timestamp
  indicates the date and time when the status change occurred.
- Action
  indicates the status change that triggered the entry in the status history.
- To status
  indicates the status to which the template entity was moved.
- Signature
  displays two items per signature (page 101):
  - User and login names of the user who performed the signature.
  - Timestamp when the signature was recorded.
- Comment
  displays the comment recorded for each signature. For double signatures, each comment is prefixed with **Comment 1** or **Comment 2** to indicate with which signature the comment was recorded.

Entity

The Details window of an entity contains the following data tabs:

- Basic (page 172)
- Specification (page 176)
- Process (page 178)
- Engineering (page 179)
- Graph (page 180)
- Class (page 181)
- Grouping (page 181)
- Context (page 183)
- Logbook (page 183)
- Migrated History (page 186) (only for migrated equipment entities)
- Change History (page 186)
- Status History (page 187)

Basic

The **Basic** tab lists the Asset, Automation, and Historian attributes of the equipment entity.

Click an attribute value to edit it. The system displays a text cursor or a button to access a suitable editor (page 10) for the required value type. Any asset attribute values you define for an equipment entity are copied to an equipment class created from the entity. The attribute values copied to the equipment class can, however, be changed on the equipment class without affecting the values defined on the entity.

When an equipment entity is used for processing in PharmaSuite for Production Execution it is bound to the process. Under very rare circumstances, in may be necessary to unbind an entity form the process in order to resolve data issues that cannot be solved in any other manner. For these cases, the system provides the **Force unbinding** function in the File menu (page 143) or the entities shortcut menu (page 152).

TIPS

Please note that once you have created an equipment entity, you can no longer modify its **Identifier**.

For scale equipment entities, which hold a property of the **Scale (RS)** EquipmentType, the **Short description** is displayed during execution as button text on the scale buttons in the **Select scale** phases of Weighing or Dispense operations. If there is no short description defined, the system displays the scale's **Identifier** instead.
Thus, the maximum number of characters for the **Short description** should not exceed 12 and should include a blank to trigger a line break when displayed on the button. One line on the button can hold approximately 6 characters. If the text exceeds the button width, it is truncated.

The **Level** attribute refers to the physical model of the S88 standard and provides the following options:

- Control module
- Equipment module
- Unit
- Process cell

The **Disposable** attribute allows you to choose (**Yes**, **No**) if the entity is intended for single use.

The **Logbook enabled** attribute allows you to choose (**Yes**, **No**) if the entity's life cycle needs to be tracked in the logbook.

If the entity was generated on the shop floor, the **Template used** attribute displays the identifier of the template entity that was used for its generation.

The **Label layout (default)** attribute shows which label layout is pre-configured by the system and defines the layout that will be used for printing the barcode labels of the entity.
With the **Label layout** attribute, you can override the default setting and define a different label layout to be used. For selecting a suitable label layout, the system provides the **Label Layout Selection** editor (page 58).
If you leave the box blank, the system uses the pre-configured default label design displayed in the **Label layout (default)** attribute.

At regular intervals, the system checks all equipment entities with expirable statuses to determine if statuses have expired. The **Automatic status change after expiry** attribute indicates if the system-performed status transition is enabled for the equipment entity. When an equipment entity is bound for processing or locked for editing, the system does not perform any status transition, even if its status has expired. The attribute retains its **Enabled** setting, and the entity is considered again during the next run of the expiry update.
If the system encounters an error while trying to update an expired status, however, it sets the **Automatic status change after expiry** attribute to **Disabled after error**. From then on, the entity is ignored during subsequent runs of the **Automatic status change after expiry** function. Once the error that prevents a successful status transition has been resolved, you need set the attribute to **Enabled**. The system will then consider the entity again for checking expiry dates and automatically performing status transitions.

When you create a new entity, the system pre-fills its **Barcode** attribute with a string of characters that consists of a system-configured equipment entity barcode prefix and the entity's identifier you have specified as first step of the creation process. The system-configured prefix may be empty so that the barcode and the entity’s identifier are identical. For barcodes of entities that are generated from a template entity during processing, however, the system calculates the **Barcode** attribute from the definitions made in the **Entity Barcode Attributes** section on the template entity's **Basic** tab (page 164).
To print the barcode label of an entity, proceed as follows:

1. Make sure the **Barcode** attribute cell contains the required unique string of characters.

   TIP
   Do not type leading of trailing blanks before or after the barcode string, since this may interfere with the system-internal formatting conversion and render the barcode unreadable for scanners.

2. Save the equipment entity.
   The system enables the **Print barcode label** button on the **File** toolbar as well as the respective function in the **File** menu.

3. Click the **Print barcode label** button or select the respective menu function to print the barcode on a connected printer, typically the default printer defined in your Windows operating system.

With the Automation attributes, you can establish the connection to the automation layer, by way of the **Automation Integration server name** and the **Live Data Area path**. The information is required if you wish to use automation phases in your workflows or master recipes. The system shows the pre-configured default paths, which are used unless you define another path, which then overrides the default.
The **Live Data Area path** specifies the path the system uses to access the tags on your equipment entity within the Factory Talk Directory. Together with the specific tag names you can configure when you edit the automation properties (page 10) of the entity, it forms the full path to a specific tag. The path consists of a protocol prefix (**RNA://$Global**), an application name, and one or more area names.
So the path to an equipment entity, located in a **Coating** sub-area of a **Building03** area within a company's **MES** application would be
RNA://$Global/MES/Building03/Coating.

With the Historian attributes, you can access the data archived in a Historian application. Unless you can use the pre-defined defaults, you need to specify the **Provider** for interfacing to the **Historian server**, the **Historian access server**, and also the name of the **Automation Integration server** through which the data access is channeled. The information is required if your master recipe or workflow holds phases for retrieving historical data.

TIP

Please note that changes made to Automation or Historian attributes of an equipment entity that is already in use on the shop floor are not automatically pushed to the Production Execution station that runs the process with the affected equipment entity. PharmaSuite for Production Execution needs to be restarted for the changes to become effective.

SPECIFICATION

The **Specification** tab lists the property types assigned as specification properties to the equipment entity.

TIP

Automation properties indicate if a tag path has been specified for them.
(-) means that they have no tag path,
(+) means that at least one tag path has been defined.
Please note, however, that the indicator only refers to the existence of a tag path and not to its validity.

Some entities require specific object assignments:

CONTAINER

Object type	Usage	Data type Content	Purpose
Property type, such as Container Type (RS)	Specification	EquipmentType Container (RS)	---
Property type, such as Container Tare (RS)	Runtime	MeasuredValue	Current Tare (RS)
Property type, such as Container Sublot (RS)	Runtime	String, used in container cleaning graph	Current Sublot (RS)
Property type, such as Container Clean Shelflife (RS)	Specification	Duration, used in container cleaning graph	---
Equipment graph, such as ContainerCleaning_RS_1	---	---	Container Cleaning (RS)

TIP

To define a reference tare for a container that can be accessed when taring the container during Dispense or Output Weighing, create a MeasuredValue runtime property type of the Reference Tare (RS) purpose.

Room

Object type	Usage	Data type Content	Purpose
Property type, such as Room Type (RS)	Specification	EquipmentType Room (RS)	---
Property type, such as Room Cleaning Rules (RS)	Runtime	RoomCleaningRules <room-specific>	---
Property type, such as Work Center Assignment (RS)	Specification	WorkCenterAssignment <room-specific>	---
Property type, such as General Clean Shelflife (RS)	Specification	Duration, used in room cleaning graph	---
Equipment graph, such as RoomCleaning_RS_1	---	---	Room Cleaning (RS)

Scale

Object type	Usage	Data type Content	Purpose
Property type, such as Scale Type (RS)	Specification	EquipmentType Scale (RS)	---
Property type, such as Scale Ranges (RS)	Specification	Ranges <scale-specific>	ScaleRanges (RS)
Property type, such as Scale Configuration (RS)	Specification	ScaleConfiguration <scale-specific>	---
Property type, such as Scale Test and Calibration (RS)	Specification	ScaleTestAndCalibration <scale-specific>	---
Property type, such as Work Center Assignment (RS)	Specification	WorkCenterAssignment <scale-specific>	---
Property type, such as Scale Test Shelflife (RS)	Specification	Duration, used in scale test graph	---
Property type, such as Scale Calibration Shelflife (RS)	Specification	Duration, used in scale calibration graph	---

Object type	Usage	Data type Content	Purpose
Property type, such as Scale Load (RS)	Runtime	String, required for handling of loaded scales	Current Load (RS)
Equipment graph, such as ScaleTest_RS_1	---	---	Scale Test (RS)
Equipment graph, such as ScaleCalibration_RS_1	---	---	Scale Calibration (RS)

- Click a property value to edit it. The system displays a text cursor or a button to access a suitable editor (page 10) for the required value type.

  TIP
  Runtime properties can change their values due to activities on the shop floor when the entity is in a read-only status, such as **Verification** or **Approved**. In order to resolve processing exceptions by administrative means, it is possible to update runtime data in these statuses in Data Manager as well. Before you change any runtime data, make sure to refresh the entity (page 72) so that it reflects the current situation on the shop floor.

- To add a property to the list, drag and drop (page 100) its property type on the specification tab.
- To change a property's position in the list, click the **Move up** or **Move down** buttons (page 74).
- To remove a property from the list, click the **Remove property** button (page 75).

PROCESS

The **Process** tab lists the property types assigned as process properties to the equipment entity.

TIP

Automation properties indicate if a tag path has been specified for them.
(-) means that they have no tag path,
(+) means that at least one tag path has been defined.
Please note, however, that the indicator only refers to the existence of a tag path and not to its validity.

- Click a property value to edit it. The system displays a text cursor or a button to access a suitable editor (page 10) for the required value type.

  TIP
  Runtime properties can change their values due to activities on the shop floor when the entity is in a read-only status, such as **Verification** or **Approved**. In order to resolve processing exceptions by administrative means, it is possible to update runtime data in these statuses in Data Manager as well. Before you change any runtime data, make sure to refresh the entity (page 72) so that it reflects the current situation on the shop floor.

- To add a property to the list, drag and drop (page 100) its property type on the process tab.
- To change a property's position in the list, click the **Move up** or **Move down** buttons (page 74).
- To remove a property from the list, click the **Remove property** button (page 75).

ENGINEERING

The **Engineering** tab lists the property types assigned as engineering properties to the equipment entity.

TIP

Automation properties indicate if a tag path has been specified for them.
(-) means that they have no tag path,
(+) means that at least one tag path has been defined.
Please note, however, that the indicator only refers to the existence of a tag path and not to its validity.

- Click a property value to edit it. The system displays a text cursor or a button to access a suitable editor (page 10) for the required value type.

  TIP
  Runtime properties can change their values due to activities on the shop floor when the entity is in a read-only status, such as **Verification** or **Approved**. In order to resolve processing exceptions by administrative means, it is possible to update runtime data in these statuses in Data Manager as well. Before you change any runtime data, make sure to refresh the entity (page 72) so that it reflects the current situation on the shop floor.

- To add a property to the list, drag and drop (page 100) its property type on the engineering tab.
- To change a property's position in the list, click the **Move up** or **Move down** buttons (page 74).
- To remove a property from the list, click the **Remove property** button (page 75).

GRAPH

The **Graph** tab lists the equipment graphs assigned to the equipment entity. Each graph shows with the current status of the entity. If the graph status can expire (page 191), the system displays its expiry date, if set. If the status has already expired, the system indicates this by marking the displayed (expired) status in strikethrough formatting.

- To add a graph to the list, drag and drop (page 100) its card or tile on the Details window of the entity to which you wish to assign it.

  TIP
  Please note that you can only assign graphs that do not have unsaved changes and contain at least one status.

- To change the current status of the entity and its expiry date, which are controlled by a graph, open the GraphStatusChange editor (page 58) of the respective graph and make the required changes.

  TIP
  Please note that changing the status of a graph forces this status and does not perform a transition into the status. This means that actions that would be performed during a transition, such as updating the retest date or a counter do not take place and need to be performed manually.

- To trigger a graph transition on the entity, thus performing all included actions and observing all conditions, click the **Trigger graph transition** button (page 75) to open the **Trigger Graph Transition** dialog (page 139).

  TIPS
  Graph statuses and their expiry dates are runtime data that can change due to activities on the shop floor when the entity is in a read-only status, such as **Verification** or **Approved**. At regular intervals, the system checks all equipment entities with expirable statuses to determine if statuses have expired. If that is the case, it performs the status transitions as defined by the graph, but only on entities that are neither locked in Data Manager nor bound by execution.
  In order to resolve processing exceptions by administrative means, it is possible to update runtime data in these statuses in Data Manager as well. Before you change any runtime data, make sure to refresh the entity (page 72) so that it reflects the current situation on the shop floor.

- To change a graph's position in the list, click the **Move up** or **Move down** buttons (page 74).
- To remove a graph from the list, click the **Remove graph** button (page 75).

CLASS

The **Class** tab lists the equipment classes assigned to the equipment entity. An equipment entity can belong to several equipment classes, but takes its styling (color, icon, properties displayed on card view) from the class set as its **Style class**.

- To add a class to the list, drag and drop (page 100) its card or tile on the Details window of the entity to which you wish to assign it.

  > TIP
  > Please note that the system only checks if the properties of the assigned class correspond to those of your entity, but not on which tabs (**Specification**, **Process**, **Engineering**) the properties are located. Thus, the system does not rearrange the entity's properties to match the properties' positioning on the class.

- To define the styling of the entity by setting an assigned class a style class, select the **Style class** checkbox of the class whose styling the entity is to adopt.
- To remove a class from the list, select or multi-select it and click the **Remove equipment class** button (page 75) or press the DEL key.
  Removing a class from the list of classes also removes the entity from the list of entities shown on the **Entity** tab of the class' detail view.

GROUPING

The **Grouping** tab displays the grouping situation of the entity in two lists.

- The **Parent** list displays the data of the entity's parent entity. An entity can have only one parent, so the list always contains only one entry with the following data:
  - Identifier
  - Short description
  - Description
  - Status

  > TIP
  > The **Parent** list of an entity is always read-only. To change the parent assignment of an entity in Data Manager, you have to open the parent entity in the Details window and remove the respective child entity from its **Children** list.

- The **Children** list displays the data of all children that are directly or indirectly assigned to the entity. Indirect assignments occur when you assign an entity as child that is already a parent entity to other child entities. The following data is shown for child entities:
    - Identifier
    - Short description
    - Description
    - Status
    - Sub-parent
      Only available for entities that were assigned indirectly when their indicated parent was assigned to the entity.

> TIP
> Please note that you can only remove child entities from the list that have been assigned directly. For removing an indirectly assigned child entity you have to open its parent entity (indicated in the **Sub-parent** column) and remove the respective child entity from its **Children** list.

- To add a child entity to the list, drag and drop (page 100) its card or tile on the Details window of the entity to which you wish to assign it.
- To remove a child entity from the list, select or multi-select it and click the **Remove child entity** button (page 75) or press the DEL key.

> TIP
> Group assignments are runtime data that can change due to activities on the shop floor when the entity is in a read-only status, such as **Verification** or **Approved**. In order to resolve processing exceptions by administrative means, it is possible to update runtime data in these statuses in Data Manager as well. Before you change any runtime data, make sure to refresh the entity (page 72) so that it reflects the current situation on the shop floor.
> A typical example of a group assignment change is a repair situation on the shop floor where an entity that is part of a group malfunctions during order processing and needs to be replaced.

Context

The **Context** tab lists the current and the previous material-related context information of the equipment entity, such as

- the product batch and material for which the entity was used,
- the order and order step during whose execution the entity was used,
- the involved material that was in direct contact with the entity,
- the order step input and order step output of the order step for which the entity was used.

TIP

Please note that the **Context** tab only contains the material-related context information for orders, not for workflows.

The information is read-only and shows the data for the current usage if the entity is bound. If it is not bound it shows the data for the entity's last usage. The complete usage history of the entity is recorded on the **Logbook** tab (page 183).

Logbook

The **Logbook** tab provides a table that lists all processing-related events in which the equipment entity was involved. For each event that affected the entity, the table displays one row.

To filter the event entries by date, select the date range you wish to consider. The system only displays the entries that have been registered between the two dates you have selected.
The date range starts at 12:00:00 AM on the date you have selected and ends at 23:59:59 PM on the date you have selected.
When you open the logbook, the date range is preset to start with the date when the last event occurred and to end with the current date. If the system has no events registered for the entity, the date range is preset to the current date.
To refresh the list of events, click the **Refresh** button (page 75). All events of the entity that have occurred since you opened the logbook are then included.
To print the **Equipment Logbook Report** for the events that have occurred during the selected date range, click the **Print equipment logbook report** button (page 74).

The following table columns are available and are filled if relevant to the respective event:

- Timestamp
  indicates the date and time when the event occurred.
- Category
  indicates the type of the event (**Binding**, **Denial**, **FSM**, **Context**, **Manual**, **Phase**, **Property update**, **Graph transition**, **Grouping**).
- Action/trigger
  for events that were triggered by external circumstances in the life cycle of the entity, the action that triggered the event (such as **Bind**, **Identify**) or any trigger of an assigned equipment graph.

> TIP
> Please note that for events of the **Graph transition** category, several trigger actions may be performed that change the same runtime property value. In this case, the logbook only records the initial and the final values of the property.

- Graph
  for events of the **Graph transition** category, indicates the graph that triggered the event.
- Property
  for events of the **FSM** category, the identifier of the FlexibleStateModel property type that governs the registered status change.
- Old content/New content
  indicates the data changed by an event, such as **Identified**/**Bound** for an event of the **Binding** category.
- Information
  for events of the **Denial**, **Manual**, **Graph transition**, **Property update**, or **Grouping** categories, additional information on the circumstances of the event.
- Workflow
  for equipment entities used for processing a workflow, identifier of the workflow.
- Order
  for equipment entities used for processing an order, identifier of the order.
- Product batch
  for equipment entities used in batch production, the identifier of the batch that was being produced when the event occurred.
- Product material
  for equipment entities used in production, the identifier of the material that was being produced when the event occurred.

- Unit procedure
  identifier of the unit procedure (with its count if it was run more than once) that was being executed when the event occurred.
- Operation
  identifier of the operation (with its count if it was run more than once) that was being executed when the event occurred.
- Phase
  identifier of the phase (with its count if it was run more than once) that was being executed when the event occurred.
- Involved material
  identifier of the material that was in direct contact with the entity when the event occurred.
- Work center
  identifier of the work center where the event occurred.
- Station
  identifier of the station where the event occurred.
- Device
  identifier of the device where the event occurred.
- Signature
  displays two items per signature (page 101):
  - User and login names of the user who performed the signature.
  - Timestamp when the signature was recorded.
- Logged-in user
  login name of the user under whose login the event occurred.
  If the logbook entry is created due to an update triggered by an external system, such as Warehouse Management, the user provided by the external system is displayed.

MIGRATED HISTORY

The **Migrated History** tab only exists for equipment entities that previously existed as data objects in PharmaSuite for Production Management and were migrated to Data Manager. It only contains the data recorded before the entity was migrated. Any events that occur after the migration are recorded in the entity's Logbook (page 183), Change History (page 186), and Status History (page 187) tabs.
A migrated history consists of two panels, one for the **Logbook** table at the top and the other one for the **Status Histories** table underneath, which comprises all status history event entries that were recorded for all status graphs that applied to the entity before it was migrated. So, for a scale, it contains both the event entries from the calibration status history and from the testing status history.

The panels are divided by a slidable separator and can thus be freely resized, if necessary.

To filter the change event entries by date, select the date range you wish to consider. The system only displays the entries that have been registered between the two dates you have selected.
The date range starts at 12:00:00 AM on the date you have selected and ends at 23:59:59 PM on the date you have selected.
To refresh the list of events, click the **Refresh** button (page 75).

CHANGE HISTORY

The **Change History** tab provides a list of all changes that have been performed on the equipment entity, which also includes indirect assignment changes that occur when a class' assignment change also involves the entity.
For each change event that affected the entity, the table displays one row.

To filter the change event entries by date, select the date range you wish to consider. The system only displays the entries that have been registered between the two dates you have selected.
The date range starts at 12:00:00 AM on the date you have selected and ends at 23:59:59 PM on the date you have selected.
When you open the change history, the date range is preset to start with the date when the last change event occurred and to end with the current date. If the system has no change events registered for the object, the date range is preset to the current date.
To refresh the list of change events, click the **Refresh** button (page 75). All change events involving the entity that have occurred since you opened the change history are then included.

The following table columns are available and are filled if relevant to the respective change event:

- Timestamp
  indicates the date and time when the change event occurred.
- Action
  indicates the change event that triggered the entry in the change history.
- Affected object
  indicates the object that is affected by the change event. So, when you update the basic data of your entity, the change history shows the identifier of the attribute you have modified and when you assign a class, graph, or property type to your entity, the change history shows the identifier of the assigned class, graph, or property type.
- Old content/New content
  for data change events on your entity, displays the changed values.
- Information
  indicates additional details on the change, such as if it was performed in the context of a mass change or a tag or point import.
- Logged-in user
  login name of the user under whose login the change event occurred.

STATUS HISTORY

The **Status History** tab provides a list of all status changes that have been performed on the equipment entity.
For each status change that affected the entity, the table displays one row.

To filter the status change entries by date, select the date range you wish to consider. The system only displays the entries that have been registered between the two dates you have selected.
The date range starts at 12:00:00 AM on the date you have selected and ends at 23:59:59 PM on the date you have selected.
When you open the status history, the date range is preset to start with the date when the last status change occurred and to end with the current date. If the system has no status changes registered for the object, the date range is preset to the current date.
To refresh the list of status changes, click the **Refresh** button (page 75). All status changes involving the entity that have occurred since you opened the status history are then included.

The following table columns are available and are filled if relevant to the respective status change:

- Timestamp
  indicates the date and time when the status change occurred.
- Action
  indicates the status change that triggered the entry in the status history.
- To status
  indicates the status to which the entity was moved.
- Signature
  displays two items per signature (page 101):
  - User and login names of the user who performed the signature.
  - Timestamp when the signature was recorded.
- Comment
  displays the comment recorded for each signature. For double signatures, each comment is prefixed with **Comment 1** or **Comment 2** to indicate with which signature the comment was recorded.

Graph

The Details window of a graph contains the following data tabs:

- Basic (page 189)
- Property Type (page 191)
- Status/Trigger (page 191)
- Transition (page 193)
- Style (page 197)
- Change History (page 197)
- Status History (page 198)

BASIC

The **Basic** tab lists the Basic Attributes of the equipment graph.

Click an attribute value to edit it. The system displays a text cursor or a button to access a suitable editor (page 10) for the required value type.

TIP

Please note that once you have created an equipment graph, you can no longer modify its **Identifier**.

DEFAULT GRAPHS

PharmaSuite provides the following equipment graphs with any default installation of the system:

- ContainerCleaning_RS_1
  The default graph for container cleaning includes the following default property types:
  - Container Clean Shelflife (RS)
  - Container Sublot (RS)
- RoomCleaning_RS_1
  The default graph for room cleaning includes the following default property type:
  - General Clean Shelflife (RS)
- ScaleCalibration_RS_1
  The default graph for scale calibration includes the following property type:
  - Scale Calibration Shelflife (RS)
- ScaleTest_RS_1 (scale test graph)
  The default graph for scale test includes the following property type:
  - Scale Test Shelflife (RS)

The **Display text** attribute defines the text shown on the card view of the graph.

The **Purpose** attribute is mandatory and indicates the context for which the graph is intended.
Especially the Dispense and Weighing phases of PharmaSuite require equipment to hold graphs of specific purposes in order to function correctly. The Dispense and Weighing-related purposes are marked with **(RS)**.
The following purposes are available:

- **Calibration**
  for graphs related to the calibration of equipment.
- **Cleaning**
  for graphs related to the cleaning of equipment.
- a **Container Cleaning (RS)** graph is required to control that vessels identified during production are fit for use.
- **Installation**
  for graphs related to installation processes for equipment.
- **Maintenance**
  for graphs related to maintenance processes for equipment.
- a **Room Cleaning (RS)** graph is required to calculate to which extent a room needs to be cleaned prior to a weighing process.
- a **Scale Calibration (RS)** graph is required to qualify scales prior to a weighing process.
- a **Scale Test (RS)** graph is required to qualify scales prior to a weighing process.
- **Sterilization**
  for graphs related to the sterilization of equipment.
- **Testing**
  for graphs related to testing processes for equipment.
- **Usage**
  for graphs related to the usage of equipment.
- If PharmaSuite is configured to communicate with Warehouse Management (page 123), a **Warehouse Load Check (RS)** graph is required to enable Warehouse Management to check if loading a logistic unit is allowed by to the status of its corresponding equipment entity.

Property Type

The **Property Type** tab lists the property types assigned to the equipment graph.

- To add a property type to the list, drag and drop (page 100) its card or tile on the Details window of the graph to which you wish to assign it.
- To change a property type's position in the list, click the **Move up** or **Move down** buttons (page 74).
- To display a property type on the card view of the equipment graph, click the **Display on card** button (page 73).
- To remove a property type from the list, click the **Remove property type** button (page 75).

Status/Trigger

The **Status/Trigger** tab provides two collapsible lists to define

- the individual statuses of the equipment graph consisting of
    - the **Key**, which is mandatory and must be unique within the list of statuses,
    - the **Display text**, which is how the status is shown to operators during processing,
    - a **Description** to further elaborate the status's meaning, and
    - the **Initial status** option, which defines which status an equipment entity receives initially when the graph is assigned to it.
    - the **Can expire** option, which defines if a status can expire automatically after a specified duration or at a specific date.
- the triggers that affect the graph's statuses, consisting of
    - the **Key**, which is mandatory and must be unique within the list of triggers,
    - the **Display text**, which is how the trigger is shown to operators during processing, and
    - a **Description** to further elaborate the trigger's meaning.

TIP
Please note that the system automatically adds the **Expired (RS)** trigger as soon as you select the **Can expire** option for one of the graph's statuses. You cannot manually delete the system-provided **Expired (RS)** trigger, but it is removed automatically when you have unselected the **Can expire** option of all statuses.

To add a new status or trigger, type its data in the empty table row, starting with its key. The system automatically adds a new empty list row after you have completed the first cell of the new status or trigger.

> TIP
> Please note that curly brackets (braces) and leading or trailing blanks are not allowed in the keys of statuses or triggers.

To delete a status or trigger, select or multi-select it and click the **Delete status** or **Delete trigger** button, respectively, or press the DEL key.

> TIP
> Statuses or triggers that contain errors are indicated by an error marker (page 76) to the left of the respective rows. Hover over the icon to view a tooltip with additional error details.

Load and Unload Synchronization with Warehouse Management

If PharmaSuite is configured to communicate with Warehouse Management, equipment entities that are movable and hold a sublot are typically represented as a logistic unit in Warehouse Management.

To check if loading or unloading a warehouse-managed logistic unit is permitted by the status of its corresponding equipment entity with sublots, a graph of the **Warehouse Load Check (RS)** purpose must be assigned to those entities that correspond to logistic units. The graph must contain transitions to model the situation when the first load is placed on the logistic unit and when the last load is removed from the logistic unit.
For these transitions the system expects two triggers with the following trigger keys:

- **FIRST_LOAD**
  Loading an empty logistic unit with a sublot in Warehouse Management is only completed if a graph transition, caused by the **First load** trigger of an equipment graph with the **Warehouse Load Check (RS)** purpose, has been performed successfully on the equipment entity.
- **TOTAL_UNLOAD**
  Removing the last sublot from a logistic unit in Warehouse Management is only completed if a graph transition, caused by the **Total unload** trigger of an equipment graph with the **Warehouse Load Check (RS)** purpose, has been performed successfully on the equipment entity.

Transition

The **Transition** tab provides a list to specify the structure and logic of the equipment graph by combining

- a **Trigger**, which needs to have been defined on the **Status/Trigger** tab (page 191) before.
- the statuses (**From status** and **To status**), also previously defined on the **Status/Trigger** tab (page 191).
- an optional transition **Condition** that determines if the expected circumstances are met so that the transition can be executed. A condition is defined as condition expression in the Expression editor for transition conditions (page 30).
  The **Condition** column of a transition is blank if there is no condition defined. If a transition condition exists, it shows the description of the condition. If the condition does not have a description, the system displays the condition expression itself, introduced by an equals sign.

  TIP
  In the transition table, the description of a condition is not editable. To change it, you have to open the Expression editor and make your changes in the **Description** box.

- one or more **Actions** that are performed along with the transition. Actions are also optional and are displayed in a collapsible sub-table under the transition row. They are defined in the Expression editor for transition actions (page 31).

  TIP
  Actions are executed in the order in which they are listed. Use the **Move action down/up** buttons (page 74) to change the order of the actions.

- the identifier (**ID**) of the transition, which is shown for exception cases during execution, in the batch report, and in the logbook and change history of affected equipment entities.

Thus, a transition specifies the cause, represented by the **Trigger**, that initiates the transition from one status, the **From status**, to another status, the **To status**, including potential preconditions and resulting actions.

GRAPH TRANSITION MODELING WITH EXPIRY DATES AND TRIGGERS

In addition to the triggers that you define yourself, the system creates the **Expired (RS)** trigger, when you define a status that can expire. Along with the trigger the system provides an **ExpiryDate** reference for use in the Expression editor in conditions or actions.

Expiry triggers are evaluated automatically during execution whenever an operator identifies an equipment entity in order to update its status and thus ensure that the entity is suitable for use. If the trigger evaluation determines that the identified entity has expired into a status that is unfit for use, the system rejects the entity. In this case, the entity has to undergo a specific treatment that returns it to the suitable status.
In the case of a container, this could mean that its "Clean" status has expired, and it needs to be run through a cleaning workflow. In the case of a scale, its "Tested" or "Calibrated" statuses may have expired so that the scale requires testing and calibration before it can be used in production again.

Your graph may contain several statuses that can expire, so that one status expires into a successor status that again expires into a successor status, thus forming a cascade of expiries until the graph has reaches a status that cannot expire or has not yet expired.

TIP

For creating the transitions of a graph with expiry-enabled statuses, we recommend to proceed as follows:

1. Create all transitions required to cover all statuses using all user-created triggers.
2. Identify which transitions have an expiry-enabled **To status** and add a **set** action that defines an expiry date.
3. Use the **Expired (RS)** trigger to create transitions out of expiry-enabled statuses, which have not been handled yet. This may be necessary to define various conditions that lead to different target statuses.
   Make sure to create an action for setting an expiry date when a transition has an expiry-enabled **To status**.

The following example shows a cleaning graph for equipment entities with four statuses, three of which can expire:

- **To be cleaned** (UNCLEAN)
  is the initial status and cannot expire. From here you can only transition into the **Clean** status.
- **Clean** (CLEAN)
  can expire. From here you can transition to the **To be cleaned**, the **Clean**, the **In use**, and the **Reusable** statuses.
- **In use** (IN_USE)
  can expire. From here you can transition to the **Reusable** and **To be cleaned** statuses.

- **Reusable** (REUSABLE)
  can expire. From here you can transition to the **In use** and **To be cleaned** statuses.

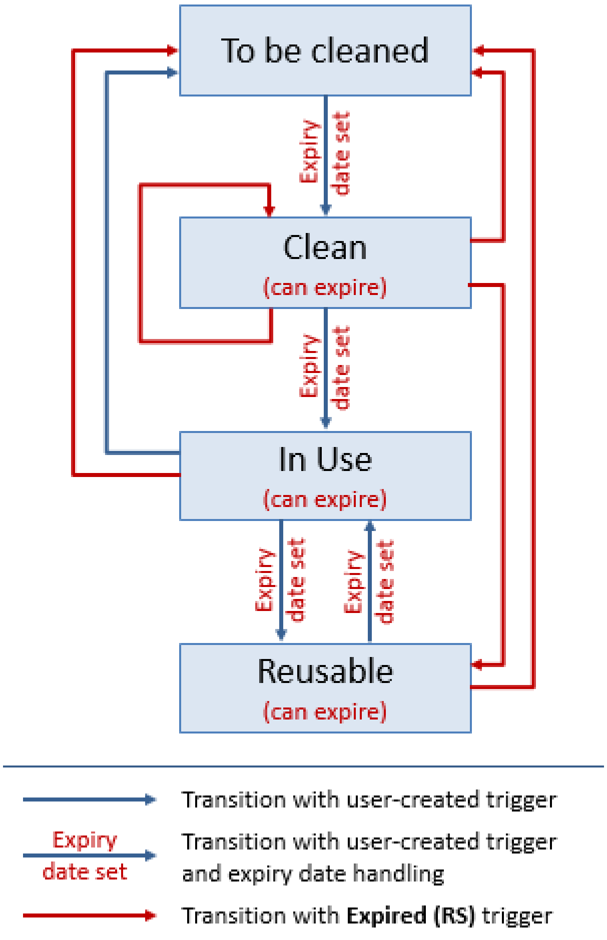


Figure 106: Example graph

The expiry dates of explicit transitions in the example are handled as follows:

- Setting the expiry date
  The expiry date for the **Clean** status is specified as action with the transition from **To be cleaned** to **Clean**. It uses the `set` command to define that the `{ExpiryDate}` reference is set to twelve hours after the date and time when the transition occurs. To retrieve the time, the action expression uses the **Current Date and Time (now)** function from the **Analysis** section.

```
set {ExpiryDate} = now() + 12h
```

- Retaining the expiry date
  To retain the expiry date of the equipment entity when it moves from the **Clean** or **Reusable** statuses to the **In use** status, the `{ExpiryDate}` reference is set to itself:
  ```
  set {ExpiryDate} = {ExpiryDate}
  ```
- Postponing the expiry date
  When the equipment entity moves from the **In use** status to the **Reusable** status, its expiry date is postponed by two hours:
  ```
  set {ExpiryDate} = {ExpiryDate} + 2h
  ```

Implicit transitions with **Expired (RS)** triggers handle the following cases:

> TIP
> When calculating a new expiry date in a transition of the **Expired (RS)** trigger, we recommend not to use the **now()** function. Since the system executes the expiry trigger when an (expired) equipment entity is identified during production, which is not necessarily a predictable point in time and is thus not really suitable as base of a calculation. The expiry date of the **From status**, on the other hand, is a defined and immutable point in time, which makes it a good base for calculating the expiry date of the **To status**:
> ```
> set {ExpiryDate} = {ExpiryDate} + <duration>
> ```

- Prolonging the time before an equipment entity is unavailable for use due to an expired status
  Under a specific condition, the **Clean** status expires into itself, adding time to its set expiry date:
  ```
  set {ExpiryDate} = {ExpiryDate} + 12h
  ```
- Prolonging the time before an equipment entity is unavailable for use after it has expired into another status
  Under a specific condition, the **Clean** status expires into the **Reusable** status, adding time to its set expiry date:
  ```
  set {ExpiryDate} = {ExpiryDate} + 2h
  ```
- Automatically moving an equipment entity to a status that makes it unavailable for use
  Under specific conditions, the **Clean**, **In use**, or **Reusable** statuses expire into the **To be cleaned** status, which cannot expire and thus does not require an action that handles an expiry date.

To add a new transition, select the required entries from the respective option lists or start to define a new condition. The system automatically adds a new empty list row after you have completed any cell of the new transition.

To delete a transition, select or multi-select it and click the **Delete transition** button or press the DEL key.

> TIP
> Transitions that contain errors are indicated by an error marker (page 76) to the left of the respective table or sub-table rows. Hover over the icon to view a tooltip with additional error details.

Style

The **Style** tab provides tools for defining the visual appearance of the equipment graph's tiles and cards:

- Background color
- Font color

Change History

The **Change History** tab provides a list of all changes that have been performed on the equipment graph.
For each change event that affected the graph, the table displays one row.

To filter the change event entries by date, select the date range you wish to consider. The system only displays the entries that have been registered between the two dates you have selected.
The date range starts at 12:00:00 AM on the date you have selected and ends at 23:59:59 PM on the date you have selected.
When you open the change history, the date range is preset to start with the date when the last change event occurred and to end with the current date. If the system has no change events registered for the object, the date range is preset to the current date.
To refresh the list of change events, click the **Refresh** button (page 75). All change events involving the graph that have occurred since you opened the change history are then included.

The following table columns are available and are filled if relevant to the respective change event:

- Timestamp
  indicates the date and time when the change event occurred.
- Action
  indicates the change event that triggered the entry in the change history.

- Affected object
  indicates the object that is affected by the change event. So, when you update the basic data of your graph, the change history shows the identifier of the attribute you have modified. When you assign a property type to your graph, the change history shows the identifier of the assigned property type. For status or trigger changes, the affected object is displayed as <key>/<display text>. Similar formatting is used for transition attributes which show as <ID>/<attribute>.
- Old content/New content
  for data change events on your graph, displays the changed values.
- Logged-in user
  login name of the user under whose login the change event occurred.

Status History

The **Status History** tab provides a list of all status changes that have been performed on the equipment graph.
For each status change that affected the graph, the table displays one row.

To filter the status change entries by date, select the date range you wish to consider. The system only displays the entries that have been registered between the two dates you have selected.
The date range starts at 12:00:00 AM on the date you have selected and ends at 23:59:59 PM on the date you have selected.
When you open the status history, the date range is preset to start with the date when the last status change occurred and to end with the current date. If the system has no status changes registered for the object, the date range is preset to the current date.
To refresh the list of status changes, click the **Refresh** button (page 75). All status changes involving the graph that have occurred since you opened the status history are then included.

The following table columns are available and are filled if relevant to the respective status change:

- Timestamp
  indicates the date and time when the status change occurred.
- Action
  indicates the status change that triggered the entry in the status history.
- To status
  indicates the status to which the graph was moved.
- Signature
  displays two items per signature (page 101):
  - User and login names of the user who performed the signature.
  - Timestamp when the signature was recorded.

- Comment
  displays the comment recorded for each signature. For double signatures, each comment is prefixed with **Comment 1** or **Comment 2** to indicate with which signature the comment was recorded.

PROPERTY TYPE

Property types are used on the one hand to model the properties inherent to equipment entities and classes and on the other hand to define the equipment requirements presented by master recipes and workflows. A property type describes a physical characteristic of an equipment entity, such as its volume, speed range, or the fact that it is under cleaning status control. For this reason, each property type can only be assigned once to an equipment entity, template entity, or class.
As a consequence, when you model an equipment entity that holds two tanks of the same volume, you still need to define two property types, one for each tank. Similarly, you also need to define two property types to model different functions the volume of one and the same tank may have, such as its storage volume, which may be greater than its fermentation volume.

During execution, an equipment entity is matched against the requirements defined with the recipe or workflow for which an operator plans to use it. Only if this validation check concludes that the properties of the equipment entity meet the requirements can the entity be used as planned.

VALIDATION RULES

The rules that determine if an equipment entity matches its requirements take place on three levels of complexity:

- The basic level only has one rule, which checks if all properties defined for the requirement are also defined as properties of the equipment entity.
- The second level has a set of rules that apply if requirement and entity both state a value for a property, if one of them states a range while the other supplies a value, or if one or both of them state neither a value nor a range for a property.

Condition	Result
Requirement: value and range are not defined	The entity is a **valid** match for the requirement.
Requirement: value or range is defined **Entity:** value and range are not defined	The entity is **no valid** match for the requirement.

Condition	Result
Requirement: value is defined (ReqValue) **Entity:** value is defined (EntValue) ReqValue = EntValue	The entity is a **valid** match for the requirement.
Requirement: range is defined (ReqLowLimit and/or ReqHighLimit) **Entity:** value is defined (EntValue) ReqLowLimit <= EntValue <= ReqHighLimit	The entity is a **valid** match for the requirement.
Requirement: value is defined (ReqValue) **Entity:** range is defined (EntLowLimit and/or EntHighLimit) EntLowLimit <= ReqValue <= EntHighLimit	The entity is a **valid** match for the requirement.
Requirement: value is not defined **Entity:** value is not defined **Either requirement or entity:** one value (Low or High) of range is defined	Covered by third level of rules, see below.
Either requirement or entity: value is defined range is defined (Low and/or High)	This condition represents an invalid configuration and is prevented by the system.

- The set of rules on the third level applies to cases when both requirement and entity have a range defined for a property.
  When matching the range of an entity (EntLowLimit, EntHighLimit) against the range of a requirement (ReqLowLimit, ReqHighLimit) the system has to consider the following possible cases:
  - Entity range is **lower** than requirement range:
    EntHighLimit < ReqLowLimit
  - Entity range **overlaps** requirement range at the **low limit**:
    EntLowLimit < ReqLowLimit <= EntHighLimit <= ReqHighLimit
  - Entity range **is contained** in requirement range:
    ReqLowLimit <= EntLowLimit <= EntHighLimit <= ReqHighLimit

- Entity range **contains** requirement range:
  EntLowLimit <= ReqLowLimit <= ReqHighLimit <= EntHighLimit
- Entity range **overlaps** requirement range at the **high limit**:
  ReqLowLimit <= EquLowLimit <= ReqHighLimit < EquHighLimit
- Entity range is **higher** than requirement range:
  ReqHighLimit < EntLowLimit

The following table lists how the system evaluates the settings made for entity and requirement ranges and indicates the match results.
If a range only has one limit, the system treats it as open range.

Limit settings specified				Range check type of requirement	
ReqLow	ReqHigh	EntLow	EntHigh	Is contained in entity range	Contains entity range
---	---	---	---	Valid match	Valid match
---	---	---	Yes	Valid match	Valid match
---	---	Yes	---	Valid match	Valid match
---	---	Yes	Yes	Valid match	Valid match
---	Yes	---	---	No valid match	No valid match
---	Yes	---	Yes	Valid match, if EntHigh <= ReqHigh	Valid match, if ReqHigh <= EntHigh
---	Yes	Yes	---	No valid match	No valid match
---	Yes	Yes	Yes	Valid match, if EntHigh <= ReqHigh	No valid match
Yes	---	---	---	No valid match	No valid match
Yes	---	---	Yes	No valid match	No valid match
Yes	---	Yes	---	Valid match, if ReqLow <= EntLow	Valid match, if EntLow <= ReqLow
Yes	---	Yes	Yes	Valid match, if ReqLow <= EntLow	No valid match
Yes	Yes	---	---	No valid match	No valid match
Yes	Yes	---	Yes	No valid match	Valid match, if Equ.max <= Req.max
Yes	Yes	Yes	---	No valid match	Valid match, if Req.min <= Equ.min

Limit settings specified				Range check type of requirement	
ReqLow	**ReqHigh**	**EntLow**	**EntHigh**	**Is contained in entity range**	**Contains entity range**
Yes	Yes	Yes	Yes	Valid match, if ReqLow <= EntLow **AND** EntHigh <= ReqHigh	Valid match, if EntLow <= ReqLow **AND** ReqHigh <= EntHigh

The Details window of a property type contains a **Basic** tab that lists its Basic Attributes.

Click an attribute value to edit it. The system displays a text cursor or a button to access a suitable editor (page 10) for the required value type.

TIP

Please note that once you have created, defined, and saved a property type, you cannot edit it anymore.

DEFAULT PROPERTY TYPES

PharmaSuite provides the following property types with any default installation of the system:

- Container Clean Shelflife (RS)
  a Specification property type of the Duration data type
- Container Sublot (RS)
  a Runtime property type of the String data type and the Current Sublot (RS) purpose
- Container Tare (RS)
  a Runtime property type of the MeasuredValue data type and the Current Tare (RS) purpose
- Container Type (RS)
  a Specification property type of the Container (RS) equipment type
- General Clean Shelflife (RS)
  a Specification property type of the Duration data type
- Room Cleaning Rules (RS)
  a Specification property type of the RoomCleaningRules data type
- Room Type (RS)
  a Specification property type of the Room (RS) equipment type
- Scale Calibration Shelflife (RS)
  a Specification property type of the Duration data type

- Scale Configuration (RS)
  a Specification property type of the ScaleConfiguration data type
- Scale Load (RS)
  a Runtime property type of the String data type and the Current load (RS) purpose
- Scale Ranges (RS)
  a Specification property type of the Ranges data type and the ScaleRanges (RS) purpose
- Scale Test and Calibration (RS)
  a Specification property type of the ScaleTestAndCalibration data type
- Scale Test Shelflife (RS)
  a Specification property type of the Duration data type
- Scale Type (RS)
  a Specification property type of the Scale (RS) equipment type
- Work Center Assignment (RS)
  a Specification property type of the WorkCenterAssignment data type

The purpose and context of a property type are defined by several attributes:

- with the **Usage** attribute you determine if a property type
  - is part of the basic **Specification** of the equipment, indicating a characteristic of an equipment object, such as a coater's drum size,
  - is intended for equipment in an **Automation** environment, so that you need to specify automation-related data, such as tag names, tag update rates, or Live Data types,
  - if it refers to a **Runtime** value of an equipment entity that can change during processing, such as a coater's cleaning status,
  - or is intended for equipment that is connected to a **Historian** server, so that you need to specify which tags are to be read for retrieving the historical data chart.
- with the **Purpose** attribute you can further specify some data types that represent required properties for equipment to be used with the Dispense and Weighing phases of PharmaSuite or with phases that generate equipment entities on the shop floor.
  - the **Base Sublot (RS)** purpose extends the String (page 207) data type in the **Runtime** usage and is required to register the identified sublot so that the system can determine which template equipment entity to use when generating an entity.

  - the **Current Tare (RS)** purpose extends the MeasuredValue (page 206) data type in the **Runtime** usage and is required to handle the tare property of a container.
  - the **Reference Tare (RS)** purpose extends the MeasuredValue (page 206) data type in the **Runtime** usage and is used for defining the optional reference tare property of a container.
  - the **Current Sublot (RS)** purpose extends the String (page 207) data type in the **Runtime** usage and is required for identifying a container.
  - the **Scale Ranges (RS)** purpose extends the Ranges (page 206) data type and is required to configure the ranges of a scale.
  - the **Current Load (RS)** purpose extends the String (page 207) data type in the **Runtime** usage and is required to handle the identifier of a container or sublot that represents a scale load.

- with the **Data type** attribute you determine the context for which the property type is suitable. Once you have selected the **Data type**, you can specify its **Content**.

  The system provides the following data types:

  - BigDecimal
    floating point number, used for numbers and ranges. It is suitable for handling numbers qualified by a unit of measure.
    BigDecimal for automation can handle the three Live Data types: **Double**, **Float**, and **Integer** that need to be specified for correct communication with equipment. The mapping of the most common OPC names to the Live Data types is as follows:

    - **Integer** represents SHORT, LONG, CHAR, BYTE, USHORT, and ULONG
    - **Float** represents REAL4
    - **Double** represents REAL8

TIP
BigDecimal values support up to 15 integral digits and are rounded to 9 fractional digits when stored to the database.

    For editing BigDecimal values, the system provides several BigDecimal editors:

    - BigDecimal (page 11)
    - Automation BigDecimal (Double) (page 13)
    - Automation BigDecimal (Float) (page 17)

  - Automation BigDecimal (Integer) (page 20)
  - Historian BigDecimal (page 24)

- Boolean
  with the values **Yes**, **No**, and in some cases **N/A**.
  For editing Boolean values for automated equipment, the system provides the Automation Boolean (page 25) editor.

- Duration
  only available for **Specification** and **Runtime** properties.
  It is used for displaying time spans and for time-related calculations.
  It supports the following time units: days (**d**), hours (**h**), minutes (**min**), seconds (**s**), and milliseconds (**ms**).
  For editing durations, the system provides the Duration (page 29) editor.

- EquipmentType
  only available for **Specification** property types.
  It is used to support you with the accurate and complete configuration of equipment entities (page 123) to be used for executing product phases, such as Dispense and Weighing phases.

TIP
When an entity holds an EquipmentType property, the system checks if it also holds all other properties or graphs required by its type. You can save an entity that does not hold all required properties or graphs, but you cannot change its status to **Verification** or **Approved**.

  The following equipment types are available:

  - Container (RS)
  - Hybrid (RS)
  - Room (RS)
  - Scale (RS)

- FlexibleAttributeDefinition
  only available for **Runtime** property types.
  It is used to define a bundle of runtime attributes, whose values are established and can change during processing.
  For defining the attribute bundle, the system provides the FlexibleAttributeDefinition (page 53) editor.

  - FlexibleStateModel
    only available for **Runtime** property types.
    It is used to put equipment entities under status control, for example, to model cleaning or maintenance cycles.
    Suitable FSMs must be maintained in the Process Designer application of FactoryTalk ProductionCentre. For further information on FSMs for use with equipment, please refer to "Configuring FSMs for Equipment Properties" in Volume 3 of the "Technical Manual Configuration and Extension".
    For selecting a FlexibleStateModel for equipment, the system provides the FlexibleStateModel (page 55) editor.

  - FlexibleTagDefinition
    only available for **Automation** property types.
    It is used to define a group of automation-relevant values to communicate as value bundle from an equipment entity for monitoring purposes.
    For defining the value bundle, the system provides the Automation FlexibleTagDefinition (page 56) editor.

- MeasuredValue
  used for displaying numeric values qualified by a unit of measure.

TIP
MeasuredValue values support up to 15 integral digits and are rounded to 9 fractional digits when stored to the database.

For editing MeasuredValue data types, the system provides the MeasuredValue (page 59) editor.

  - Ranges
    only available for **Specification** property types.
    It is used to define up to three ranges for an equipment object.
    For defining the ranges, the system provides the Ranges (page 60) editor.

  - RoomCleaningRules
    only available for **Specification** property types.
    It is used to define the cleaning rules and cleaning demands of rooms used in a Dispense environment.
    For cleaning rule data, the system provides the CleaningRules (page 26) editor.

  - ScaleConfiguration
    only available for **Specification** property types.
    It is used to define the driver and connection data of a scale.
    For defining the scale data, the system provides the ScaleConfiguration (page 61) editor.

- ScaleTestAndCalibration
  only available for **Specification** property types.
  It is used to define the test weight and operator instructions for scale tests and calibration.
  For scale test and calibration data, the system provides the ScaleTestAndCalibration (page 65) editor.

- String
  used for handling any sequence of characters.

- WorkCenterAssignment
  only available for **Specification** property types.
  It is used to assign work centers to scales or rooms. Work centers are maintained in the Data Manager - Work Center mode of Data Manager.
  For selecting a work center, the system provides the Searchable Option List (page 67) editor.

Managing Equipment - Basic Search

Data Manager - Equipment - Basic Search is recommended for usage scenarios when specific objects whose identifiers are known need to be located within a very large number of equipment objects. A scenario of this kind can evolve when large numbers of equipment entities are generated for one-time use on the shop floor, followed by their immediate archival.

Menus and Toolbars

You can access all relevant functions either from the menu bar in the main Data Manager - Equipment - Basic Search window, from context-sensitive shortcut menus, or from toolbars provided for quick access to frequently used functions.

MENUS

Data Manager - Equipment - Basic Search provides a main menu bar (page 208) with all relevant functions as well as context-sensitive shortcut menus (page 213) for quick access to specific functions.

MAIN MENU BAR

The **main menu bar** offers the following menus and functions:

File

- New equipment class (CTRL+SHIFT+C)
  Creates a new equipment class, thus first opening the **New Equipment Class** dialog and afterwards a new tab in the lower tab bar of the Details window.
- New template equipment entity (CTRL+SHIFT+T)
  Creates a new template equipment entity, thus first opening the **New Template Equipment Entity** dialog and afterwards a new tab in the lower tab bar of the Details window.
- New equipment entity (CTRL+SHIFT+E)
  Creates a new equipment entity, thus first opening the **New Equipment Entity** dialog and afterwards a new tab in the lower tab bar of the Details window.
- New equipment graph (CTRL+SHIFT+G)
  Creates a new equipment graph, thus first opening the **New Equipment Graph** dialog and afterwards a new tab in the lower tab bar of the Details window.
- New property type (CTRL+SHIFT+P)
  Creates a new property type, thus first opening the **New Property Type** dialog and afterwards a new tab in the lower tab bar of the Details window.
- Import
  Starts the process of importing (page 110) an equipment data object with all referenced objects by selecting, checking, and importing a PharmaSuite import ZIP archive.
- Import tags/points
  Starts the process of importing (page 115) the tags and points of equipment entities by selecting, checking, and importing a PharmaSuite import XML file.
- Export <object identifier>
  Starts the process of exporting (page 103) an equipment object with all referenced objects by selecting a target location for the PharmaSuite export ZIP archive.
- Export tags/points of <object identifier>
  Starts the process of exporting (page 107) the tags and points of equipment entities by selecting a target location for the PharmaSuite export XML file.
- Verify tags/points of <object identifier>
  Starts the process of verifying (page 118) the tag paths and point names of equipment entities.

- Save <object identifier> (CTRL+S)
  Saves all changes made to the object that is currently active in the lower tab bar of the Details window. This action saves all changes made on the object's attributes and properties tabs.

- Save all (CTRL+SHIFT+S)
  Saves all changes made to all object that are currently active in the lower tab bar of the Details window. This action saves all changes made on the objects' properties tabs.

  TIP
  Please note that you can only save an object if none of its assigned objects have unsaved changes.

- Duplicate <equipment class identifier> without entities (CTRL+F12)
  Only available for equipment classes.
  Opens the **Duplicate Equipment Class** dialog to create a copy of the class. The new class opens as new tab in the lower tab bar of the Details window.
  The **Duplicate** function copies all attributes and properties of the original class except for its template entity and entity assignments.

- Duplicate <(template) equipment entity identifier> without classes (CTRL+F12)
  Only available for template equipment entities or equipment entities.

  - For template equipment entities
    Opens the **Duplicate Template Equipment Entity** dialog to create a copy of the template entity. The new template entity opens as new tab in the lower tab bar of the Details window.

  - For equipment entities
    Opens the **Duplicate Equipment Entity** dialog to create a copy of the entity. The new entity opens as new tab in the lower tab bar of the Details window.

  The **Duplicate** function copies all attributes and properties of the original including all of its assignments.

- Duplicate <(template) equipment entity identifier> with classes (CTRL+SHIFT+F12)
  Only available for template equipment entities or equipment entities.

  - For template equipment entities
    Opens the **Duplicate Template Equipment Entity** dialog to create a copy of the template entity. The new template entity opens as new tab in the lower tab bar of the Details window.

  - For equipment entities
    Opens the **Duplicate Equipment Entity** dialog to create a copy of the entity. The new entity opens as new tab in the lower tab bar of the Details window.

  The **Duplicate** function copies all attributes and properties of the original including all of its assignments.

  > TIP
  > Please note that tag names defined with automation properties are suffixed with "_COPY" when they are duplicated.

  Duplicate <equipment graph identifier> (CTRL+F12)
  Only available for equipment graphs.
  Opens the **Duplicate Equipment Graph** dialog to create a copy of the graph. The new graph opens as new tab in the lower tab bar of the Details window.

- Duplicate <property type identifier> (CTRL+F12)
  Only available for property types.
  Opens the **Duplicate Property Type** dialog to create a copy of the property type. The new property type opens as new tab in the lower tab bar of the Details window.

- Change status of <object identifier> (CTRL+SHIFT+H)
  Opens the **Change Status** dialog to perform a status change (page 130) on the equipment class, template entity, or entity that is currently active in the lower tab bar of the Details window.

- Force unbinding of <equipment entity identifier>
  Only available for equipment entities that are currently bound for processing in PharmaSuite for Production Execution.
  Opens an **Electronic Signature** dialog (page 101) to perform and sign for unbinding of the equipment entity that is currently active in the lower tab bar of the Details window.

  > TIP
  > This function should only be used as a last resort for scenarios such as cleaning up after processing has met with unresolvable issues in PharmaSuite for Production Execution.

- Update (template) entities of <class identifier>
  Only available for equipment classes that hold template entities or entities.
  Opens the **Update (Template) Entities** dialog to perform a mass change operation on all entities and template entities assigned to the class.

- Compile usage list of <class identifier>
  Only available for equipment classes.
  Opens the **Usage List** dialog to compile a list of recipes, workflows, and building blocks that reference the equipment class as requirement or in an expression.

- Close <object identifier> (CTRL+F4)
  Closes the object that is currently active in the lower tab bar of the Details window.
- Close all (CTRL+SHIFT+F4)
  Closes all objects that are currently open in the lower tab bar of the Details window.
- Restore <object identifier> (CTRL+R)
  Only available if there are unsaved changes.
  Restores the object that is currently active in the lower tab bar of the Details window to its last saved state, thus undoing all changes made in the meantime.
- Refresh <object identifier> (F5)
  Only available if there are no unsaved changes.
  Refreshes the object that is currently active in the lower tab bar of the Details window by retrieving its data again from the database. Thus, it updates all runtime data, such as runtime properties, graph statuses or grouping situations, which may have changed due to processing activities on the shop floor.
- Delete <object identifier>
  Deletes the object that is currently active in the lower tab bar of the Details window.
- Print barcode label for <object identifier> (CTRL+B)
  Only available for equipment entities.
  Sends the barcode of the equipment entity that is currently active in the Details window for printing to a connected printer.
- Exit (ALT+F4)
  Closes the application window.

View

- New search (CTRL+N)
  Opens a new Search window to define another set of search criteria and process its search results.
- Zoom in (CTRL+PLUS)
  Zooms in on the currently active search result, switching the display to the next larger view, either tiles or cards.
- Zoom out (CTRL+MINUS)
  Zooms out from the currently active search result, switching the display to the next smaller view, either tiles or mini tiles.
- Hide Archived objects (CTRL+SHIFT+A)
  Toggles the display of objects in the **Archived** status in the results panel.

- Card tooltips (CTRL+T)
  Toggles the display of the card view as tooltip when you hover over a tile or mini tile.

Window

- Equipment Manager - Smart Search
  Switches to the Equipment - Smart Search mode of Data Manager. Before switching, the system closes all open objects, prompting you to save if you have unsaved changes.
  A checkmark to the left of the selected menu option indicates which mode is currently active.
  Only available if you have the user rights necessary for accessing Data Manager - Equipment - Smart Search.
- Equipment Manager - Basic Search
  Switches to the Equipment - Basic Search mode of Data Manager. Before switching, the system closes all open objects, prompting you to save if you have unsaved changes.
  A checkmark to the left of the selected menu option indicates which mode is currently active.
  Only available if you have the user rights necessary for accessing Data Manager - Equipment - Basic Search.
- Work Center Manager
  Switches to the Work Center mode of Data Manager. Before switching, the system closes all open objects, prompting you to save if you have unsaved changes.
  A checkmark to the left of the selected menu option indicates which mode is currently active.
  Only available if you have the user rights necessary for accessing Data Manager - Work Center.
- Undo layout change
  Revokes the last layout change you have performed. You can undo up to 100 actions, thus you can step by step revoke the last 100 layout changes you have performed. Once you have performed a layout change, the menu function changes to a more precise description of the undo action, such as **Undo resizing** or **Undo dragging**.
- Redo layout change
  Redoes the last layout change you have revoked with the **Undo layout change** function. You can redo up to 100 actions, thus you can step by step redo the last 100 layout changes you have revoked. Once you have performed a layout change, the menu function changes to a more precise description of the undo action, such as **Redo resizing** or **Redo dragging**.

- Save user layout
  Saves the current window layout and overwrites the layout that was last saved by you on this computer.
- Load user layout
  Loads the last layout you have saved on this computer with the **Save user layout** function.
- Reset layout
  Resets the window layout to the system-defined default layout. This function does not affect the saved user layout, which can be restored with the **Load user layout** function.

Help

- Data Manager Help (ALT+F1)
  Opens a web browser to display the start page of the help system (page 7).
- About PharmaSuite
  Opens the **About PharmaSuite** dialog (page 8).

Shortcut Menus

The card and tile views make object type-specific **shortcut menus** available, which you can access by right-clicking a card or tile in the search results panel.

Equipment class

- Open <equipment class identifier> in Details window
  Opens the detail view of the equipment class as new tab in the Details window.
- Duplicate <equipment class identifier> without (template) entities
  Opens the **Duplicate Equipment Class** dialog to create a copy of the class. The new class opens as new tab in the lower tab bar of the Details window.
  The **Duplicate** function copies all attributes and properties of the original class except for its template entity and entity assignments.
- New equipment class
  Creates a new equipment class, thus first opening the **New Equipment Class** dialog and afterwards a new tab in the lower tab bar of the Details window.
- Change status of <equipment class identifier>
  Opens the **Change Status** dialog to perform a status change (page 130) on the equipment class.
- Update (template) entities of <class identifier>
  Only available for equipment classes that hold template entities or entities.
  Opens the **Update (Template) Entities** dialog to perform a mass change operation on all entities and template entities assigned to the class.

- Restore <equipment class identifier>
  Only available if there are unsaved changes.
  Restores the equipment class to its last saved state, thus undoing all changes made in the meantime.
- Refresh <equipment class identifier>
  Only available if there are no unsaved changes.
  Refreshes the equipment class by retrieving its data again from the database.
- Delete <equipment class identifier>
  Only available if the class is in an editable status, such as **Draft**.
  Deletes the equipment class.
- Export <equipment class identifier>
  Starts the process of exporting (page 103) an equipment class with all referenced template entities, entities, graphs, and property types by selecting a target location for the PharmaSuite export ZIP archive.
- Export tags/points of <equipment class identifier>
  Starts the process of exporting (page 107) the tags and points of the entities of an equipment class by selecting a target location for the PharmaSuite export XML file.
- Verify tags/points of <equipment class identifier>
  Starts the process of verifying (page 118) the tag paths and point names of the equipment entities of a class.
- Compile usage list of <class identifier>
  Opens the **Usage List** dialog to compile a list of recipes, workflows, and building blocks that reference the equipment class as requirement or in an expression.
- Assign <equipment class identifier> to <equipment entity identifier> entity
  Only available if the active tab of the Details window holds a template equipment entity or an equipment entity.
  Assigns the class to the template entity or entity that is displayed in the Details window.
- Create entity from <equipment class identifier> class
  Creates a new equipment entity on the basis of the data defined for the equipment class. The function passes all attributes and properties defined with the class on to the new entity. Moreover, the class is assigned to the entity as its style class.
- Create template entity from <equipment class identifier> class
  Creates a new template equipment entity on the basis of the data defined for the equipment class. The function passes all attributes and properties defined with the class on to the new template entity. Moreover, the class is assigned to the template entity as its style class.

Template equipment entity

- Open <template equipment entity identifier> in Details window
  Opens the detail view of the template equipment entity as new tab in the Details window.
- Duplicate <template equipment entity identifier> without classes
  Opens the **Duplicate Template Equipment Entity** dialog to create a copy of the template entity. The new template entity opens as new tab in the lower tab bar of the Details window.
  The **Duplicate** function copies all attributes and properties of the template entity except for its class assignments.
- Duplicate <template equipment entity identifier> with classes
  Opens the **Duplicate Template Equipment Entity** dialog to create a copy of the template entity. The new template entity opens as new tab in the lower tab bar of the Details window.
  The **Duplicate with classes** function copies all attributes and properties of the original including all of its assignments.
- New template equipment entity (CTRL+SHIFT+T)
  Creates a new template equipment entity, thus first opening the **New Template Equipment Entity** dialog and afterwards a new tab in the lower tab bar of the Details window.
- Change status of <template equipment entity identifier>
  Opens the **Change Status** dialog to perform a status change (page 130) on the template equipment entity.
- Restore <template equipment entity identifier>
  Only available if there are unsaved changes.
  Restores the template equipment entity to its last saved state, thus undoing all changes made in the meantime.
- Refresh <template equipment entity identifier>
  Only available if there are no unsaved changes.
  Refreshes the template equipment entity by retrieving its data again from the database.
- Delete <template equipment entity identifier>
  Only available if the template entity is in an editable status, such as **Draft**, and has not been used as template for generating equipment entities during execution.
  Deletes the template equipment entity.
- Export <template equipment entity identifier>
  Starts the process of exporting (page 103) a template equipment entity with all referenced graphs and property types by selecting a target location for the PharmaSuite export ZIP archive.

- Assign <template equipment entity identifier> to <equipment class identifier> class
  Only available if the active tab of the Details window holds an equipment class. Assigns the template entity to the equipment class that is displayed in the Details window.
- Create class from <template equipment entity identifier> template equipment entity
  Creates a new equipment class on the basis of the data defined for the template equipment entity. The function passes all attributes and properties defined with the template entity on to the new class. Thus, the data on the **Style** tab of the new class is not defined.
- Create entity from <template equipment entity identifier> template equipment entity
  Creates a new equipment entity on the basis of the data defined for the template equipment entity. The function passes all attributes, properties, and assignments defined with the template entity on to the new entity.

Equipment entity

- Open <equipment entity identifier> in Details window
  Opens the detail view of the equipment entity as new tab in the Details window.
- Duplicate <equipment entity identifier> without classes
  Opens the **Duplicate Equipment Entity** dialog to create a copy of the entity. The new entity opens as new tab in the lower tab bar of the Details window.
  The **Duplicate** function copies all attributes and properties of the original except for its class assignments.
- Duplicate <equipment entity identifier> with classes
  Opens the **Duplicate Equipment Entity** dialog to create a copy of the entity. The new entity opens as new tab in the lower tab bar of the Details window.
  The **Duplicate with classes** function copies all attributes and properties of the original including its assignments.
- New equipment entity
  Creates a new equipment entity, thus first opening the **New Equipment Entity** dialog and afterwards a new tab in the lower tab bar of the Details window.
- Change status of <equipment entity identifier>
  Opens the **Change Status** dialog to perform a status change (page 130) on the equipment entity.

- Force unbinding of <equipment entity identifier>
  Only available for equipment entities that are currently bound for processing in PharmaSuite for Production Execution.
  Opens an **Electronic Signature** dialog (page 101) to perform and sign for unbinding of the equipment entity that is currently active in the lower tab bar of the Details window.

  > TIP
  > This function should only be used as a last resort for scenarios such as cleaning up after processing has met with unresolvable issues in PharmaSuite for Production Execution.

- Restore <equipment entity identifier>
  Only available if there are unsaved changes.
  Restores the equipment entity to its last saved state, thus undoing all changes made in the meantime.

- Refresh <equipment entity identifier>
  Only available if there are no unsaved changes.
  Refreshes the equipment entity by retrieving its data again from the database. Thus, it updates all runtime data, such as runtime properties, graph statuses or grouping situations, which may have changed due to processing activities on the shop floor.

- Delete <equipment entity identifier>
  Only available if the entity is in an editable status, such as **Draft**, and has not been used for processing yet.
  Deletes the equipment entity.

- Export <equipment entity identifier>
  Starts the process of exporting (page 103) an equipment entity with all referenced graphs and property types by selecting a target location for the PharmaSuite export ZIP archive.

- Export tags/points of <equipment entity identifier>
  Starts the process of exporting (page 107) the tags and points of an equipment entity by selecting a target location for the PharmaSuite export XML file.

- Verify tags/points of <equipment entity identifier>
  Starts the process of verifying (page 118) the tag paths and point names of an equipment entity.

- Assign <equipment entity identifier> to <equipment class identifier> class
  Only available if the active tab of the Details window holds an equipment class.
  Assigns the entity to the equipment class that is displayed in the Details window.

- Create class from <equipment entity identifier> equipment entity
Creates a new equipment class on the basis of the data defined for the equipment entity. The function passes all attributes and properties defined with the entity on to the new class. Thus, the data on the **Style** tab of the new class is not defined.

Equipment graph

- Open <equipment graph identifier> in Details window
Opens the detail view of the equipment graph as new tab in the Details window.
- Duplicate <equipment graph identifier>
Opens the **Duplicate Equipment Graph** dialog to create a copy of the graph. The new graph opens as new tab in the lower tab bar of the Details window.
- New equipment graph
Creates a new equipment graph, thus first opening the **New Equipment Graph** dialog and afterwards a new tab in the lower tab bar of the Details window.
- Change status of <equipment graph identifier>
Opens the **Change Status** dialog to perform a status change (page 130) on the equipment graph.
- Restore <equipment graph identifier>
Only available if there are unsaved changes.
Restores the equipment graph to its last saved state, thus undoing all changes made in the meantime.
- Refresh <equipment graph identifier>
Only available if there are no unsaved changes.
Refreshes the equipment graph by retrieving its data again from the database.
- Delete <equipment graph identifier>
Deletes the equipment graph.
Only available if the graph is not assigned to another object, such as a class or an entity.
- Export <equipment graph identifier>
Starts the process of exporting (page 103) an equipment graph with all referenced property types by selecting a target location for the PharmaSuite export ZIP archive.
- Assign <equipment graph identifier> to <object type identifier> <object type>
Only available if the active tab of the Details window holds an equipment class, template entity, or entity and the graph has not been assigned already to the class, template entity, or entity.
Assigns the graph to the equipment class, template entity, or entity that is displayed in the Details window.

Property type

- Open <property type identifier> in Details window
  Opens the detail view of the property type as new tab in the Details window.
- Duplicate <property type identifier>
  Opens the **Duplicate Property Type** dialog to create a copy of the property type. The new property type opens as new tab in the lower tab bar of the Details window.
- New property type
  Creates a new property type, thus first opening the **New Property Type** dialog and afterwards a new tab in the lower tab bar of the Details window.
- Restore <property type identifier>
  Only available if there are unsaved changes.
  Restores the property type to its last saved state, thus undoing all changes made in the meantime.
- Refresh <property type identifier>
  Only available if there are no unsaved changes.
  Refreshes the property type by retrieving its data again from the database.
- Delete <property type identifier>
  Only available if the property type is not assigned to another object, such as a class, an entity, or a graph.
  Deletes the property type.
- Export <property type identifier>
  Starts the process of exporting (page 103) a property type by selecting a target location for the PharmaSuite export ZIP archive.
- Assign <property type identifier> to <object type identifier> <object type>
  Only available if the active tab of the Details window holds a suitable tab of an equipment class, template entity, entity, or graph and the property type has not been assigned already to the class, template entity, entity, or graph.
  Assigns the property type to the property tab (**Specification**, **Process**, **Engineering**) of the equipment class, template entity, or entity or to the **Property Type** tab of the equipment graph that is displayed in the Details window.

TOOLBARS

Data Manager provides several toolbars for context-sensitive quick access to specific functions.

Main Toolbars

The **main toolbars** offer shortcuts to the more frequently required menu functions:

- File toolbar with
  - New equipment class (page 71)
  - New template equipment entity (page 72)
  - New equipment entity (page 71)
  - New equipment graph (page 71)
  - New property type (page 72)
  - Refresh (page 72)
  - Save (page 72)
  - Close (page 71)
  - Delete (page 71)
  - Change status (page 71)
  - Print barcode label (page 72)
- View toolbar with
  - Zoom in (page 73)
  - Zoom out (page 73)
- Help toolbar with
  - Help (page 71)

Property Tab Toolbar

The **property tab toolbar** provides the following functions for sorting and managing the property types assigned to an equipment class, entity, or graph:

- Move down (page 74)
- Move up (page 74)
- Display on card (page 73)
  Only available for equipment classes and graphs.
- Remove (page 75)

Detail Data

From the upper tab bar in the Details window, you can access the attributes, properties, assignments, and usage data of the following equipment-related objects:

- Equipment classes (page 221)
- Template equipment entities (page 228)
- Equipment entities (page 236)
- Equipment graphs (page 251)
- Property types (page 262)

Class

The Details window of a class contains the following data tabs:

- Basic (page 221)
- Specification (page 223)
- Process (page 223)
- Engineering (page 224)
- Graph (page 224)
- Entity (page 225)
- Style (page 225)
- Change History (page 225)
- Status History (page 227)

Basic

The **Basic** tab lists the Asset Attributes of the equipment class.

Click an attribute value to edit it. The system displays a text cursor or a button to access a suitable editor (page 10) for the required value type. Any asset attribute values you define for an equipment class are copied to an equipment entity created from the class. The attribute values copied to the equipment entity can, however, be changed on the equipment entity without affecting the values defined on the class.

TIP

Please note that once you have created an equipment class, you can no longer modify its **Identifier**.

DEFAULT CLASSES

PharmaSuite provides the following equipment classes with any default installation of the system. They are intended as templates to create entities that contain all required graphs and property types:

- Container_RS_1
  The default class for containers includes the following default objects:
  - Graph:
    ContainerCleaning_RS_1
  - Property types:
    - Container Clean Shelflife (RS)
    - Container Sublot (RS)
    - Container Tare (RS)
    - Container Type (RS)
- Room_RS_1
  The default class for rooms includes the following default objects:
  - Graph
    RoomCleaning_RS_1
  - Property types:
    - General Clean Shelflife (RS)
    - Room Cleaning Rules (RS)
    - Room Type (RS)
    - Work Center Assignment (RS)
- Scale_RS_1
  The default class for scales includes the following default objects:
  - Graphs:
    - ScaleCalibration_RS_1
    - ScaleTest_RS_1
  - Property types:
    - Scale Calibration Shelflife (RS)
    - Scale Configuration (RS)
    - Scale Load (RS)
    - Scale Ranges (RS)
    - Scale Test and Calibration (RS)

- Scale Test Shelflife (RS)
- Scale Type (RS)
- Work Center Assignment (RS)

The **Level** attribute refers to the physical model of the S88 standard and provides the following options:

- Control module
- Equipment module
- Unit
- Process cell

The **Disposable** attribute allows you to choose (**Yes**, **No**, **N/A**) if the class groups (template) equipment entities that are intended for single use.

The **Logbook enabled** attribute allows you to choose (**Yes**, **No**, **N/A**) if the class groups equipment entities whose life cycle needs to be tracked in the logbook.

Specification

The **Specification** tab lists the property types assigned as specification properties to the equipment class.

- Click a property value to edit it. The system displays a text cursor or a button to access a suitable editor (page 10) for the required value type.
- To add a property to the list, drag and drop (page 100) its property type on the specification tab.
- To change a property's position in the list, click the **Move up** or **Move down** buttons (page 74).
- To display a property on the card view of the equipment class, click the **Display on card** button (page 73).
- To remove a property from the list, click the **Remove property** button (page 75).

Process

The **Process** tab lists the property types assigned as process properties to the equipment class.

- Click a property value to edit it. The system displays a text cursor or a button to access a suitable editor (page 10) for the required value type.
- To add a property to the list, drag and drop (page 100) its property type on the process tab.

- To change a property's position in the list, click the **Move up** or **Move down** buttons (page 74).
- To display a property on the card view of the equipment class, click the **Display on card** button (page 73).
- To remove a property from the list, click the **Remove property** button (page 75).

ENGINEERING

The **Engineering** tab lists the property types assigned as engineering properties to the equipment class.

- Click a property value to edit it. The system displays a text cursor or a button to access a suitable editor (page 10) for the required value type.
- To add a property to the list, drag and drop (page 100) its property type on the engineering tab.
- To change a property's position in the list, click the **Move up** or **Move down** buttons (page 74).
- To display a property on the card view of the equipment class, click the **Display on card** button (page 73).
- To remove a property from the list, click the **Remove property** button (page 75).

GRAPH

The **Graph** tab lists the equipment graphs assigned to the equipment class.

- To add a graph to the list, drag and drop (page 100) its card or tile on the Details window of the class to which you wish to assign it.

> TIP
> Please note that you can only assign graphs that do not have unsaved changes and contain at least one status.

- To change a graph's position in the list, click the **Move up** or **Move down** buttons (page 74).
- To display a graph with its short description on the card view of the equipment class, click the **Display on card** button (page 73).
- To remove a graph from the list, click the **Remove graph** button (page 75).

ENTITY

The **Entity** tab lists the equipment entities and template equipment entities assigned to the equipment class. The checkbox in the **Style class** column indicates which of the assigned (template) entities take their styling (color, icon, properties displayed on card view) from the class. The checkbox in the **Template** column indicates which of the assigned objects are plain entities and which of them are template entities.

- To add a (template) entity to the list, drag and drop (page 100) its card or tile on the Details window of the class to which you wish to assign it.

  TIP
  Please note that the system only checks if the properties of the assigned (template) entity correspond to those of your class, but not on which tabs (**Specification**, **Process**, **Engineering**) the properties are located. Thus, the system does not rearrange the (template) entity's properties to match the properties' positioning on the class.

- To remove a (template) entity from the list, select or multi-select it and click the **Remove equipment entity** button (page 75) or press the DEL key.
  Removing a (template) entity from the list of entities also removes the class from the list of classes shown on the **Class** tab of the (template) entity's detail view.

STYLE

The **Style** tab provides tools for defining the visual appearance of the equipment classes' tiles and cards:

- Background color
- Font color
- Icon

TIP

Please note that (template) equipment entities assigned to a class can have the class set as their style class and thus inherit its visual appearance.

CHANGE HISTORY

The **Change History** tab provides a list of all changes that have been performed on the equipment class, which also includes indirect assignment changes that occur when an entity's assignment change also involves the class.
For each change event that affected the class, the table displays one row.

To filter the change event entries by date, select the date range you wish to consider. The system only displays the entries that have been registered between the two dates you have selected.
The date range starts at 12:00:00 AM on the date you have selected and ends at 23:59:59 PM on the date you have selected.
When you open the change history, the date range is preset to start with the date when the last change event occurred and to end with the current date. If the system has no change events registered for the object, the date range is preset to the current date.
To refresh the list of change events, click the **Refresh** button (page 75). All change events involving the class that have occurred since you opened the change history are then included.

The following table columns are available and are filled if relevant to the respective change event:

- Timestamp
  indicates the date and time when the change event occurred.
- Action
  indicates the change event that triggered the entry in the change history.
- Affected object
  indicates the object that is affected by the change event. So, when you update the basic data of your class, the change history shows the identifier of the attribute you have modified and when you assign a template entity, entity, graph, or property type to your class, the change history shows the identifier of the assigned object.
- Old content/New content
  for data change events on your class, displays the changed values.
- Signature
  displays two items per signature (page 101):
  - User and login names of the user who performed the signature.
  - Timestamp when the signature was recorded.
- Comment
  displays the comment recorded for each signature. For double signatures, each comment is prefixed with **Comment 1** or **Comment 2** to indicate with which signature the comment was recorded.

  > **TIP**
  > Signatures and their comments are only recorded for mass change operations.

- Logged-in user
  login name of the user under whose login the change event occurred.

STATUS HISTORY

The **Status History** tab provides a list of all status changes that have been performed on the equipment class.
For each status change that affected the class, the table displays one row.

To filter the status change entries by date, select the date range you wish to consider. The system only displays the entries that have been registered between the two dates you have selected.
The date range starts at 12:00:00 AM on the date you have selected and ends at 23:59:59 PM on the date you have selected.
When you open the status history, the date range is preset to start with the date when the last status change occurred and to end with the current date. If the system has no status changes registered for the object, the date range is preset to the current date.
To refresh the list of status changes, click the **Refresh** button (page 75). All status changes involving the class that have occurred since you opened the status history are then included.

The following table columns are available and are filled if relevant to the respective status change:

- Timestamp
  indicates the date and time when the status change occurred.
- Action
  indicates the status change that triggered the entry in the status history.
- To status
  indicates the status to which the class was moved.
- Signature
  displays two items per signature (page 101):
  - User and login names of the user who performed the signature.
  - Timestamp when the signature was recorded.
- Comment
  displays the comment recorded for each signature. For double signatures, each comment is prefixed with **Comment 1** or **Comment 2** to indicate with which signature the comment was recorded.

Template Entity

The Details window of an entity contains the following data tabs:

- Basic (page 228)
- Specification (page 231)
- Process (page 232)
- Engineering (page 232)
- Graph (page 233)
- Class (page 233)
- Material (page 233)
- Change History (page 234)
- Status History (page 235)

Basic

The **Basic** tab lists the Asset, Automation, Historian, and Entity Barcode attributes of the template equipment entity.

Click an attribute value to edit it. The system displays a text cursor or a button to access a suitable editor (page 10) for the required value type.
Asset, Automation, and Historian attribute values defined for a template entity are copied to all equipment entities created from the template in Data Manager or generated from the template on the shop floor.
Asset attribute values you define for a template equipment entity are copied to an equipment class created from the entity.
The attribute values copied to the new object can, however, be changed on the new object without affecting the values defined on the template entity.

> **TIP**
>
> Please note that once you have created a template equipment entity, you can no longer modify its **Identifier**.

The **Level** attribute refers to the physical model of the S88 standard and provides the following options:

- Control module
- Equipment module
- Unit
- Process cell

The **Disposable** attribute allows you to choose (**Yes**, **No**) if the entities generated from the template are intended for single use.

The **Logbook enabled** attribute allows you to choose (**Yes**, **No**) if the life cycles of all generated entities need to be tracked in the logbook.

The **Label layout (default)** attribute shows which label layout is pre-configured by the system and defines the layout that will be used for printing entity barcode labels when the template entity is used for generating entities on the shop floor.
With the **Label layout** attribute, you can override the default setting and define a different label layout to be used. For selecting a suitable label layout, the system provides the **Label Layout Selection** editor (page 58).
If you leave the box blank, the system uses the pre-configured default label design displayed in the **Label layout (default)** attribute.

When a new entity is generated from the template entity during execution, the **Entity Barcode** attributes determine what the identifier and barcode of the new entity will look like.
The identifier of a generated entity consists of a prefix, which is a configurable sequence of characters and a count, whose number of digits is also configurable.

- The **Barcode prefix character (default)** is read-only and shows which character (if any) is pre-configured as barcode prefix by the system.
- With the **Barcode prefix source** attribute, you can define if the barcode will be prefixed with the default character (**From configuration key**) or with a character specifically defined with the template (**Template-specific**).
- The **Character** attribute is only available when the **Barcode prefix source** is set to **Template-specific**. It must be no longer than one character but can also be left blank.
- The **Identifier prefix (default)** is read-only and shows the sequence of characters that is pre-configured by the system to precede the count of a generated entity.
- With the **Identifier prefix** you can override the system default with another sequence of characters. It can have a length of up to 30 characters but can also remain blank.
- The **Sequencer length (default)** is read-only and shows the pre-configured number of digits to be used for the count of generated entities.
- With the **Sequencer length** you can override the system default with another number of digits for the count. The system uses leading zeros to fill up the digits if the actual count would require fewer digits.

Examples:

Barcode prefix source	Character	Identifier prefix	Sequencer length	Entity identifier Entity barcode
Template-specific	&	FL	5	FL00001 &FL00001
Template-specific		F-Liner	10	F-Liner0000000001 F-Liner0000000001
Template-specific	§		15	000000000000001 §000000000000001
Template-specific	$	FL-	8	FL-00000001 $FL-00000001

With the Automation attributes, you can establish the connection to the automation layer, by way of the **Automation Integration server name** and the **Live Data Area path**. The information is required if you wish to use automation phases in your workflows or master recipes. The system shows the pre-configured default paths, which are used unless you define another path, which then overrides the default.
The **Live Data Area path** specifies the path the system uses to access the tags on your generated equipment entity within the Factory Talk Directory. Together with the specific tag names you can configure when you edit the automation properties (page 10) of the entity, it forms the full path to a specific tag. The path consists of a protocol prefix (**RNA://$Global**), an application name, and one or more area names.
So the path to an equipment entity, located in a **Coating** sub-area of a **Building03** area within a company's **MES** application would be
RNA://$Global/MES/Building03/Coating.

With the Historian attributes, you can access the data archived in a Historian application. Unless you can use the pre-defined defaults, you need to specify the **Provider** for interfacing to the **Historian server**, the **Historian access server**, and also the name of the **Automation Integration server** through which the data access is channeled. The information is required if your master recipe or workflow holds phases for retrieving historical data.

SPECIFICATION

The **Specification** tab lists the property types assigned as specification properties to the template equipment entity.

TIP

Automation properties indicate if a tag path has been specified for them.
(-) means that they have no tag path,
(+) means that at least one tag path has been defined.
Please note, however, that the indicator only refers to the existence of a tag path and not to its validity.

Template entities to be used for generating material-related equipment entities during production require specific object assignments:

Object type	Usage	Data type Content	Purpose
Property type	Specification	EquipmentType Hybrid (RS)	---
Property type	Runtime	String	Current Sublot (RS)
Property type	Runtime	String	Base Sublot (RS)

When a phase building block generates an entity during identification of a sublot, the system determines the suitable template entity on account of the template entity assigned to the sublot's material.

When the identified sublot is split, but continues to exist, the system fills the new entity's **Current Sublot (RS)** runtime property with the identifier of the sublot that is generated along with the entity. The **Base Sublot (RS)** runtime property is filled with the identifier of the sublot that has been split in order to generate the sublot of the **Current Sublot (RS)** runtime property.

When the identified sublot is consumed immediately and thus ceases to exist, its identifier is shown in the **Base Sublot (RS)** runtime property of the generated entity. The **Current Sublot (RS)** runtime property of the entity remains blank in this case.

- Click a property value to edit it. The system displays a text cursor or a button to access a suitable editor (page 10) for the required value type.
- To add a property to the list, drag and drop (page 100) its property type on the specification tab.
- To change a property's position in the list, click the **Move up** or **Move down** buttons (page 74).
- To remove a property from the list, click the **Remove property** button (page 75).

Process

The **Process** tab lists the property types assigned as process properties to the template equipment entity.

> **TIP**
>
> Automation properties indicate if a tag path has been specified for them.
> (-) means that they have no tag path,
> (+) means that at least one tag path has been defined.
> Please note, however, that the indicator only refers to the existence of a tag path and not to its validity.

- Click a property value to edit it. The system displays a text cursor or a button to access a suitable editor (page 10) for the required value type.
- To add a property to the list, drag and drop (page 100) its property type on the process tab.
- To change a property's position in the list, click the **Move up** or **Move down** buttons (page 74).
- To remove a property from the list, click the **Remove property** button (page 75).

Engineering

The **Engineering** tab lists the property types assigned as engineering properties to the template equipment entity.

> **TIP**
>
> Automation properties indicate if a tag path has been specified for them.
> (-) means that they have no tag path,
> (+) means that at least one tag path has been defined.
> Please note, however, that the indicator only refers to the existence of a tag path and not to its validity.

- Click a property value to edit it. The system displays a text cursor or a button to access a suitable editor (page 10) for the required value type.
- To add a property to the list, drag and drop (page 100) its property type on the engineering tab.
- To change a property's position in the list, click the **Move up** or **Move down** buttons (page 74).
- To remove a property from the list, click the **Remove property** button (page 75).

Graph

The **Graph** tab lists the equipment graphs assigned to the template equipment entity. Each graph shows with its initial status and expiry date (if set), both of which cannot be changed. The statuses controlled by the graph do not apply to the template entity itself, but to all entities generated from it.

- To add a graph to the list, drag and drop (page 100) its card or tile on the Details window of the template entity to which you wish to assign it.

> TIP
> Please note that you can only assign graphs that do not have unsaved changes and contain at least one status.

- To change a graph's position in the list, click the **Move up** or **Move down** buttons (page 74).
- To remove a graph from the list, click the **Remove graph** button (page 75).

Class

The **Class** tab lists the equipment classes assigned to the template equipment entity. A template equipment entity can belong to several equipment classes, but takes its styling (color, icon, properties displayed on card view) from the class set as its **Style class**.

- To add a class to the list, drag and drop (page 100) its card or tile on the Details window of the entity to which you wish to assign it.

> TIP
> Please note that the system only checks if the properties of the assigned class correspond to those of your template entity, but not on which tabs (**Specification**, **Process**, **Engineering**) the properties are located. Thus, the system does not rearrange the template entity's properties to match the properties' positioning on the class.

- To define the styling of the template entity by setting an assigned class as style class, select the **Style class** checkbox of the class whose styling the template entity is to adopt.
- To remove a class from the list, select or multi-select it and click the **Remove equipment class** button (page 75) or press the DEL key.
  Removing a class from the list of classes also removes the template entity from the list of entities shown on the **Entity** tab of the class' detail view.

Material

The **Material** tab lists the materials assigned to the template equipment entity. The assignments between materials and template entities are maintained in PharmaSuite for Production Management, where only those templates entities are available for assignment that have a specification property of the **Hybrid (RS)** equipment type.

CHANGE HISTORY

The **Change History** tab provides a list of all changes that have been performed on the template equipment entity, which also includes indirect assignment changes that occur when a class' assignment change also involves the template entity.
For each change event that affected the template entity, the table displays one row.

To filter the change event entries by date, select the date range you wish to consider. The system only displays the entries that have been registered between the two dates you have selected.
The date range starts at 12:00:00 AM on the date you have selected and ends at 23:59:59 PM on the date you have selected.
When you open the change history, the date range is preset to start with the date when the last change event occurred and to end with the current date. If the system has no change events registered for the object, the date range is preset to the current date.
To refresh the list of change events, click the **Refresh** button (page 75). All change events involving the template entity that have occurred since you opened the change history are then included.

The following table columns are available and are filled if relevant to the respective change event:

- Timestamp
  indicates the date and time when the change event occurred.
- Action
  indicates the change event that triggered the entry in the change history.
- Affected object
  indicates the object that is affected by the change event. So, when you update the basic data of your template entity, the change history shows the identifier of the attribute you have modified and when you assign a class, graph, or property type to your template entity, the change history shows the identifier of the assigned class, graph, or property type.
- Old content/New content
  for data change events on your template entity, displays the changed values.
- Information
  indicates additional details on the change, such as if it was performed in the context of a mass change or a tag or point import.
- Logged-in user
  login name of the user under whose login the change event occurred.

STATUS HISTORY

The **Status History** tab provides a list of all status changes that have been performed on the template equipment entity.
For each status change that affected the template entity, the table displays one row.

To filter the status change entries by date, select the date range you wish to consider. The system only displays the entries that have been registered between the two dates you have selected.
The date range starts at 12:00:00 AM on the date you have selected and ends at 23:59:59 PM on the date you have selected.
When you open the status history, the date range is preset to start with the date when the last status change occurred and to end with the current date. If the system has no status changes registered for the object, the date range is preset to the current date.
To refresh the list of status changes, click the **Refresh** button (page 75). All status changes involving the template entity that have occurred since you opened the status history are then included.

The following table columns are available and are filled if relevant to the respective status change:

- Timestamp
  indicates the date and time when the status change occurred.
- Action
  indicates the status change that triggered the entry in the status history.
- To status
  indicates the status to which the template entity was moved.
- Signature
  displays two items per signature (page 101):
  - User and login names of the user who performed the signature.
  - Timestamp when the signature was recorded.
- Comment
  displays the comment recorded for each signature. For double signatures, each comment is prefixed with **Comment 1** or **Comment 2** to indicate with which signature the comment was recorded.

ENTITY

The Details window of an entity contains the following data tabs:

- Basic (page 236)
- Specification (page 239)
- Process (page 242)
- Engineering (page 243)
- Graph (page 243)
- Class (page 244)
- Grouping (page 245)
- Context (page 246)
- Logbook (page 247)
- Migrated History (page 249) (only for migrated equipment entities)
- Change History (page 249)
- Status History (page 250)

BASIC

The **Basic** tab lists the Asset, Automation, and Historian attributes of the equipment entity.

Click an attribute value to edit it. The system displays a text cursor or a button to access a suitable editor (page 10) for the required value type. Any asset attribute values you define for an equipment entity are copied to an equipment class created from the entity. The attribute values copied to the equipment class can, however, be changed on the equipment class without affecting the values defined on the entity.

When an equipment entity is used for processing in PharmaSuite for Production Execution it is bound to the process. Under very rare circumstances, in may be necessary to unbind an entity form the process in order to resolve data issues that cannot be solved in any other manner. For these cases, the system provides the **Force unbinding** function in the File menu (page 208) or the entities shortcut menu (page 216).

TIPS

Please note that once you have created an equipment entity, you can no longer modify its **Identifier**.

For scale equipment entities, which hold a property of the **Scale (RS)** EquipmentType, the **Short description** is displayed during execution as button text on the scale buttons in the **Select scale** phases of Weighing or Dispense operations. If there is no short description defined, the system displays the scale's **Identifier** instead.
Thus, the maximum number of characters for the **Short description** should not exceed 12 and should include a blank to trigger a line break when displayed on the button. One line on the button can hold approximately 6 characters. If the text exceeds the button width, it is truncated.

The **Level** attribute refers to the physical model of the S88 standard and provides the following options:

- Control module
- Equipment module
- Unit
- Process cell

The **Disposable** attribute allows you to choose (**Yes**, **No**) if the entity is intended for single use.

The **Logbook enabled** attribute allows you to choose (**Yes**, **No**) if the entity's life cycle needs to be tracked in the logbook.

If the entity was generated on the shop floor, the **Template used** attribute displays the identifier of the template entity that was used for its generation.

The **Label layout (default)** attribute shows which label layout is pre-configured by the system and defines the layout that will be used for printing the barcode labels of the entity.
With the **Label layout** attribute, you can override the default setting and define a different label layout to be used. For selecting a suitable label layout, the system provides the **Label Layout Selection** editor (page 58).
If you leave the box blank, the system uses the pre-configured default label design displayed in the **Label layout (default)** attribute.

At regular intervals, the system checks all equipment entities with expirable statuses to determine if statuses have expired. The **Automatic status change after expiry** attribute indicates if the system-performed status transition is enabled for the equipment entity. When an equipment entity is bound for processing or locked for editing, the system does not perform any status transition, even if its status has expired. The attribute retains its **Enabled** setting, and the entity is considered again during the next run of the expiry update.
If the system encounters an error while trying to update an expired status, however, it sets the **Automatic status change after expiry** attribute to **Disabled after error**. From then on, the entity is ignored during subsequent runs of the **Automatic status change after expiry** function. Once the error that prevents a successful status transition has been resolved, you need set the attribute to **Enabled**. The system will then consider the entity again for checking expiry dates and automatically performing status transitions.

When you create a new entity, the system pre-fills its **Barcode** attribute with a string of characters that consists of a system-configured equipment entity barcode prefix and the entity's identifier you have specified as first step of the creation process. The system-configured prefix may be empty so that the barcode and the entity’s identifier are identical. For barcodes of entities that are generated from a template entity during processing, however, the system calculates the **Barcode** attribute from the definitions made in the **Entity Barcode Attributes** section on the template entity's **Basic** tab (page 228).
To print the barcode label of an entity, proceed as follows:

1. Make sure the **Barcode** attribute cell contains the required unique string of characters.

   > **TIP**
   > Do not type leading of trailing blanks before or after the barcode string, since this may interfere with the system-internal formatting conversion and render the barcode unreadable for scanners.

2. Save the equipment entity.
   The system enables the **Print barcode label** button on the **File** toolbar as well as the respective function in the **File** menu.

3. Click the **Print barcode label** button or select the respective menu function to print the barcode on a connected printer, typically the default printer defined in your Windows operating system.

With the Automation attributes, you can establish the connection to the automation layer, by way of the **Automation Integration server name** and the **Live Data Area path**. The information is required if you wish to use automation phases in your workflows or master recipes. The system shows the pre-configured default paths, which are used unless you define another path, which then overrides the default.

The **Live Data Area path** specifies the path the system uses to access the tags on your equipment entity within the Factory Talk Directory. Together with the specific tag names you can configure when you edit the automation properties (page 10) of the entity, it forms the full path to a specific tag. The path consists of a protocol prefix (**RNA://$Global**), an application name, and one or more area names.

So the path to an equipment entity, located in a **Coating** sub-area of a **Building03** area within a company's **MES** application would be **RNA://$Global/MES/Building03/Coating**.

With the Historian attributes, you can access the data archived in a Historian application. Unless you can use the pre-defined defaults, you need to specify the **Provider** for interfacing to the **Historian server**, the **Historian access server**, and also the name of the **Automation Integration server** through which the data access is channeled. The information is required if your master recipe or workflow holds phases for retrieving historical data.

> **TIP**
>
> Please note that changes made to Automation or Historian attributes of an equipment entity that is already in use on the shop floor are not automatically pushed to the Production Execution station that runs the process with the affected equipment entity. PharmaSuite for Production Execution needs to be restarted for the changes to become effective.

SPECIFICATION

The **Specification** tab lists the property types assigned as specification properties to the equipment entity.

> **TIP**
>
> Automation properties indicate if a tag path has been specified for them.
> (-) means that they have no tag path,
> (+) means that at least one tag path has been defined.
> Please note, however, that the indicator only refers to the existence of a tag path and not to its validity.

Some entities require specific object assignments:

Container

Object type	Usage	Data type Content	Purpose
Property type, such as Container Type (RS)	Specification	EquipmentType Container (RS)	---
Property type, such as Container Tare (RS)	Runtime	MeasuredValue	Current Tare (RS)
Property type, such as Container Sublot (RS)	Runtime	String, used in container cleaning graph	Current Sublot (RS)
Property type, such as Container Clean Shelflife (RS)	Specification	Duration, used in container cleaning graph	---
Equipment graph, such as ContainerCleaning_RS_1	---	---	Container Cleaning (RS)

TIP

To define a reference tare for a container that can be accessed when taring the container during Dispense or Output Weighing, create a MeasuredValue runtime property type of the Reference Tare (RS) purpose.

Room

Object type	Usage	Data type Content	Purpose
Property type, such as Room Type (RS)	Specification	EquipmentType Room (RS)	---
Property type, such as Room Cleaning Rules (RS)	Runtime	RoomCleaningRules <room-specific>	---
Property type, such as Work Center Assignment (RS)	Specification	WorkCenterAssignment <room-specific>	---
Property type, such as General Clean Shelflife (RS)	Specification	Duration, used in room cleaning graph	---
Equipment graph, such as RoomCleaning_RS_1	---	---	Room Cleaning (RS)

SCALE

Object type	Usage	Data type Content	Purpose
Property type, such as Scale Type (RS)	Specification	EquipmentType Scale (RS)	---
Property type, such as Scale Ranges (RS)	Specification	Ranges <scale-specific>	ScaleRanges (RS)
Property type, such as Scale Configuration (RS)	Specification	ScaleConfiguration <scale-specific>	---
Property type, such as Scale Test and Calibration (RS)	Specification	ScaleTestAndCalibration <scale-specific>	---
Property type, such as Work Center Assignment (RS)	Specification	WorkCenterAssignment <scale-specific>	---
Property type, such as Scale Test Shelflife (RS)	Specification	Duration, used in scale test graph	---
Property type, such as Scale Calibration Shelflife (RS)	Specification	Duration, used in scale calibration graph	---
Property type, such as Scale Load (RS)	Runtime	String, required for handling of loaded scales	Current Load (RS)
Equipment graph, such as ScaleTest_RS_1	---	---	Scale Test (RS)
Equipment graph, such as ScaleCalibration_RS_1	---	---	Scale Calibration (RS)

- Click a property value to edit it. The system displays a text cursor or a button to access a suitable editor (page 10) for the required value type.

  TIP
  Runtime properties can change their values due to activities on the shop floor when the entity is in a read-only status, such as **Verification** or **Approved**. In order to resolve processing exceptions by administrative means, it is possible to update runtime data in these statuses in Data Manager as well. Before you change any runtime data, make sure to refresh the entity (page 72) so that it reflects the current situation on the shop floor.

- To add a property to the list, drag and drop (page 100) its property type on the specification tab.
- To change a property's position in the list, click the **Move up** or **Move down** buttons (page 74).
- To remove a property from the list, click the **Remove property** button (page 75).

PROCESS

The **Process** tab lists the property types assigned as process properties to the equipment entity.

TIP

Automation properties indicate if a tag path has been specified for them.
(-) means that they have no tag path,
(+) means that at least one tag path has been defined.
Please note, however, that the indicator only refers to the existence of a tag path and not to its validity.

- Click a property value to edit it. The system displays a text cursor or a button to access a suitable editor (page 10) for the required value type.

  TIP
  Runtime properties can change their values due to activities on the shop floor when the entity is in a read-only status, such as **Verification** or **Approved**. In order to resolve processing exceptions by administrative means, it is possible to update runtime data in these statuses in Data Manager as well. Before you change any runtime data, make sure to refresh the entity (page 72) so that it reflects the current situation on the shop floor.

- To add a property to the list, drag and drop (page 100) its property type on the process tab.
- To change a property's position in the list, click the **Move up** or **Move down** buttons (page 74).
- To remove a property from the list, click the **Remove property** button (page 75).

Engineering

The **Engineering** tab lists the property types assigned as engineering properties to the equipment entity.

> **TIP**
>
> Automation properties indicate if a tag path has been specified for them.
> (-) means that they have no tag path,
> (+) means that at least one tag path has been defined.
> Please note, however, that the indicator only refers to the existence of a tag path and not to its validity.

- Click a property value to edit it. The system displays a text cursor or a button to access a suitable editor (page 10) for the required value type.

  > **TIP**
  > Runtime properties can change their values due to activities on the shop floor when the entity is in a read-only status, such as **Verification** or **Approved**. In order to resolve processing exceptions by administrative means, it is possible to update runtime data in these statuses in Data Manager as well. Before you change any runtime data, make sure to refresh the entity (page 72) so that it reflects the current situation on the shop floor.

- To add a property to the list, drag and drop (page 100) its property type on the engineering tab.
- To change a property's position in the list, click the **Move up** or **Move down** buttons (page 74).
- To remove a property from the list, click the **Remove property** button (page 75).

Graph

The **Graph** tab lists the equipment graphs assigned to the equipment entity. Each graph shows with the current status of the entity. If the graph status can expire (page 253), the system displays its expiry date, if set. If the status has already expired, the system indicates this by marking the displayed (expired) status in strikethrough formatting.

- To add a graph to the list, drag and drop (page 100) its card or tile on the Details window of the entity to which you wish to assign it.

  > **TIP**
  > Please note that you can only assign graphs that do not have unsaved changes and contain at least one status.

- To change the current status of the entity and its expiry date, which are controlled by a graph, open the GraphStatusChange editor (page 58) of the respective graph and make the required changes.

> TIP
> Please note that changing the status of a graph forces this status and does not perform a transition into the status. This means that actions that would be performed during a transition, such as updating the retest date or a counter do not take place and need to be performed manually.

- To trigger a graph transition on the entity, thus performing all included actions and observing all conditions, click the **Trigger graph transition** button (page 75) to open the **Trigger Graph Transition** dialog (page 139).

> TIP
> Graph statuses and their expiry dates are runtime data that can change due to activities on the shop floor when the entity is in a read-only status, such as **Verification** or **Approved**. At regular intervals, the system checks all equipment entities with expirable statuses to determine if statuses have expired. If that is the case, it performs the status transitions as defined by the graph, but only on entities that are neither locked in Data Manager nor bound by execution.
> In order to resolve processing exceptions by administrative means, it is possible to update runtime data in these statuses in Data Manager as well. Before you change any runtime data, make sure to refresh the entity (page 72) so that it reflects the current situation on the shop floor.

- To change a graph's position in the list, click the **Move up** or **Move down** buttons (page 74).
- To remove a graph from the list, click the **Remove graph** button (page 75).

CLASS

The **Class** tab lists the equipment classes assigned to the equipment entity. An equipment entity can belong to several equipment classes, but takes its styling (color, icon, properties displayed on card view) from the class set as its **Style class**.

- To add a class to the list, drag and drop (page 100) its card or tile on the Details window of the entity to which you wish to assign it.

> TIP
> Please note that the system only checks if the properties of the assigned class correspond to those of your entity, but not on which tabs (**Specification**, **Process**, **Engineering**) the properties are located. Thus, the system does not rearrange the entity's properties to match the properties' positioning on the class.

- To define the styling of the entity by setting an assigned class a style class, select the **Style class** checkbox of the class whose styling the entity is to adopt.

- To remove a class from the list, select or multi-select it and click the **Remove equipment class** button (page 75) or press the DEL key.
  Removing a class from the list of classes also removes the entity from the list of entities shown on the **Entity** tab of the class' detail view.

GROUPING

The **Grouping** tab displays the grouping situation of the entity in two lists.

- The **Parent** list displays the data of the entity's parent entity. An entity can have only one parent, so the list always contains only one entry with the following data:
  - Identifier
  - Short description
  - Description
  - Status

  TIP
  The **Parent** list of an entity is always read-only. To change the parent assignment of an entity in Data Manager, you have to open the parent entity in the Details window and remove the respective child entity from its **Children** list.

- The **Children** list displays the data of all children that are directly or indirectly assigned to the entity. Indirect assignments occur when you assign an entity as child that is already a parent entity to other child entities. The following data is shown for child entities:
  - Identifier
  - Short description
  - Description
  - Status
  - Sub-parent
    Only available for entities that were assigned indirectly when their indicated parent was assigned to the entity.

  TIP
  Please note that you can only remove child entities from the list that have been assigned directly. For removing an indirectly assigned child entity you have to open its parent entity (indicated in the **Sub-parent** column) and remove the respective child entity from its **Children** list.

- To add a child entity to the list, drag and drop (page 100) its card or tile on the Details window of the entity to which you wish to assign it.
- To remove a child entity from the list, select or multi-select it and click the **Remove child entity** button (page 75) or press the DEL key.

TIP
Group assignments are runtime data that can change due to activities on the shop floor when the entity is in a read-only status, such as **Verification** or **Approved**. In order to resolve processing exceptions by administrative means, it is possible to update runtime data in these statuses in Data Manager as well. Before you change any runtime data, make sure to refresh the entity (page 72) so that it reflects the current situation on the shop floor.
A typical example of a group assignment change is a repair situation on the shop floor where an entity that is part of a group malfunctions during order processing and needs to be replaced.

CONTEXT

The **Context** tab lists the current and the previous material-related context information of the equipment entity, such as

- the product batch and material for which the entity was used,
- the order and order step during whose execution the entity was used,
- the involved material that was in direct contact with the entity,
- the order step input and order step output of the order step for which the entity was used.

TIP

Please note that the **Context** tab only contains the material-related context information for orders, not for workflows.

The information is read-only and shows the data for the current usage if the entity is bound. If it is not bound it shows the data for the entity's last usage. The complete usage history of the entity is recorded on the **Logbook** tab (page 247).

Logbook

The **Logbook** tab provides a table that lists all processing-related events in which the equipment entity was involved. For each event that affected the entity, the table displays one row.

To filter the event entries by date, select the date range you wish to consider. The system only displays the entries that have been registered between the two dates you have selected.
The date range starts at 12:00:00 AM on the date you have selected and ends at 23:59:59 PM on the date you have selected.
When you open the logbook, the date range is preset to start with the date when the last event occurred and to end with the current date. If the system has no events registered for the entity, the date range is preset to the current date.
To refresh the list of events, click the **Refresh** button (page 75). All events of the entity that have occurred since you opened the logbook are then included.
To print the **Equipment Logbook Report** for the events that have occurred during the selected date range, click the **Print equipment logbook report** button (page 74).

The following table columns are available and are filled if relevant to the respective event:

- Timestamp
  indicates the date and time when the event occurred.
- Category
  indicates the type of the event (**Binding**, **Denial**, **FSM**, **Context**, **Manual**, **Phase**, **Property update**, **Graph transition**, **Grouping**).
- Action/trigger
  for events that were triggered by external circumstances in the life cycle of the entity, the action that triggered the event (such as **Bind**, **Identify**) or any trigger of an assigned equipment graph.

  > TIP
  > Please note that for events of the **Graph transition** category, several trigger actions may be performed that change the same runtime property value. In this case, the logbook only records the initial and the final values of the property.

- Graph
  for events of the **Graph transition** category, indicates the graph that triggered the event.
- Property
  for events of the **FSM** category, the identifier of the FlexibleStateModel property type that governs the registered status change.

- Old content/New content
indicates the data changed by an event, such as **Identified**/**Bound** for an event of the **Binding** category.
- Information
for events of the **Denial**, **Manual**, **Graph transition**, **Property update**, or **Grouping** categories, additional information on the circumstances of the event.
- Workflow
for equipment entities used for processing a workflow, identifier of the workflow.
- Order
for equipment entities used for processing an order, identifier of the order.
- Product batch
for equipment entities used in batch production, the identifier of the batch that was being produced when the event occurred.
- Product material
for equipment entities used in production, the identifier of the material that was being produced when the event occurred.
- Unit procedure
identifier of the unit procedure (with its count if it was run more than once) that was being executed when the event occurred.
- Operation
identifier of the operation (with its count if it was run more than once) that was being executed when the event occurred.
- Phase
identifier of the phase (with its count if it was run more than once) that was being executed when the event occurred.
- Involved material
identifier of the material that was in direct contact with the entity when the event occurred.
- Work center
identifier of the work center where the event occurred.
- Station
identifier of the station where the event occurred.
- Device
identifier of the device where the event occurred.
- Signature
displays two items per signature (page 101):
    - User and login names of the user who performed the signature.
    - Timestamp when the signature was recorded.

- Logged-in user
  login name of the user under whose login the event occurred.
  If the logbook entry is created due to an update triggered by an external system, such as Warehouse Management, the user provided by the external system is displayed.

MIGRATED HISTORY

The **Migrated History** tab only exists for equipment entities that previously existed as data objects in PharmaSuite for Production Management and were migrated to Data Manager. It only contains the data recorded before the entity was migrated. Any events that occur after the migration are recorded in the entity's Logbook (page 247), Change History (page 249), and Status History (page 250) tabs.
A migrated history consists of two panels, one for the **Logbook** table at the top and the other one for the **Status Histories** table underneath, which comprises all status history event entries that were recorded for all status graphs that applied to the entity before it was migrated. So, for a scale, it contains both the event entries from the calibration status history and from the testing status history.

The panels are divided by a slidable separator and can thus be freely resized, if necessary.

To filter the change event entries by date, select the date range you wish to consider. The system only displays the entries that have been registered between the two dates you have selected.
The date range starts at 12:00:00 AM on the date you have selected and ends at 23:59:59 PM on the date you have selected.
To refresh the list of events, click the **Refresh** button (page 75).

CHANGE HISTORY

The **Change History** tab provides a list of all changes that have been performed on the equipment entity, which also includes indirect assignment changes that occur when a class' assignment change also involves the entity.
For each change event that affected the entity, the table displays one row.

To filter the change event entries by date, select the date range you wish to consider. The system only displays the entries that have been registered between the two dates you have selected.
The date range starts at 12:00:00 AM on the date you have selected and ends at 23:59:59 PM on the date you have selected.
When you open the change history, the date range is preset to start with the date when the last change event occurred and to end with the current date. If the system has no change events registered for the object, the date range is preset to the current date.
To refresh the list of change events, click the **Refresh** button (page 75). All change events involving the entity that have occurred since you opened the change history are then included.

The following table columns are available and are filled if relevant to the respective change event:

- Timestamp
  indicates the date and time when the change event occurred.
- Action
  indicates the change event that triggered the entry in the change history.
- Affected object
  indicates the object that is affected by the change event. So, when you update the basic data of your entity, the change history shows the identifier of the attribute you have modified and when you assign a class, graph, or property type to your entity, the change history shows the identifier of the assigned class, graph, or property type.
- Old content/New content
  for data change events on your entity, displays the changed values.
- Information
  indicates additional details on the change, such as if it was performed in the context of a mass change or a tag or point import.
- Logged-in user
  login name of the user under whose login the change event occurred.

STATUS HISTORY

The **Status History** tab provides a list of all status changes that have been performed on the equipment entity.
For each status change that affected the entity, the table displays one row.

To filter the status change entries by date, select the date range you wish to consider. The system only displays the entries that have been registered between the two dates you have selected.
The date range starts at 12:00:00 AM on the date you have selected and ends at 23:59:59 PM on the date you have selected.
When you open the status history, the date range is preset to start with the date when the last status change occurred and to end with the current date. If the system has no status changes registered for the object, the date range is preset to the current date.
To refresh the list of status changes, click the **Refresh** button (page 75). All status changes involving the entity that have occurred since you opened the status history are then included.

The following table columns are available and are filled if relevant to the respective status change:

- Timestamp
  indicates the date and time when the status change occurred.
- Action
  indicates the status change that triggered the entry in the status history.
- To status
  indicates the status to which the entity was moved.
- Signature
  displays two items per signature (page 101):
  - User and login names of the user who performed the signature.
  - Timestamp when the signature was recorded.
- Comment
  displays the comment recorded for each signature. For double signatures, each comment is prefixed with **Comment 1** or **Comment 2** to indicate with which signature the comment was recorded.

GRAPH

The Details window of a graph contains the following data tabs:

- Basic (page 251)
- Property Type (page 253)
- Status/Trigger (page 253)
- Transition (page 255)
- Style (page 260)
- Change History (page 260)
- Status History (page 261)

BASIC

The **Basic** tab lists the Basic Attributes of the equipment graph.

Click an attribute value to edit it. The system displays a text cursor or a button to access a suitable editor (page 10) for the required value type.

> **TIP**
>
> Please note that once you have created an equipment graph, you can no longer modify its **Identifier**.

DEFAULT GRAPHS

PharmaSuite provides the following equipment graphs with any default installation of the system:

- ContainerCleaning_RS_1
  The default graph for container cleaning includes the following default property types:
  - Container Clean Shelflife (RS)
  - Container Sublot (RS)
- RoomCleaning_RS_1
  The default graph for room cleaning includes the following default property type:
  - General Clean Shelflife (RS)
- ScaleCalibration_RS_1
  The default graph for scale calibration includes the following property type:
  - Scale Calibration Shelflife (RS)
- ScaleTest_RS_1 (scale test graph)
  The default graph for scale test includes the following property type:
  - Scale Test Shelflife (RS)

The **Display text** attribute defines the text shown on the card view of the graph.

The **Purpose** attribute is mandatory and indicates the context for which the graph is intended.
Especially the Dispense and Weighing phases of PharmaSuite require equipment to hold graphs of specific purposes in order to function correctly. The Dispense and Weighing-related purposes are marked with **(RS)**.
The following purposes are available:

- **Calibration**
  for graphs related to the calibration of equipment.
- **Cleaning**
  for graphs related to the cleaning of equipment.
- a **Container Cleaning (RS)** graph is required to control that vessels identified during production are fit for use.
- **Installation**
  for graphs related to installation processes for equipment.
- **Maintenance**
  for graphs related to maintenance processes for equipment.

- a **Room Cleaning (RS)** graph is required to calculate to which extent a room needs to be cleaned prior to a weighing process.
- a **Scale Calibration (RS)** graph is required to qualify scales prior to a weighing process.
- a **Scale Test (RS)** graph is required to qualify scales prior to a weighing process.
- **Sterilization**
  for graphs related to the sterilization of equipment.
- **Testing**
  for graphs related to testing processes for equipment.
- **Usage**
  for graphs related to the usage of equipment.
- If PharmaSuite is configured to communicate with Warehouse Management (page 123), a **Warehouse Load Check (RS)** graph is required to enable Warehouse Management to check if loading a logistic unit is allowed by to the status of its corresponding equipment entity.

PROPERTY TYPE

The **Property Type** tab lists the property types assigned to the equipment graph.

- To add a property type to the list, drag and drop (page 100) its card or tile on the Details window of the graph to which you wish to assign it.
- To change a property type's position in the list, click the **Move up** or **Move down** buttons (page 74).
- To display a property type on the card view of the equipment graph, click the **Display on card** button (page 73).
- To remove a property type from the list, click the **Remove property type** button (page 75).

STATUS/TRIGGER

The **Status/Trigger** tab provides two collapsible lists to define

- the individual statuses of the equipment graph consisting of
  - the **Key**, which is mandatory and must be unique within the list of statuses,
  - the **Display text**, which is how the status is shown to operators during processing,
  - a **Description** to further elaborate the status's meaning, and
  - the **Initial status** option, which defines which status an equipment entity receives initially when the graph is assigned to it.

    - the **Can expire** option, which defines if a status can expire automatically after a specified duration or at a specific date.
- the triggers that affect the graph's statuses, consisting of
    - the **Key**, which is mandatory and must be unique within the list of triggers,
    - the **Display text**, which is how the trigger is shown to operators during processing, and
    - a **Description** to further elaborate the trigger's meaning.

> TIP
> Please note that the system automatically adds the **Expired (RS)** trigger as soon as you select the **Can expire** option for one of the graph's statuses. You cannot manually delete the system-provided **Expired (RS)** trigger, but it is removed automatically when you have unselected the **Can expire** option of all statuses.

To add a new status or trigger, type its data in the empty table row, starting with its key. The system automatically adds a new empty list row after you have completed the first cell of the new status or trigger.

> TIP
> Please note that curly brackets (braces) and leading or trailing blanks are not allowed in the keys of statuses or triggers.

To delete a status or trigger, select or multi-select it and click the **Delete status** or **Delete trigger** button, respectively, or press the DEL key.

> TIP
> Statuses or triggers that contain errors are indicated by an error marker (page 76) to the left of the respective rows. Hover over the icon to view a tooltip with additional error details.

Load and unload synchronization with Warehouse Management

If PharmaSuite is configured to communicate with Warehouse Management, equipment entities that are movable and hold a sublot are typically represented as a logistic unit in Warehouse Management.

To check if loading or unloading a warehouse-managed logistic unit is permitted by the status of its corresponding equipment entity with sublots, a graph of the **Warehouse Load Check (RS)** purpose must be assigned to those entities that correspond to logistic units. The graph must contain transitions to model the situation when the first load is placed on the logistic unit and when the last load is removed from the logistic unit.
For these transitions the system expects two triggers with the following trigger keys:

- **FIRST_LOAD**
  Loading an empty logistic unit with a sublot in Warehouse Management is only completed if a graph transition, caused by the **First load** trigger of an equipment graph with the **Warehouse Load Check (RS)** purpose, has been performed successfully on the equipment entity.
- **TOTAL_UNLOAD**
  Removing the last sublot from a logistic unit in Warehouse Management is only completed if a graph transition, caused by the **Total unload** trigger of an equipment graph with the **Warehouse Load Check (RS)** purpose, has been performed successfully on the equipment entity.

Transition

The **Transition** tab provides a list to specify the structure and logic of the equipment graph by combining

- a **Trigger**, which needs to have been defined on the **Status/Trigger** tab (page 191) before.
- the statuses (**From status** and **To status**), also previously defined on the **Status/Trigger** tab (page 191).
- an optional transition **Condition** that determines if the expected circumstances are met so that the transition can be executed. A condition is defined as condition expression in the Expression editor for transition conditions (page 30).
  The **Condition** column of a transition is blank if there is no condition defined. If a transition condition exists, it shows the description of the condition. If the condition does not have a description, the system displays the condition expression itself, introduced by an equals sign.

> **TIP**
> In the transition table, the description of a condition is not editable. To change it, you have to open the Expression editor and make your changes in the **Description** box.

- one or more **Actions** that are performed along with the transition. Actions are also optional and are displayed in a collapsible sub-table under the transition row. They are defined in the Expression editor for transition actions (page 31).

  > TIP
  > Actions are executed in the order in which they are listed. Use the **Move action down/up** buttons (page 74) to change the order of the actions.

- the identifier (**ID**) of the transition, which is shown for exception cases during execution, in the batch report, and in the logbook and change history of affected equipment entities.

Thus, a transition specifies the cause, represented by the **Trigger**, that initiates the transition from one status, the **From status**, to another status, the **To status**, including potential preconditions and resulting actions.

Graph Transition Modeling with Expiry Dates and Triggers

In addition to the triggers that you define yourself, the system creates the **Expired (RS)** trigger, when you define a status that can expire. Along with the trigger the system provides an **ExpiryDate** reference for use in the Expression editor in conditions or actions.

Expiry triggers are evaluated automatically during execution whenever an operator identifies an equipment entity in order to update its status and thus ensure that the entity is suitable for use. If the trigger evaluation determines that the identified entity has expired into a status that is unfit for use, the system rejects the entity. In this case, the entity has to undergo a specific treatment that returns it to the suitable status.
In the case of a container, this could mean that its "Clean" status has expired, and it needs to be run through a cleaning workflow. In the case of a scale, its "Tested" or "Calibrated" statuses may have expired so that the scale requires testing and calibration before it can be used in production again.

Your graph may contain several statuses that can expire, so that one status expires into a successor status that again expires into a successor status, thus forming a cascade of expiries until the graph has reaches a status that cannot expire or has not yet expired.

TIP

For creating the transitions of a graph with expiry-enabled statuses, we recommend to proceed as follows:

1. Create all transitions required to cover all statuses using all user-created triggers.
2. Identify which transitions have an expiry-enabled **To status** and add a **set** action that defines an expiry date.
3. Use the **Expired (RS)** trigger to create transitions out of expiry-enabled statuses, which have not been handled yet. This may be necessary to define various conditions that lead to different target statuses.
   Make sure to create an action for setting an expiry date when a transition has an expiry-enabled **To status**.

The following example shows a cleaning graph for equipment entities with four statuses, three of which can expire:

- **To be cleaned** (UNCLEAN)
  is the initial status and cannot expire. From here you can only transition into the **Clean** status.
- **Clean** (CLEAN)
  can expire. From here you can transition to the **To be cleaned**, the **Clean**, the **In use**, and the **Reusable** statuses.
- **In use** (IN_USE)
  can expire. From here you can transition to the **Reusable** and **To be cleaned** statuses.
- **Reusable** (REUSABLE)
  can expire. From here you can transition to the **In use** and **To be cleaned** statuses.

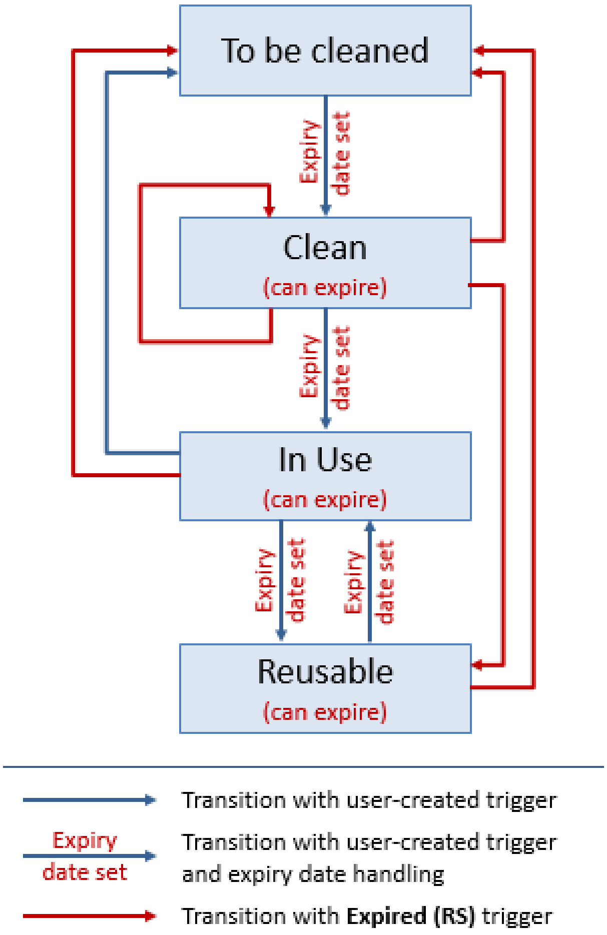


Figure 107: Example graph

The expiry dates of explicit transitions in the example are handled as follows:

- Setting the expiry date
  The expiry date for the **Clean** status is specified as action with the transition from **To be cleaned** to **Clean**. It uses the `set` command to define that the `{ExpiryDate}` reference is set to twelve hours after the date and time when the transition occurs. To retrieve the time, the action expression uses the **Current Date and Time (now)** function from the **Analysis** section.

  ```
  set {ExpiryDate} = now() + 12h
  ```

- Retaining the expiry date
  To retain the expiry date of the equipment entity when it moves from the **Clean** or **Reusable** statuses to the **In use** status, the `{ExpiryDate}` reference is set to itself:

  ```
  set {ExpiryDate} = {ExpiryDate}
  ```

- Postponing the expiry date
  When the equipment entity moves from the **In use** status to the **Reusable** status, its expiry date is postponed by two hours:
  `set {ExpiryDate} = {ExpiryDate} + 2h`

Implicit transitions with **Expired (RS)** triggers handle the following cases:

> TIP
> When calculating a new expiry date in a transition of the **Expired (RS)** trigger, we recommend not to use the **now()** function. Since the system executes the expiry trigger when an (expired) equipment entity is identified during production, which is not necessarily a predictable point in time and is thus not really suitable as base of a calculation. The expiry date of the **From status**, on the other hand, is a defined and immutable point in time, which makes it a good base for calculating the expiry date of the **To status**:
> `set {ExpiryDate} = {ExpiryDate} + <duration>`

- Prolonging the time before an equipment entity is unavailable for use due to an expired status
  Under a specific condition, the **Clean** status expires into itself, adding time to its set expiry date:
  `set {ExpiryDate} = {ExpiryDate} + 12h`
- Prolonging the time before an equipment entity is unavailable for use after it has expired into another status
  Under a specific condition, the **Clean** status expires into the **Reusable** status, adding time to its set expiry date:
  `set {ExpiryDate} = {ExpiryDate} + 2h`
- Automatically moving an equipment entity to a status that makes it unavailable for use
  Under specific conditions, the **Clean**, **In use**, or **Reusable** statuses expire into the **To be cleaned** status, which cannot expire and thus does not require an action that handles an expiry date.

To add a new transition, select the required entries from the respective option lists or start to define a new condition. The system automatically adds a new empty list row after you have completed any cell of the new transition.

To delete a transition, select or multi-select it and click the **Delete transition** button or press the DEL key.

> TIP
> Transitions that contain errors are indicated by an error marker (page 76) to the left of the respective table or sub-table rows. Hover over the icon to view a tooltip with additional error details.

Style

The **Style** tab provides tools for defining the visual appearance of the equipment graph's tiles and cards:

- Background color
- Font color

Change History

The **Change History** tab provides a list of all changes that have been performed on the equipment graph.
For each change event that affected the graph, the table displays one row.

To filter the change event entries by date, select the date range you wish to consider. The system only displays the entries that have been registered between the two dates you have selected.
The date range starts at 12:00:00 AM on the date you have selected and ends at 23:59:59 PM on the date you have selected.
When you open the change history, the date range is preset to start with the date when the last change event occurred and to end with the current date. If the system has no change events registered for the object, the date range is preset to the current date.
To refresh the list of change events, click the **Refresh** button (page 75). All change events involving the graph that have occurred since you opened the change history are then included.

The following table columns are available and are filled if relevant to the respective change event:

- Timestamp
  indicates the date and time when the change event occurred.
- Action
  indicates the change event that triggered the entry in the change history.
- Affected object
  indicates the object that is affected by the change event. So, when you update the basic data of your graph, the change history shows the identifier of the attribute you have modified. When you assign a property type to your graph, the change history shows the identifier of the assigned property type. For status or trigger changes, the affected object is displayed as <key>/<display text>. Similar formatting is used for transition attributes which show as <ID>/<attribute>.
- Old content/New content
  for data change events on your graph, displays the changed values.
- Logged-in user
  login name of the user under whose login the change event occurred.

STATUS HISTORY

The **Status History** tab provides a list of all status changes that have been performed on the equipment graph.
For each status change that affected the graph, the table displays one row.

To filter the status change entries by date, select the date range you wish to consider. The system only displays the entries that have been registered between the two dates you have selected.
The date range starts at 12:00:00 AM on the date you have selected and ends at 23:59:59 PM on the date you have selected.
When you open the status history, the date range is preset to start with the date when the last status change occurred and to end with the current date. If the system has no status changes registered for the object, the date range is preset to the current date.
To refresh the list of status changes, click the **Refresh** button (page 75). All status changes involving the graph that have occurred since you opened the status history are then included.

The following table columns are available and are filled if relevant to the respective status change:

- Timestamp
  indicates the date and time when the status change occurred.
- Action
  indicates the status change that triggered the entry in the status history.
- To status
  indicates the status to which the graph was moved.
- Signature
  displays two items per signature (page 101):
  - User and login names of the user who performed the signature.
  - Timestamp when the signature was recorded.
- Comment
  displays the comment recorded for each signature. For double signatures, each comment is prefixed with **Comment 1** or **Comment 2** to indicate with which signature the comment was recorded.

PROPERTY TYPE

Property types are used on the one hand to model the properties inherent to equipment entities and classes and on the other hand to define the equipment requirements presented by master recipes and workflows. A property type describes a physical characteristic of an equipment entity, such as its volume, speed range, or the fact that it is under cleaning status control. For this reason, each property type can only be assigned once to an equipment entity, template entity, or class.
As a consequence, when you model an equipment entity that holds two tanks of the same volume, you still need to define two property types, one for each tank. Similarly, you also need to define two property types to model different functions the volume of one and the same tank may have, such as its storage volume, which may be greater than its fermentation volume.

During execution, an equipment entity is matched against the requirements defined with the recipe or workflow for which an operator plans to use it. Only if this validation check concludes that the properties of the equipment entity meet the requirements can the entity be used as planned.

VALIDATION RULES

The rules that determine if an equipment entity matches its requirements take place on three levels of complexity:

- The basic level only has one rule, which checks if all properties defined for the requirement are also defined as properties of the equipment entity.
- The second level has a set of rules that apply if requirement and entity both state a value for a property, if one of them states a range while the other supplies a value, or if one or both of them state neither a value nor a range for a property.

Condition	Result
Requirement: value and range are not defined	The entity is a **valid** match for the requirement.
Requirement: value or range is defined **Entity:** value and range are not defined	The entity is **no valid** match for the requirement.
Requirement: value is defined (ReqValue) **Entity:** value is defined (EntValue) ReqValue = EntValue	The entity is a **valid** match for the requirement.

Condition	Result
Requirement: range is defined (ReqLowLimit and/or ReqHighLimit) **Entity:** value is defined (EntValue) ReqLowLimit <= EntValue <= ReqHighLimit	The entity is a **valid** match for the requirement.
Requirement: value is defined (ReqValue) **Entity:** range is defined (EntLowLimit and/or EntHighLimit) EntLowLimit <= ReqValue <= EntHighLimit	The entity is a **valid** match for the requirement.
Requirement: value is not defined **Entity:** value is not defined **Either requirement or entity:** one value (Low or High) of range is defined	Covered by third level of rules, see below.
Either requirement or entity: value is defined range is defined (Low and/or High)	This condition represents an invalid configuration and is prevented by the system.

- The set of rules on the third level applies to cases when both requirement and entity have a range defined for a property.
  When matching the range of an entity (EntLowLimit, EntHighLimit) against the range of a requirement (ReqLowLimit, ReqHighLimit) the system has to consider the following possible cases:
  - Entity range is **lower** than requirement range:
    EntHighLimit < ReqLowLimit
  - Entity range **overlaps** requirement range at the **low limit**:
    EntLowLimit < ReqLowLimit <= EntHighLimit <= ReqHighLimit
  - Entity range **is contained** in requirement range:
    ReqLowLimit <= EntLowLimit <= EntHighLimit <= ReqHighLimit
  - Entity range **contains** requirement range:
    EntLowLimit <= ReqLowLimit <= ReqHighLimit <= EntHighLimit
  - Entity range **overlaps** requirement range at the **high limit**:
    ReqLowLimit <= EquLowLimit <= ReqHighLimit < EquHighLimit

- Entity range is **higher** than requirement range:
  ReqHighLimit < EntLowLimit

The following table lists how the system evaluates the settings made for entity and requirement ranges and indicates the match results.
If a range only has one limit, the system treats it as open range.

Limit settings specified				Range check type of requirement	
ReqLow	**ReqHigh**	**EntLow**	**EntHigh**	**Is contained in entity range**	**Contains entity range**
---	---	---	---	Valid match	Valid match
---	---	---	Yes	Valid match	Valid match
---	---	Yes	---	Valid match	Valid match
---	---	Yes	Yes	Valid match	Valid match
---	Yes	---	---	No valid match	No valid match
---	Yes	---	Yes	Valid match, if EntHigh <= ReqHigh	Valid match, if ReqHigh <= EntHigh
---	Yes	Yes	---	No valid match	No valid match
---	Yes	Yes	Yes	Valid match, if EntHigh <= ReqHigh	No valid match
Yes	---	---	---	No valid match	No valid match
Yes	---	---	Yes	No valid match	No valid match
Yes	---	Yes	---	Valid match, if ReqLow <= EntLow	Valid match, if EntLow <= ReqLow
Yes	---	Yes	Yes	Valid match, if ReqLow <= EntLow	No valid match
Yes	Yes	---	---	No valid match	No valid match
Yes	Yes	---	Yes	No valid match	Valid match, if Equ.max <= Req.max
Yes	Yes	Yes	---	No valid match	Valid match, if Req.min <= Equ.min
Yes	Yes	Yes	Yes	Valid match, if ReqLow <= EntLow **AND** EntHigh <= ReqHigh	Valid match, if EntLow <= ReqLow **AND** ReqHigh <= EntHigh

The Details window of a property type contains a **Basic** tab that lists its Basic Attributes.

Click an attribute value to edit it. The system displays a text cursor or a button to access a suitable editor (page 10) for the required value type.

TIP

Please note that once you have created, defined, and saved a property type, you cannot edit it anymore.

DEFAULT PROPERTY TYPES

PharmaSuite provides the following property types with any default installation of the system:

- Container Clean Shelflife (RS)
  a Specification property type of the Duration data type
- Container Sublot (RS)
  a Runtime property type of the String data type and the Current Sublot (RS) purpose
- Container Tare (RS)
  a Runtime property type of the MeasuredValue data type and the Current Tare (RS) purpose
- Container Type (RS)
  a Specification property type of the Container (RS) equipment type
- General Clean Shelflife (RS)
  a Specification property type of the Duration data type
- Room Cleaning Rules (RS)
  a Specification property type of the RoomCleaningRules data type
- Room Type (RS)
  a Specification property type of the Room (RS) equipment type
- Scale Calibration Shelflife (RS)
  a Specification property type of the Duration data type
- Scale Configuration (RS)
  a Specification property type of the ScaleConfiguration data type
- Scale Load (RS)
  a Runtime property type of the String data type and the Current load (RS) purpose
- Scale Ranges (RS)
  a Specification property type of the Ranges data type and the ScaleRanges (RS) purpose
- Scale Test and Calibration (RS)
  a Specification property type of the ScaleTestAndCalibration data type

- Scale Test Shelflife (RS)
a Specification property type of the Duration data type
- Scale Type (RS)
a Specification property type of the Scale (RS) equipment type
- Work Center Assignment (RS)
a Specification property type of the WorkCenterAssignment data type

The purpose and context of a property type are defined by several attributes:

- with the **Usage** attribute you determine if a property type
  - is part of the basic **Specification** of the equipment, indicating a characteristic of an equipment object, such as a coater's drum size,
  - is intended for equipment in an **Automation** environment, so that you need to specify automation-related data, such as tag names, tag update rates, or Live Data types,
  - if it refers to a **Runtime** value of an equipment entity that can change during processing, such as a coater's cleaning status,
  - or is intended for equipment that is connected to a **Historian** server, so that you need to specify which tags are to be read for retrieving the historical data chart.
- with the **Purpose** attribute you can further specify some data types that represent required properties for equipment to be used with the Dispense and Weighing phases of PharmaSuite or with phases that generate equipment entities on the shop floor.
  - the **Base Sublot (RS)** purpose extends the String (page 269) data type in the **Runtime** usage and is required to register the identified sublot so that the system can determine which template equipment entity to use when generating an entity.
  - the **Current Tare (RS)** purpose extends the MeasuredValue (page 269) data type in the **Runtime** usage and is required to handle the tare property of a container.
  - the **Reference Tare (RS)** purpose extends the MeasuredValue (page 269) data type in the **Runtime** usage and is used for defining the optional reference tare property of a container.
  - the **Current Sublot (RS)** purpose extends the String (page 269) data type in the **Runtime** usage and is required for identifying a container.
  - the **Scale Ranges (RS)** purpose extends the Ranges (page 269) data type and is required to configure the ranges of a scale.

  - the **Current Load (RS)** purpose extends the String (page 269) data type in the **Runtime** usage and is required to handle the identifier of a container or sublot that represents a scale load.

- with the **Data type** attribute you determine the context for which the property type is suitable. Once you have selected the **Data type**, you can specify its **Content**.

  The system provides the following data types:

  - BigDecimal
    floating point number, used for numbers and ranges. It is suitable for handling numbers qualified by a unit of measure.
    BigDecimal for automation can handle the three Live Data types: **Double**, **Float**, and **Integer** that need to be specified for correct communication with equipment. The mapping of the most common OPC names to the Live Data types is as follows:

    - **Integer** represents SHORT, LONG, CHAR, BYTE, USHORT, and ULONG
    - **Float** represents REAL4
    - **Double** represents REAL8

> TIP
> BigDecimal values support up to 15 integral digits and are rounded to 9 fractional digits when stored to the database.

    For editing BigDecimal values, the system provides several BigDecimal editors:

    - BigDecimal (page 11)
    - Automation BigDecimal (Double) (page 13)
    - Automation BigDecimal (Float) (page 17)
    - Automation BigDecimal (Integer) (page 20)
    - Historian BigDecimal (page 24)

  - Boolean
    with the values **Yes**, **No**, and in some cases **N/A**.
    For editing Boolean values for automated equipment, the system provides the Automation Boolean (page 25) editor.

- Duration
  only available for **Specification** and **Runtime** properties.
  It is used for displaying time spans and for time-related calculations.
  It supports the following time units: days (**d**), hours (**h**), minutes (**min**), seconds (**s**), and milliseconds (**ms**).
  For editing durations, the system provides the Duration (page 29) editor.

- EquipmentType
  only available for **Specification** property types.
  It is used to support you with the accurate and complete configuration of equipment entities (page 123) to be used for executing product phases, such as Dispense and Weighing phases.

  TIP
  When an entity holds an EquipmentType property, the system checks if it also holds all other properties or graphs required by its type. You can save an entity that does not hold all required properties or graphs, but you cannot change its status to **Verification** or **Approved**.

  The following equipment types are available:

  - Container (RS)
  - Hybrid (RS)
  - Room (RS)
  - Scale (RS)

- FlexibleAttributeDefinition
  only available for **Runtime** property types.
  It is used to define a bundle of runtime attributes, whose values are established and can change during processing.
  For defining the attribute bundle, the system provides the FlexibleAttributeDefinition (page 53) editor.

- FlexibleStateModel
  only available for **Runtime** property types.
  It is used to put equipment entities under status control, for example, to model cleaning or maintenance cycles.
  Suitable FSMs must be maintained in the Process Designer application of FactoryTalk ProductionCentre. For further information on FSMs for use with equipment, please refer to "Configuring FSMs for Equipment Properties" in Volume 3 of the "Technical Manual Configuration and Extension".
  For selecting a FlexibleStateModel for equipment, the system provides the FlexibleStateModel (page 55) editor.

- FlexibleTagDefinition
  only available for **Automation** property types.
  It is used to define a group of automation-relevant values to communicate as value bundle from an equipment entity for monitoring purposes.
  For defining the value bundle, the system provides the Automation FlexibleTagDefinition (page 56) editor.

- MeasuredValue
  used for displaying numeric values qualified by a unit of measure.

> TIP
> MeasuredValue values support up to 15 integral digits and are rounded to 9 fractional digits when stored to the database.

  For editing MeasuredValue data types, the system provides the MeasuredValue (page 59) editor.

- Ranges
  only available for **Specification** property types.
  It is used to define up to three ranges for an equipment object.
  For defining the ranges, the system provides the Ranges (page 60) editor.

- RoomCleaningRules
  only available for **Specification** property types.
  It is used to define the cleaning rules and cleaning demands of rooms used in a Dispense environment.
  For cleaning rule data, the system provides the CleaningRules (page 26) editor.

- ScaleConfiguration
  only available for **Specification** property types.
  It is used to define the driver and connection data of a scale.
  For defining the scale data, the system provides the ScaleConfiguration (page 61) editor.

- ScaleTestAndCalibration
  only available for **Specification** property types.
  It is used to define the test weight and operator instructions for scale tests and calibration.
  For scale test and calibration data, the system provides the ScaleTestAndCalibration (page 65) editor.

- String
  used for handling any sequence of characters.

- WorkCenterAssignment
  only available for **Specification** property types.
  It is used to assign work centers to scales or rooms. Work centers are maintained in the Data Manager - Work Center mode of Data Manager.
  For selecting a work center, the system provides the Searchable Option List (page 67) editor.

Managing Work Centers

Data Manager provides a comprehensive framework for creating and managing all data you need to maintain the work centers and stations used for processing recipes or workflows with PharmaSuite.

What Is a Work Center?

A work center can be characterized as a suitably equipped location or area within the operational system where personnel utilizes machines, tools, and devices. A work center can comprise one or more stations.

Work center-related data is relevant to several other objects in the production environment.

- Storage areas for order-related material receipt.
- Dispense-related equipment, such as scales and rooms, which are maintained as equipment entities (page 123) in Data Manager - Equipment.

What Is a Station?

A station represents an operator's access point to execution activities with PharmaSuite at a work center. To actually perform execution activities, an operator uses a stationary or mobile device registered at the station. It is possible to register more than one device at a station.

Menus and Toolbars

You can access all relevant functions either from the menu bar in the main Data Manager - Work Center window, from context-sensitive shortcut menus, or from toolbars provided for quick access to frequently used functions.

Menus

Data Manager - Work Center provides a main menu bar (page 271) with all relevant functions as well as context-sensitive shortcut menus (page 275) for quick access to specific functions.

Main Menu Bar

The **main menu bar** offers the following menus and functions:

File

- New work center (CTRL+SHIFT+W)
  Creates a new work center, thus first opening the **New Work Center** dialog and afterwards a new tab in the lower tab bar of the Details window.
- New station (CTRL+SHIFT+T)
  Creates a new station, thus first opening the **New Station** dialog and afterwards a new tab in the lower tab bar of the Details window.
- Save <object identifier> (CTRL+S)
  Saves all changes made to the object that is currently active in the lower tab bar of the Details window. This action saves all changes made on the object's attributes and properties tabs.
- Save all (CTRL+SHIFT+S)
  Saves all changes made to all object that are currently active in the lower tab bar of the Details window. This action saves all changes made on the objects' properties tabs.

  TIP
  Please note that you can only save an object if none of its assigned objects have unsaved changes.
- Duplicate <work center identifier> (CTRL+F12)
  Only available for work centers.
  Opens the **Duplicate Work Center** dialog to create a copy of the work center. The new work center opens as new tab in the lower tab bar of the Details window.
  The **Duplicate** function copies all attributes and properties of the original work center except for its station assignments.
- Duplicate <station identifier> (CTRL+F12)
  Only available for stations.
  Opens the **Duplicate Station** dialog to create a copy of the station. The new station opens as new tab in the lower tab bar of the Details window.
  The **Duplicate** function copies all attributes and properties of the original station except for its work center assignment.
- Close <object identifier> (CTRL+F4)
  Closes the object that is currently active in the lower tab bar of the Details window.
- Close all (CTRL+SHIFT+F4)
  Closes all objects that are currently open in the lower tab bar of the Details window.
- Restore <object identifier> (CTRL+R)
  Only available if there are unsaved changes.
  Restores the object that is currently active in the lower tab bar of the Details window to its last saved state, thus undoing all changes made in the meantime.

- Refresh <object identifier> (F5)
  Only available if there are no unsaved changes.
  Refreshes the object that is currently active in the lower tab bar of the Details window by retrieving its data again from the database.
- Delete <object identifier>
  Deletes the object that is currently active in the lower tab bar of the Details window.
- Print barcode label for <object identifier> (CTRL+B)
  Only available for stations.
  Sends the barcode of the station that is currently active in the Details window for printing to a connected printer.
- Exit (ALT+F4)
  Closes the application window.

View

- New search (CTRL+N)
  Opens a new Search window to define another set of search criteria and process its search results.
- Zoom in (CTRL+PLUS)
  Zooms in on the currently active search result, switching the display to the next larger view, either tiles or cards.
- Zoom out (CTRL+MINUS)
  Zooms out from the currently active search result, switching the display to the next smaller view, either tiles or mini tiles.
- Card tooltips (CTRL+T)
  Toggles the display of the card view as tooltip when you hover over a tile or mini tile.

Window

- Equipment Manager - Smart Search
  Switches to the Equipment - Smart Search mode of Data Manager. Before switching, the system closes all open objects, prompting you to save if you have unsaved changes.
  A checkmark to the left of the selected menu option indicates which mode is currently active.
  Only available if you have the user rights necessary for accessing Data Manager - Equipment - Smart Search.

- Work Center Manager
  Switches to the Work Center mode of Data Manager. Before switching, the system closes all open objects, prompting you to save if you have unsaved changes.
  A checkmark to the left of the selected menu option indicates which mode is currently active.
  Only available if you have the user rights necessary for accessing Data Manager - Work Center.
- Undo layout change
  Revokes the last layout change you have performed. You can undo up to 100 actions, thus you can step by step revoke the last 100 layout changes you have performed. Once you have performed a layout change, the menu function changes to a more precise description of the undo action, such as **Undo resizing** or **Undo dragging**.
- Redo layout change
  Redoes the last layout change you have revoked with the **Undo layout change** function. You can redo up to 100 actions, thus you can step by step redo the last 100 layout changes you have revoked. Once you have performed a layout change, the menu function changes to a more precise description of the undo action, such as **Redo resizing** or **Redo dragging**.
- Save user layout
  Saves the current window layout and overwrites the layout that was last saved by you on this computer.
- Load user layout
  Loads the last layout you have saved on this computer with the **Save user layout** function.
- Reset layout
  Resets the window layout to the system-defined default layout. This function does not affect the saved user layout, which can be restored with the **Load user layout** function.

Help

- Data Manager Help (ALT+F1)
  Opens a web browser to display the start page of the help system (page 7).
- About PharmaSuite
  Opens the **About PharmaSuite** dialog (page 8).

Shortcut Menus

The card and tile views make object type-specific **shortcut menus** available, which you can access by right-clicking a card or tile in the search results panel.

Work center

- Open <work center identifier> in Details window
  Opens the detail view of the work center as new tab in the Details window.
- Duplicate <work center identifier>
  Opens the **Duplicate Work Center** dialog to create a copy of the work center. The new work center opens as new tab in the lower tab bar of the Details window.
  The **Duplicate** function copies all attributes and properties of the original work center except for its station assignments.
- New work center
  Creates a new work center, thus first opening the **New Work Center** dialog and afterwards a new tab in the lower tab bar of the Details window.
- Restore <work center identifier>
  Only available if there are unsaved changes.
  Restores the work center to its last saved state, thus undoing all changes made in the meantime.
- Refresh <work center identifier>
  Only available if there are no unsaved changes.
  Refreshes the work center by retrieving its data again from the database.
- Delete <work center identifier>
  Deletes the work center

Station

- Open <station identifier> in Details window
  Opens the detail view of the station as new tab in the Details window.
- Duplicate <station identifier>
  Opens the **Duplicate Station** dialog to create a copy of the station. The new station opens as new tab in the lower tab bar of the Details window.
  The **Duplicate** function copies all attributes and properties of the original station except for its work center assignment.
- New station
  Creates a new station, thus first opening the **New Station** dialog and afterwards a new tab in the lower tab bar of the Details window.

- Restore <station identifier>
  Only available if there are unsaved changes.
  Restores the station to its last saved state, thus undoing all changes made in the meantime.
- Refresh <station identifier>
  Only available if there are no unsaved changes.
  Refreshes the station by retrieving its data again from the database.
- Delete <station identifier>
  Deletes the station

Toolbars

Data Manager provides several toolbars for context-sensitive quick access to specific functions.

Main Toolbars

The **main toolbars** offer shortcuts to the more frequently required menu functions:

- File toolbar with
  - New work center (page 72)
  - New station (page 72)
  - Refresh (page 72)
  - Save (page 72)
  - Close (page 71)
  - Delete (page 71)
  - Print barcode label (page 72)
- View toolbar with
  - Zoom in (page 73)
  - Zoom out (page 73)
- Help toolbar with
  - Help (page 71)

Detail Data

From the upper tab bar in the Details window, you can access the attributes, assignments, and usage data of the following work center-related objects:

- Work centers (page 277)
- Stations (page 279)

Work Center

The Details window of a work center contains the following data tabs:

- Basic (page 277)
- Station (page 277)
- Change History (page 278)

BASIC

The **Basic** tab lists the attributes of the work center.

Click an attribute value to edit it. The system displays a text cursor or a button to access a suitable editor (page 10) for the required value type.

> TIP
>
> Please note that once you have created a work center, you can no longer modify its **Identifier**. The maximum length of a work center identifier is 20 characters.

With the **Type** attribute, you can define if your work center is available for Dispense operations. Dispense work centers can only have one station.

The **Storage area** attribute allows you to select the storage area associated with the work center. As this list can be very long, the system supports you with a searchable option list (page 67) editor.

STATION

The **Station** tab lists the stations assigned to the work center.

- To add a station to the list, drag and drop (page 100) its card or tile on the Details window of the work center to which you wish to assign it.
- To remove a station from the list, select or multi-select it and click the **Remove station** button or press the DEL key.
  Removing a station from the list of stations also clears the **Work center** attribute shown on the **Basic** tab of the station's detail view.

TIP

Please note that you can assign no more than one station to a work center of the **Dispense** type.

CHANGE HISTORY

The **Change History** tab provides a list of all changes that have been performed on the work center, including its station assignments.
For each change event that affected the work center, the table displays one row.

To filter the change event entries by date, select the date range you wish to consider. The system only displays the entries that have been registered between the two dates you have selected.
The date range starts at 12:00:00 AM on the date you have selected and ends at 23:59:59 PM on the date you have selected.
When you open the change history, the date range is preset to start with the date when the last change event occurred and to end with the current date. If the system has no change events registered for a work center, the date range is preset to the current date.
To refresh the list of change events, click the **Refresh** button (page 75). All change events involving the work center that have occurred since you opened the change history are then included.

The following table columns are available and are filled if relevant to the respective change event:

- Timestamp
  indicates the date and time when the change event occurred.
- Action
  indicates the change event that triggered the entry in the change history.
- Affected object
  indicates the object that is affected by the change event. So, when you update the basic data of your work center, the change history shows the identifier of the attribute you have modified and when you assign a station to your work center, the change history shows the identifier of the station.
- Old content/New content
  for data change events on your work center, displays the changed values.
- Logged-in user
  login name of the user under whose login the change event occurred.

Station

The Details window of a station contains the following data tabs:

- Basic (page 279)
- Change History (page 280)

Basic

The **Basic** tab lists the attributes of the station.

Click an attribute value to edit it. The system displays a text cursor or a button to access a suitable editor (page 10) for the required value type.

TIP

Please note that once you have created a station, you can no longer modify its **Identifier**. The maximum length of a station identifier is 12 characters.

The **Access privilege** attribute allows you to restrict access to the station by selecting the access privilege an operator must have to be permitted to log in at the station. Not selecting an access privilege means that access to the station is free for all operators who have the access privileges required for working with PharmaSuite for Production Execution.

The **Work center** attribute is always read-only.
To assign a station to a work center, open the details view of the respective work center and drag the station's card or tile onto the work center's **Station** tab (page 277). A station can only be assigned to one work center at any given time. So, for changing a station's work center assignment you first need to remove the station from the list of stations on the **Station** tab of its current work center before you can assign it to another work center.

During execution with PharmaSuite for Production Execution the system supports device registration at a station by barcode scanning. To create a barcode label for a station, proceed as follows:

1. Type the barcode string in the **Barcode** attribute cell.
   By default, the barcode is preset with a string of characters that consists of a system-configured station barcode prefix and the station's identifier you have specified as first step of the creation process. The system-configured prefix may be empty so that the barcode and the station’s identifier are identical the identifier of the station.

   TIP
   Do not type leading or trailing blanks before or after the barcode string, since this may interfere with the system-internal formatting conversion and render the barcode unreadable for scanners.

2. Save the station.
   The system enables the **Print barcode label** button on the **File** toolbar as well as the respective function in the **File** menu.
3. Click the **Print barcode label** button or select the respective menu function to print the barcode on a connected printer, typically the default printer defined in your Windows operating system.

CHANGE HISTORY

The **Change History** tab provides a list of all changes that have been performed on the station, which also includes indirect assignment changes that occur when the station has been removed from the list of stations on the **Station** tab of its work center.
For each change event that affected the station, the table displays one row.

To filter the change event entries by date, select the date range you wish to consider. The system only displays the entries that have been registered between the two dates you have selected.
The date range starts at 12:00:00 AM on the date you have selected and ends at 23:59:59 PM on the date you have selected.
When you open the change history, the date range is preset to start with the date when the last change event occurred and to end with the current date. If the system has no change events registered for a station, the date range is preset to the current date.
To refresh the list of change events, click the **Refresh** button (page 75). All change events involving the station that have occurred since you opened the change history are then included.

The following table columns are available and are filled if relevant to the respective change event:

- Timestamp
  indicates the date and time when the change event occurred.
- Action
  indicates the change event that triggered the entry in the change history.
- Affected object
  indicates the object that is affected by the change event. So, when you update the basic data of your station, the change history shows the identifier of the attribute you have modified and when you assign the station to a work center, the change history shows the identifier of the work center.
- Old content/New content
  for data change events on your station, displays the changed values.
- Logged-in user
  login name of the user under whose login the change event occurred.

Index

A

About PharmaSuite • 8
Action (keyboard operation) • 80
Asset attribute (Basic Search) • 221
  Equipment class • 221
  Equipment entity • 236
Asset attribute (Smart Search) • 156
  Equipment class • 157
  Equipment entity • 172
Assigning a class to an entity • 100
Assigning an entity to a class • 100
Assigning property types to equipment • 100
Assignment (equipment - Basic Search) • 221
  Child entity to parent entity • 245
  Class to entity • 244
  Class to template entity • 233
  Entity to class • 225
  Entity to entity group • 245
  Graph to class • 224
  Graph to entity • 243
  Graph to template entity • 233
  Material to template entity • 233
  Property type to graph • 253
Assignment (equipment - Smart Search) • 156
  Child entity to parent entity • 181
  Class to entity • 181
  Class to template entity • 169
  Entity to class • 160
  Entity to entity group • 181
  Graph to class • 160
  Graph to entity • 180
  Graph to template entity • 169
  Material to template entity • 169
  Property type to graph • 191
Assignment (work center) • 277
  Station to work center • 277
Attribute (equipment - Basic Search)
  Automatic status change after expiry • 236
  Automation (equipment entity) • 236
  Automation (template equipment entity) • 228
  Basic (equipment class) • 221
  Basic (equipment entity) • 236
  Basic (equipment graph) • 251
  Basic (property type) • 262
  BigDecimal data type (property type) • 267
  Boolean data type (property type) • 267
  Data type (property type) • 267
  Duration data type (property type) • 268
  Equipment • 221
  EquipmentType data type (property type) • 268
  FlexibleAttributeDefinition data type (property type) • 268
  FlexibleStateModel data type (property type) • 268
  FlexibleTagDefinition data type (property type) • 269
  Historian (equipment entity) • 236
  Historian (template equipment entity) • 228
  Label layout (equipment entity) • 236
  Label layout (template equipment entity) • 228
  MeasuredValue data type (property type) • 269
  Purpose (graph) • 251
  Purpose (property type) • 266
  Ranges data type (property type) • 269
  RoomCleaningRules data type (property type) • 269
  ScaleConfiguration data type (property type) • 269
  ScaleTestAndCalibration data type (property type) • 269
  Short description (scale) • 236
  Status/trigger (equipment graph) • 253
  String data type (property type) • 269
  Style (equipment class) • 225
  Transition (equipment graph) • 255
  Usage (property type) • 266
  WorkCenterAssignment data type (property type) • 270
Attribute (equipment - Smart Search)
  Automatic status change after expiry • 172

Automation (equipment entity) • 172
Automation (template equipment entity) • 164
Basic (equipment class) • 157
Basic (equipment entity) • 172
Basic (equipment graph) • 189
Basic (property type) • 199
BigDecimal data type (property type) • 204
Boolean data type (property type) • 205
Data type (property type) • 204
Duration data type (property type) • 205
Equipment • 156
EquipmentType data type (property type) • 205
FlexibleAttributeDefinition data type (property type) • 205
FlexibleStateModel data type (property type) • 206
FlexibleTagDefinition data type (property type) • 206
Historian (equipment entity) • 172
Historian (template equipment entity) • 164
Label layout (equipment entity) • 172
Label layout (template equipment entity) • 164
MeasuredValue data type (property type) • 206
Purpose (graph) • 189
Purpose (property type) • 203
Ranges data type (property type) • 206
RoomCleaningRules data type (property type) • 206
ScaleConfiguration data type (property type) • 206
ScaleTestAndCalibration data type (property type) • 207
Short description (scale) • 172
Status/trigger (equipment graph) • 191
String data type (property type) • 207
Style (equipment class) • 161
Style (equipment graph) • 197
Transition (equipment graph) • 193
Usage (property type) • 203
WorkCenterAssignment data type (property type) • 207

Attribute (scale configuration)
Calibration with adjustment • 63
Calibration without adjustment • 63
COM port • 61
Disable scale keyboard • 63
Get serial number • 63
Manual scale • 61
Manual tare • 62
Range • 60
Resolution factor • 62
Scale driver • 61
Scale keyboard lockable • 63
Send nominal value • 63
Tare • 62
TCP server/port • 61
Terminal number • 61
Weighing • 62
Zeroing • 62

Attribute (work center)
Access privilege • 279
Basic (station) • 279
Basic (work center) • 277
Storage area • 277
Work center • 277

Automatic status change after expiry (Basic Search) • 236
Automatic status change after expiry (Smart Search) • 172
Automation BigDecimal (Double) editor • 13
Automation BigDecimal (Float) editor • 17
Automation BigDecimal (Integer) editor • 20
Automation Boolean editor • 25
Automation FlexibleTagDefinition editor • 56
Automation Integration server name (Basic Search)
Equipment entity • 236
Template equipment entity • 228
Automation Integration server name (Smart Search)
Equipment entity • 172
Template equipment entity • 164
Automation String editor • 67
Automation tag
Exporting • 107
Importing • 115
Verifying • 118

B

Barcode
Station • 279

Barcode (equipment - Basic Search)

Equipment entity • 236

Template equipment entity • 228

Barcode (equipment - Smart Search)

Equipment entity • 172

Template equipment entity • 164

Basic attribute (equipment - Basic Search) • 221

Equipment class • 221

Equipment entity • 236

Equipment graph • 251

Property type • 199

Template equipment entity • 228

Basic attribute (equipment - Smart Search) • 156

Equipment class • 157

Equipment entity • 172

Equipment graph • 189

Property type • 199

Template equipment entity • 164

Basic attribute (work center)

Station • 279

Work center • 279

Basic operation • 87

Electronic signature • 101

Login • 88

Logout • 90

Password change • 90

Signature • 101

Start • 87

BigDecimal editor • 11

Bracket (Expression editor) • 32

Button (action) • 73

Delete • 73

Display on card • 73

Filter for classes • 73

Filter for entities • 73

Filter for graphs • 73

Filter for property types • 73

Filter for stations • 74

Filter for template entities • 74

Filter for work centers • 74

Move down • 74

Move up • 74

Open cell editor • 75

Print equipment logbook report • 74

Refresh • 75

Remove • 75

Trigger graph transition • 75

Button (toolbar) • 71

Change status • 71

Close • 71

Delete • 71

Help • 71

New equipment class • 71

New equipment entity • 71

New equipment graph • 71

New property type • 72

New station • 72

New template equipment entity • 72

New work center • 72

Print barcode label • 72

Refresh • 72

Save • 72

Zoom in • 73

Zoom out • 73

C

Change history (equipment - Basic Search)

Equipment class • 225

Equipment entity • 249

Equipment graph • 260

Template equipment entity • 234

Change history (equipment - Smart Search)

Equipment class • 161

Equipment entity • 186

Equipment graph • 197

Template equipment entity • 170

Change history (work center)

Station • 280

Work center • 278

Change password • 90

Change station • 88

Change status (toolbar button) • 71

CleaningRules editor • 26
Close (toolbar button) • 71
Comment (Expression editor) • 32
Container equipment entity (definition) • 123
Context (Basic Search) • 246
Context (Smart Search) • 183
Conventions (typographical) • 2
Cursor • 77
  Drop target allowed • 77
  Drop target unallowed • 77
  Select • 77
  Wait • 77

D

Data type (Expression editor) • 37
  BigDecimal • 37
  Boolean • 37
  Duration • 38
  Float • 37
  Long • 37
  MeasuredValue • 39
  String • 38
  Timestamp • 38
Data type (property type - Basic Search) • 267
  BigDecimal • 267
  Boolean • 267
  Duration • 268
  EquipmentType • 268
  FlexibleAttributeDefinition • 268
  FlexibleStateModel • 268
  FlexibleTagDefinition • 269
  MeasuredValue • 269
  Ranges • 269
  RoomCleaningRules • 269
  ScaleConfiguration • 269
  ScaleTestAndCalibration • 269
  String • 269
  WorkCenterAssignment • 270
Data type (property type - Smart Search) • 204
  BigDecimal • 204
  Boolean • 205
  Duration • 205
  EquipmentType • 205
  FlexibleAttributeDefinition • 205
  FlexibleStateModel • 206
  FlexibleTagDefinition • 206
  MeasuredValue • 206
  Ranges • 206
  RoomCleaningRules • 206
  ScaleConfiguration • 206
  ScaleTestAndCalibration • 207
  String • 207
  WorkCenterAssignment • 207
Date/time picker editor • 27
Default object (Basic Search) • 121
  Class • 221
  Graph • 251
  Property type • 262
Default object (Smart Search) • 121
  Class • 157
  Graph • 189
  Property type • 199
Definition
  Equipment class • 121
  Equipment entity • 123
  Equipment graph • 126
  Equipment requirement • 121
  Property type • 126
  Station • 271
  Template equipment entity • 122
  Work center • 271
Delete (action button) • 73
Delete (toolbar button) • 71
Details window (equipment - Basic Search)
  Equipment • 221
  Equipment class • 221
  Equipment entity • 236
  Property type • 262
  Template equipment entity • 228
Details window (equipment - Smart Search)
  Equipment • 156
  Equipment class • 156

Equipment entity • 172
Property type • 199
Template equipment entity • 163
Details window (work center)
Station • 279
Work center • 277
Display on card (action button) • 73
Documentation • 87
Double signature • 101
Drop target allowed (cursor) • 77
Drop target unallowed (cursor) • 77
Duration editor • 29

E

Editing data • 100
Editor • 10
Automation BigDecimal (Double) • 13
Automation BigDecimal (Float) • 17
Automation BigDecimal (Integer) • 20
Automation Boolean • 25
Automation FlexibleTagDefinition • 56
Automation String • 67
BigDecimal • 11
CleaningRules • 26
Date/time picker • 27
Duration • 29
Expression • 29
FlexibleAttributeDefinition • 53
FlexibleStateModel • 55
GraphStatusChange • 58
Historian String • 69
Label layout selection • 58
MeasuredValue • 59
Multi-line text • 59
Ranges • 60
ScaleConfiguraton • 61
ScaleTestAndCalibration • 65
Searchable option list • 67
Tag Browser • 70
Engineering property (Basic Search) • 221
Equipment class • 224
Equipment entity • 243
Template equipment entity • 232
Engineering property (Smart Search) • 156
Equipment class • 159
Equipment entity • 179
Template equipment entity • 168
Equipment class (Basic Search) • 221
Assigning graphs • 224
Default class • 221
Equipment class (definition) • 121
Compiling usage list • 141
Exporting • 103
Importing • 110
Performing mass change operations on (template) entities • 133
Equipment class (Smart Search) • 156
Assigning graphs • 160
Default class • 157
Equipment entity (Basic Search) • 236
Assigning graphs • 243
Barcode printing • 236
Entity grouping • 245
Ungroup entity for repair • 245
Equipment entity (definition) • 123
Container • 123
Exporting • 103
Importing • 110
Room • 123
Scale • 123
Template • 122
Triggering a graph transition • 139
Equipment entity (Smart Search) • 172
Assigning graphs • 180
Barcode printing • 172
Entity grouping • 181
Ungroup entity for repair • 181
Equipment graph (Basic Search) • 251
Assigning property types • 253
Default graph • 251
Handling expiry dates • 255
Equipment graph (definition) • 126

Equipment graph (Smart Search) • 188
- Assigning property types • 191
- Default graph • 189
- Handling expiry dates • 193

Equipment group (Basic Search) • 245

Equipment group (Smart Search) • 181

Equipment logbook report (Basic Search) • 247

Equipment logbook report (Smart Search) • 183

Equipment requirement (definition - Smart Search) • 121

Exporting Automation tags • 107

Exporting data objects • 103

Exporting Historian points • 107

Expression editor • 29
- Aggregation functions • 52
- Analysis functions • 49
- Arithmetic operators • 43
- Bracket • 32
- Comment • 32
- Comparison operators • 45
- Constant • 37
- Conversion functions • 50
- Data type • 37
- Expression editor (keyboard operation) • 82
- Function • 48
- Keyboard operation • 82
- Logical operators • 46
- Null handling functions • 48
- Operator • 42
- Reference • 32
- Section • 32
- Transition action • 31
- Transition condition • 30

Expression editor icon • 78
- Attribute • 78
- Command • 78
- Constant • 78
- Error • 78
- Function • 78
- List option • 78
- Operator • 78
- Reference • 78
- Separator • 78

F

File menu
- Equipment - Basic Search • 208
- Equipment - Smart Search • 143
- Work center • 272

File toolbar
- Equipment - Basic Search • 220
- Equipment - Smart Search • 155
- Work center • 276

Filter for classes (action button) • 73

Filter for entities (action button) • 73

Filter for graphs (action button) • 73

Filter for property types (action button) • 73

Filter for stations (action button) • 74

Filter for template entities (action button) • 74

Filter for work centers (action button) • 74

FlexibleAttributeDefinition editor • 53

FlexibleStateModel editor • 55

Forced unbinding of equipment entity (Basic Search) • 236

Forced unbinding of equipment entity (Smart Search) • 172

Function (Expression editor) • 48
- Average (aggregation) • 52
- Contains Item (analysis) • 49
- Convert to BigDecimal (conversion) • 50
- Convert to Long (conversion) • 51
- Convert to MeasuredValue (conversion) • 51
- Convert to unitless number (conversion) • 50
- Current date and time (analysis) • 49
- Is undefined (null handling) • 49
- Maximum (aggregation) • 52
- Minimum (aggregation) • 53
- Undefined - one of many (null handling) • 49
- Undefined - one of two (null handling) • 48

G

Generated equipment entity (definition) • 122

Graphical element • 71
- Button (action) • 73
- Button (toolbar) • 71
- Cursor • 77

Expression editor • 78

Marker • 76

GraphStatusChange editor • 58

Grouping entities (Basic Search) • 245

Grouping entities (Smart Search) • 181

H

Help (toolbar button) • 71

Help access • 7

Help menu

Equipment - Basic Search • 213

Equipment - Smart Search • 148

Work center • 274

Help toolbar

Equipment - Basic Search • 220

Equipment - Smart Search • 155

Work center • 276

Help window • 7

Historian access server (Basic Search)

Equipment entity • 236

Template equipment entity • 228

Historian access server (Smart Search)

Equipment entity • 172

Template equipment entity • 164

Historian point

Exporting • 107

Importing • 115

Verifying • 118

Historian server (Basic Search)

Equipment entity • 236

Template equipment entity • 228

Historian server (Smart Search)

Equipment entity • 172

Template equipment entity • 164

Historian String editor • 69

I

Importing Automation tags • 115

Importing data objects • 110

Importing Historian points • 115

K

Keyboard operation • 79

Action • 80

Expression editor • 82

Framework • 79

L

Label layout (Basic Search)

Equipment entity • 236

Template equipment entity • 228

Label layout (Smart Search)

Equipment entity • 172

Template equipment entity • 164

Label layout selection editor • 58

Live Data Area path (Basic Search)

Equipment entity • 236

Template equipment entity • 228

Live Data Area path (Smart Search)

Equipment entity • 172

Template equipment entity • 164

Logbook (equipment - Basic Search)

Report printing • 247

Viewing • 247

Logbook (equipment - Smart Search)

Report printing • 183

Viewing • 183

Login • 88

Change password • 90

Change station • 88

Password • 88

Logout • 90

M

Marker icon • 76

Automation usage type • 77

Class • 76

Displayed on card • 76

Entity • 76

Error • 76

Graph • 76

Graph object type • 77

Historian usage type • 77
Property type • 76
Runtime usage type • 77
Specification usage type • 77
Station • 76
Template entity • 76
Work center • 77
Mass change operations on (template) entities • 133
MeasuredValue editor • 59
Menu (equipment - Basic Search) • 207
File • 208
Help • 213
View • 211
Window • 212
Menu (equipment - Smart Search) • 143
File • 143
Help • 148
View • 147
Window • 147
Menu (work center) • 271
File • 272
Help • 274
View • 273
Window • 273
Migrated history (Basic Search) • 249
Migrated history (Smart Search) • 186
Mismatch message • 6
Move down (action button) • 74
Move up (action button) • 74
Multi-line text editor • 59

N

New equipment class (toolbar button) • 71
New equipment entity (toolbar button) • 71
New equipment graph (toolbar button) • 71
New property type (toolbar button) • 72
New station (toolbar button) • 72
New template equipment entity (toolbar button) • 72
New work center (toolbar button) • 72

O

Open cell editor (action button) • 75
Operator (Expression editor) • 42
Conjunction (logical) • 46
Disjunction (logical) • 47
Divide (arithmetic) • 45
Equal to (comparison) • 45
Greater than (comparison) • 46
Greater than or equal to (comparison) • 46
Less than (comparison) • 46
Less than or equal to (comparison) • 46
Minus (arithmetic) • 43
Multiply (arithmetic) • 44
Negation (logical) • 48
Not equal to (comparison) • 45
Plus (arithmetic) • 43

P

Panel management • 4
Password • 90
Password change • 90
Print barcode label (toolbar button) • 72
Print equipment logbook report (action button) • 74
Process property (Basic Search) • 221
Equipment class • 223
Equipment entity • 242
Template equipment entity • 232
Process property (Smart Search) • 156
Equipment class • 159
Equipment entity • 178
Template equipment entity • 168
Property (Basic Search) • 221
Property (Smart Search) • 156
Property tab toolbar (Basic Search) • 220
Property tab toolbar (Smart Search) • 156
Property type (Basic Search) • 262
Default property type • 262
Property type (definition) • 126
Exporting • 103
Importing • 110
Property type (Smart Search) • 199
Default property type • 199
Purpose (Basic Search)

- Graph • 251
- Property type • 266
- Purpose (Smart Search)
  - Graph • 189
  - Property type • 203

R

- Ranges editor • 60
- Reference (Expression editor) • 32
- Refresh (action button) • 75
  - Refresh • 75
- Refresh (toolbar button) • 72
- Remove (action button) • 75
- Room equipment entity (definition) • 123

S

- Save (toolbar button) • 72
- Scale configuration • 61
  - Calibration with adjustment • 63
  - Calibration without adjustment • 63
  - COM port • 61
  - Disable scale keyboard • 63
  - Get serial number • 63
  - Manual scale • 61
  - Manual tare • 62
  - Resolution factor • 62
  - Scale driver • 61
  - Scale keyboard lockable • 63
  - Set nominal value • 63
  - Tare • 62
  - TCP server/port • 61
  - Terminal number • 61
  - Weighing • 62
  - Zeroing • 62
- Scale equipment entity (definition) • 123
- ScaleConfiguration editor • 61
- ScaleTestAndCalibration editor • 65
- Screen layout • 3
  - Details window • 100
  - Panel management • 4
  - Search criteria panel • 92
  - Search results panel • 92
- Screen layout (equipment - Basic Search)
  - Menu • 207
  - Toolbar • 219
- Screen layout (equipment - Smart Search)
  - Menu • 143
  - Toolbar • 155
- Screen layout (work center)
  - Menu • 271
  - Toolbar • 276
- Searchable option list editor • 67
- Searching data • 92
- Section (Expression editor) • 32
- Select (cursor) • 77
- Signature • 101
- Single signature • 101
- Specification property (Basic Search) • 221
  - Equipment class • 223
  - Equipment entity • 239
  - Template equipment entity • 231
- Specification property (Smart Search) • 156
  - Equipment class • 158
  - Equipment entity • 176
  - Template equipment entity • 166
- Start PharmaSuite • 87
- Station (definition) • 271
  - Barcode printing • 279
- Status handling • 126
  - Changing status of equipment • 130
  - Equipment class • 127
  - Equipment entity • 129
  - Equipment graph • 130
  - Template equipment entity • 128
- Status handling (Basic Search)
  - Status history of equipment class • 227
  - Status history of equipment entity • 250
  - Status history of equipment graph • 261
  - Status history of template equipment entity • 235
- Status handling (Smart Search)
  - Status history of equipment class • 162
  - Status history of equipment entity • 187

Status history of equipment graph • 198

Status history of template equipment entity • 171

Status history (Basic Search)

Equipment class • 227

Equipment entity • 250

Equipment graph • 261

Template equipment entity • 235

Status history (Smart Search)

Equipment class • 162

Equipment entity • 187

Equipment graph • 198

Template equipment entity • 171

Status/trigger attribute (equipment graph - Basic Search) • 253

Status/trigger attribute (equipment graph - Smart Search) • 191

Style (equipment graph - Basic Search) • 260

Style attribute (Basic Search) • 221

Equipment class • 161

Equipment graph • 197

Style attribute (Smart Search) • 156

Equipment class • 161

Equipment graph • 197

T

Tag Browser editor • 70

Template equipment entity (Basic Search) • 228

Assigning graphs • 233

Template equipment entity (definition) • 122

Template equipment entity (Smart Search) • 163

Assigning graphs (Smart Search) • 169

Test weight (scale test and calibration) • 65

Toolbar (equipment - Basic Search) • 219

File toolbar • 220

Help toolbar • 220

Property tab toolbar • 220

View toolbar • 220

Toolbar (equipment - Smart Search) • 155

File toolbar • 155

Help toolbar • 155

Property tab toolbar • 156

View toolbar • 155

Toolbar (work center) • 276

File toolbar • 276

Help toolbar • 276

View toolbar • 276

Transition (equipment graph)

Equipment - Basic Search • 255

Equipment - Smart Search • 193

Transition action (Expression editor) • 31

Transition condition (Expression editor) • 30

Trigger graph transition (action) • 75

Triggering a graph transition on an entity • 139

U

Unbind equipment entity (Basic Search) • 236

Unbind equipment entity (Smart Search) • 172

Updating multiple (template) entities • 133

Usage list of class • 141

V

Verifying Automation tags • 118

Verifying Historian points • 118

View menu

Equipment - Basic Search • 211

Equipment - Smart Search • 147

Work center • 273

View toolbar

Equipment - Basic Search • 220

Equipment - Smart Search • 155

Work center • 276

W

Wait (cursor) • 77

Warehouse Management communication • 123

Load/unload trigger (Basic Search) • 253

Load/unload trigger (Smart Search) • 191

Window menu

Equipment - Basic Search • 212

Equipment - Smart Search • 147

Work center • 273

Work center (definition) • 271

Z

Zoom in (toolbar button) • 73

Zoom out (toolbar button) • 73