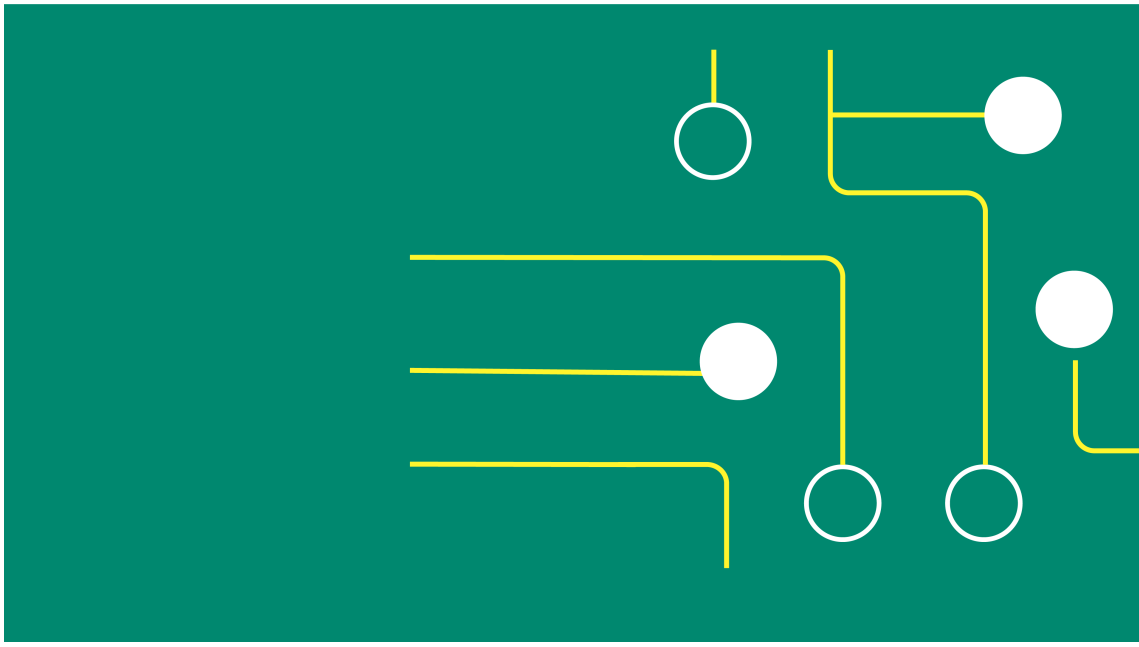
cytiva

UNICORN™ 7.10

Method Manual



cytiva.com

Table of Contents

1 Introducing the UNICORN Method Editor

Introduction

This chapter contains:

- A general introduction to creating methods using the UNICORN™ system control software.
- Information about the user documentation for UNICORN, including an overview of related documents describing the use of the software.

In this chapter

Software declaration of conformity

UNICORN 7.10 is technically compatible with all relevant sections of FDA 21 CFR Part 11.

A part 11-system assessment checklist is available on request from your local Cytiva representative.

1.1 About the UNICORN Method Editor

Introduction

This section is a brief introduction to creating methods in UNICORN and a description of the scope of this manual.

What is UNICORN?

UNICORN is a complete software package for:

- control and supervision of chromatography systems.
- evaluation and analysis of the results from separation runs.

Workflow

The workflow in UNICORN can be divided into four distinct stages. The flow chart below shows the workflow stages.

Create a method

A method in UNICORN is a user-defined set of instructions that can be used to run an entire process on a system, for example a purification run or a column performance test. A method is composed of one or several predefined or user defined phases which are reusable sets of instructions. Examples of predefined phases are equilibration and eluation phases. An empty user defined phase is also available.

The UNICORN ***Method Editor*** module is a comprehensive tool for creating or editing methods either by using predefined methods and phases, wizard generated methods or user defined text edited methods. Depending on the system, the Method Editor can for example be used to:

- build a method from a library of phases.
- build a method with guidance from a method wizard.
- create custom phases.
- create method queues to run multiple methods on up to three separate systems.
- keep track of Column types or columns using the ***Column Handling*** tool.

- design and optimize purification schemes using the ***Design of Experiments*** and ***Scouting*** tools.
- automatically mix and titrate buffers using the ***BufferPro*** tool.

1.2 About this manual

Introduction

This section describes the purpose of the manual, the general structure and conventions applied in the text, and some prerequisites that should be fulfilled before you start to apply any of the procedures described in the following chapters.

The purpose of the UNICORN Method Manual

The purpose of the UNICORN Method Manual is to provide a comprehensive guide to creating methods that can be run on an ÄKTA™ system. It covers the features and tools included in the Method Editor module of the UNICORN software with practical instructions. Some functionality is only available for some systems.

The manual covers the following:

- how to create methods and phases.
- how to use ***BufferPro***.
- how to design and optimize experiments using ***Design of Experiments*** and ***Scouting***.
- how to use method queues.
- how to handle Column types and columns.
- how to convert and scale methods.

For advanced users, an overview of how to edit methods at the level of individual instructions is also given.

Note: *The Method Manual does not describe the function of every command in all panes and dialog boxes of the user interface. Refer to the online help for information about commands that are not described in this manual. The online help in the* ***Method Editor*** *module is accessed either by clicking Help buttons in software dialog boxes, by pressing the* **F1** *key, or selecting* ***Help →Help for Method Editor****.*

Document structure

Each chapter starts with a brief overview that presents the contents and the headings for the sections that the chapter contains. Most sections begin with an introduction that summarizes the content. Some sections are divided into sub-sections, each with an overview of the contents.

A section is divided into blocks of information with separating lines. The blocks are identified by a label extending into the margin (such as the label Document structure above). This makes it easier for you to quickly scan a page to find the exact topic you are looking for.

Typographical conventions

Menu commands, field names and other text items from the software are quoted exactly as they appear on the screen, in a bold italic typeface:

Example: ***Result Navigator***, ***Method Navigator***, ***Method Navigator***, ***UNICORN User Setup*** etc.

Menu paths are shown in a bold italic typeface with a separating colon between each level:

Tools →***UNICORN User Setup*** i.e., the menu option ***UNICORN User Setup*** from the ***Tools*** menu.

Controls on the instrument, computer or keyboard keys are shown with a bold, regular typeface:

Example: Press the **Delete** key.

Text that the user must either type exactly as shown in the manual, or that UNICORN displays as a response (not a regular part of the graphic user interface), is represented by a monospaced typeface:

Example: `Connection change`

File system paths are represented by a monospaced typeface:

Example: `C:\Program Files\Cytiva\UNICORN\`

Prerequisites

The following prerequisites must be fulfilled before you can use this manual the way it is intended:

- You need to have a general understanding of how your PC and Windows work. In most cases universal computer functions will not be explained.
- UNICORN must be installed and configured correctly on your computer.
- You need to understand the general concepts of liquid chromatography. Terminology and functionalities will be explained only when they differ from normal practice.

1.3 Associated documentation

Introduction

This section describes the user documentation that is delivered with UNICORN.

User documentation

The user documentation listed in the table below is available from the ***Help*** menu in UNICORN and as printed books.

Document	Main contents
UNICORN Method Manual	Overview and detailed descriptions of the method creation features in UNICORN. Instructions on how to use the software. Workflow descriptions for common operations.
UNICORN Evaluation Manual	
UNICORN Administration and Technical Manual	Overview and detailed description of network setup and complete software installation. Administration of UNICORN and the UNICORN database.
UNICORN System Control Manual	Overview and detailed description of the system control features in UNICORN. Includes general operation, system settings and instructions on how to perform a run.
UNICORN Contextual Help	Dialog box descriptions for UNICORN (from the ***Help*** menu).

2 The UNICORN Method Editor

About this chapter

This chapter gives an introduction to the ***Method Editor*** in UNICORN 7.10. It gives a brief description of the ***Method Editor*** interface and describes the concept of methods in UNICORN 7.10.

For information about how to create, open and edit methods as well as signing methods and importing/exporting methods, see *Chapter 3 Create and edit methods, on page 23*.

In this chapter

2.1 The Method Editor

Introduction

The ***Method Editor*** provides complete facilities for:

- creating and editing methods.
- copying, saving and deleting methods.
- converting methods for use with different system types.

The ***Method Editor*** also provides a number of tools to assist the user in optimizing runs and a tool for handling Column types and columns (see below for more information). Functions like signing methods electronically and importing/exporting methods are also included.

Tools in the Method Editor

The table below describes the different tools included in the ***Method Editor***.

Tool	Description
Design of Experiments (***DoE***) (system specific)	***DoE*** is used to find out, in a systematic way, which run parameters affect a process to be run and how to find optimal values for these parameters to obtain the best possible result using a minimum number of runs. When creating a method and setting up an experimental design using ***DoE***, an optimized ***Scouting scheme*** will automatically be created. See *Chapter 5 Design of Experiments, on page 98* for more information. ***Note:*** ***DoE*** *requires an e-license.*
Scouting	***Scouting*** is used to repeat a series of ***Method runs*** automatically, where the user can change the values of predetermined variables before starting the method. A ***Scouting scheme*** is defined as part of the method. See *Chapter 4 Scouting, on page 86* for more information.

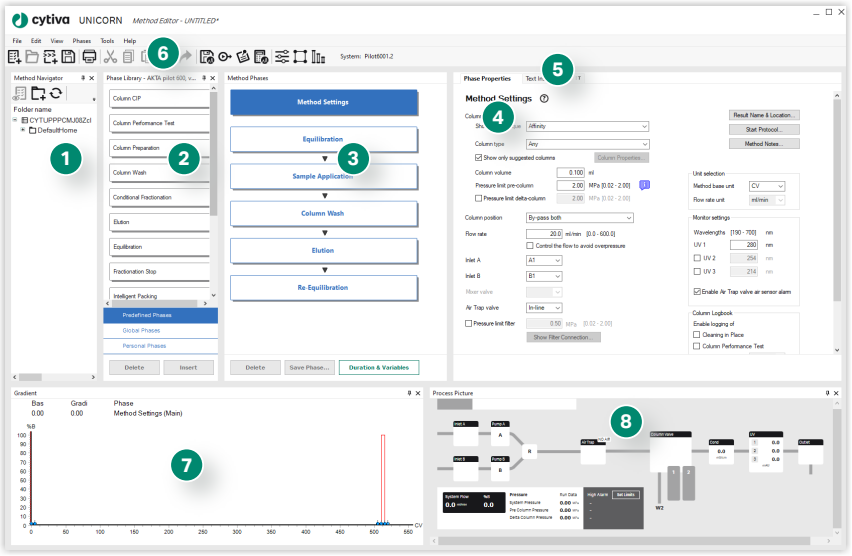
Tool	Description
BufferPro (system specific)	***BufferPro*** allows a buffer of defined pH, and with defined salt concentrations to be prepared from four stock solutions (one Buffer stock solution, one Titrant, Water and a Salt stock solution). pH and salt concentration can be used as variable scouting parameters included in a ***Scouting scheme*** or in a ***Design of Experiments*** (***DoE***). ***BufferPro*** is optimized for cation and anion exchange chromatography, but can also be used when running other chromatographic techniques. See *Chapter 6 BufferPro, on page 176* for more information.
Column Handling	***Column Handling*** enables handling of Column types and columns. See *Chapter 9 Column Handling, on page 210* for more information. ***Note:*** *Parts of* ***Column Handling*** *requires an e-license.*

Illustration of the Method Editor

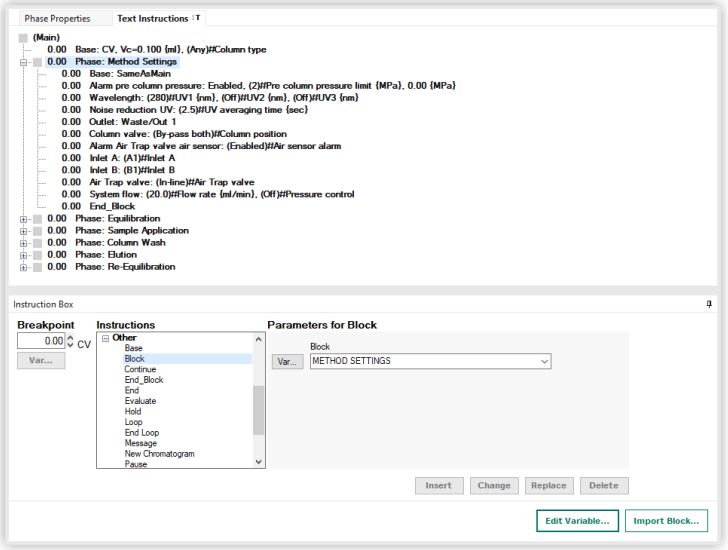
The basic ***Method Editor*** interface consists of two panes, the ***Method outline*** and the ***Phase Properties***/***Text Instructions*** pane.

By default, the ***Toolbar***, ***Phase Library*** pane and ***Gradient*** pane are also displayed in the ***Method Editor***. The display of these panes is however optional. Two more panes may be displayed in the ***Method Editor***, the ***Method Navigator*** and ***Process Picture***.

The illustration below shows the ***Method Editor*** with all the optional panes displayed.



Area	Description
1	***Method Navigator*** (optional pane): Shows all the user folders, methods and method queues that are available in the database.
2	***Phase Library*** (optional pane): Contains all available phases.
3	***Method Outline***: Shows the phases included in the opened method.
4	***Phase Properties*** tab: Click to display the ***Phase Properties***. ***Phase Properties*** shows the settings for the highlighted phase in the ***Method Outline***. For wizard generated and text edited methods, the ***Phase Properties*** shows a list of variables.

Area	Description
5	***Text Instructions*** tab: Click to display the ***Text Instructions***. ***Text Instructions*** shows the method in a text format. The illustration below shows the ***Text Instructions*** tab. 
6	***Toolbar*** (optional pane): Shows the toolbar buttons.
7	***Gradient*** (optional pane): Shows the programmed gradient and break points for included phases and blocks.
8	***Process Picture*** (optional pane): Illustrates the flow path of the instrument graphically.

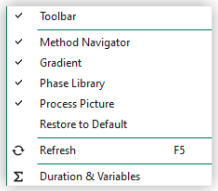
Note: *For detailed information on the toolbar and the different panes in the* ***Method Editor****, see Getting Help on the Toolbar and panes in the Method Editor below.*

Display optional panes

The optional panes in the ***Method Editor*** are displayed by selecting them in the ***View*** menu. To restore the appearance of the ***Method Editor*** to display the default panes, select ***Restore to Default*** in the ***View*** menu. Then, the ***Toolbar***, ***Gradient*** and ***Phase Library*** are displayed. The appearance of the optional panes can also be controlled using the ***Auto Hide*** function (see below for more information).

Note: *Settings made by a user are automatically remembered by the software and are applied next time the same user opens the* ***Method Editor****.*

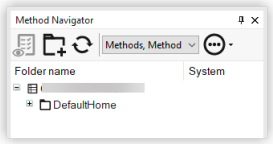
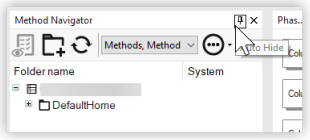
The illustration below shows the ***View*** menu with the default panes selected.



Auto hide optional panes

The optional panes may either be displayed statically in the position where they open, or the ***Auto Hide*** function can be selected to automatically hide/display the pane when moving the mouse pointer over the position of the pane.

The table below describes how to turn on the ***Auto Hide*** function and how to hide/display, in this example, the ***Method Navigator*** pane.

Step	Action
1	If not already displayed, open the ***Method Navigator*** in the ***Method Editor*** by clicking ***Method Navigator*** on the ***View*** menu. *Result:* The ***Method Navigator*** pane is displayed. 
2	To turn on the ***Auto Hide*** function, click the vertical pin symbol in the top right hand corner.  *Result:* The pin symbol is rotated to horizontal position and a tab named ***Method Navigator*** is displayed to the left.
3	Click outside the ***Method Navigator***. *Result:* The ***Method Navigator*** is hidden and only the ***Method Navigator*** tab is displayed.
4	**a.** To display the ***Method Navigator*** again, move the mouse pointer over the ***Method Navigator*** tab.

Step	Action
	b. To turn off the ***Auto Hide*** function, click the horizontal pin symbol in the top right hand corner of the ***Method Navigator*** pane. *Result:* The ***Method Navigator*** pane is displayed statically.

Getting help on the toolbar and panes in the Method Editor

The table below describes how to find detailed information about the toolbar and the different panes in the ***Method Editor*** by opening the Online Help.

Step	Action
1	To display detailed information about the toolbar and different panes in the ***Method Editor*** interface, click ***Help For Method Editor*** on the ***Help*** menu. *Result:* The online help opens displaying the ***Method Editor*** help start page.
2	To display help for a specific pane, click in the pane and press the **F1** keyboard key. *Result:* The online help page describing that pane is opened.

2.2 Methods in UNICORN 7.10

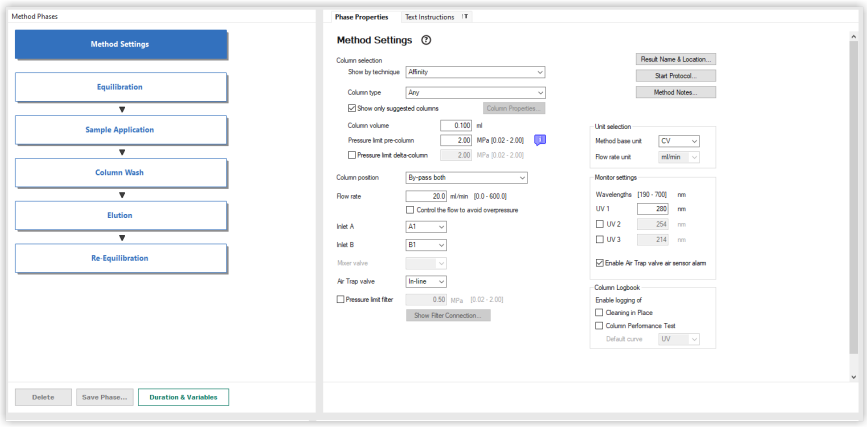
About methods

The program instructions for a chromatography run are defined in a ***Method***. The instructions are specific for each instrument configuration and component set up and follow certain syntactical and hierarchical rules.

Instructions are combined into blocks. Individual instructions and minor blocks are combined into the major method blocks, called ***Phases***. In a predefined method (only available for some systems) each phase reflects a step in the chromatography run, for example, equilibration or sample application. A number of settings are available for each type of phase. By building methods in this way, methods are easily created and edited.

See *Chapter 3 Create and edit methods, on page 23* for information about creating and editing methods in the ***Method Editor***.

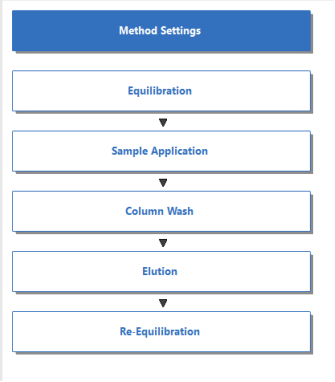
The illustration below shows the phases in a predefined method in the ***Method Outline*** and the corresponding settings for the highlighted phase in the ***Phase Properties*** tab. The image is specific for systems that can use predefined methods.



Method structure

A method always starts with the ***Method Settings*** phase. This phase contains general settings that affect the rest of the method. For example, all systems have the settings ***Column type*** and ***Column volume*** and some systems have ***Flow rate*** and ***Method Base Unit***. If ***Column type*** is changed for a predefined method, UNICORN will automatically calculate correct settings for volume, flow rate, and pressure limits. For wizard generated methods, UNICORN will only check that the pressure and flow limits are not exceeded when saving a method. Subsequent phases reflect steps included in the chromatography run.

The figure below shows a predefined method with the different phases in the ***Method Outline*** in the ***Method Editor***.

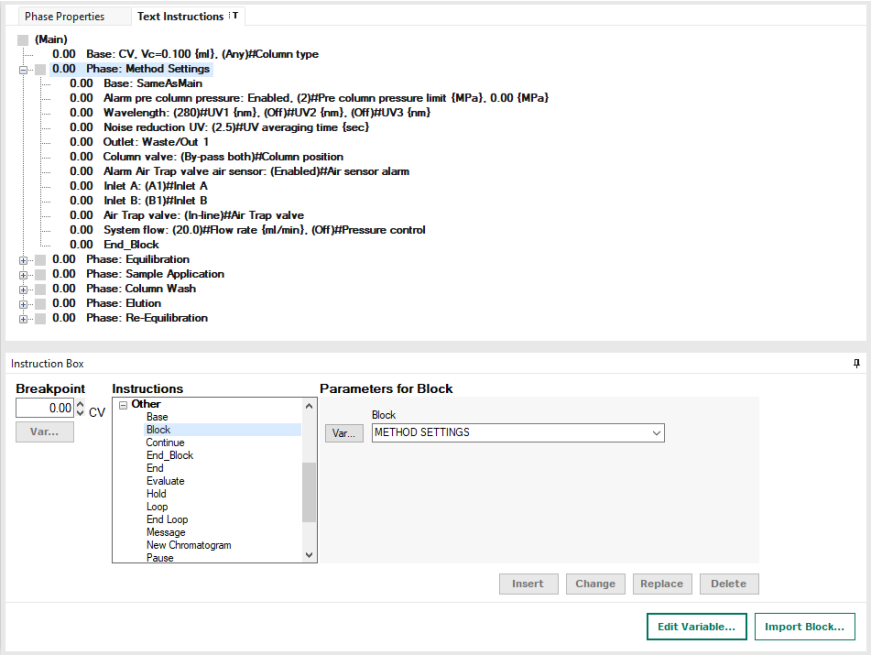


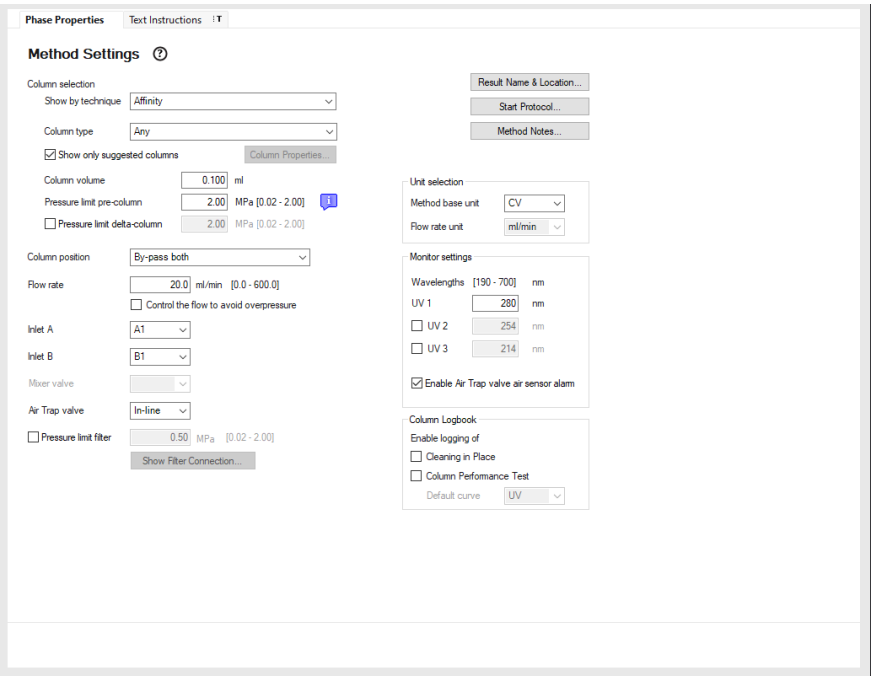
Working with methods

It is recommended to create and edit methods using ***Phase Properties***. Phases can easily be dragged-and-dropped into the ***Method Outline*** from the ***Phase Library*** and the phases are easily rearranged. Settings for each phase are set in the ***Phase Properties*** tab. When working like this, the text method is automatically built up in the ***Text Instructions*** tab and settings for blocks and instructions are updated accordingly.

The illustrations below show the text instructions and the phase properties settings for the ***Method Settings*** phase in a predefined method.

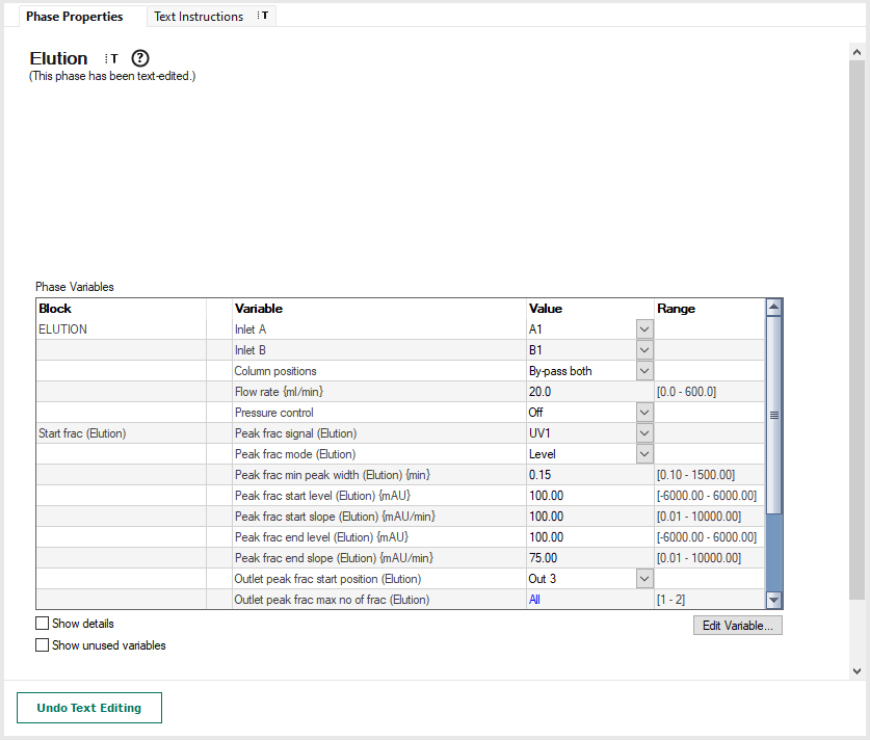
Note: *A wizard generated method consists of a basic method settings phase and a user defined phase that contains the variable list.*





It is possible to use the text editor in ***Text Instructions*** to create a phase from scratch and to edit methods. Instructions are then created or edited one by one. This can be an option for fine-tuning or optimization of a method. If the text editor is used for a predefined phase, ***Phase Properties*** will subsequently only show a list of variables for the phase, as shown in the following illustration. For a predefined phase this can always be restored by clicking on the ***Restore Phase Properties*** button.

Phases that have been edited in the text editor are noted with a blue letter ***T*** as shown in the illustration below.



The phase ***User Defined*** is an empty phase designed for text editing methods. Such phases will only be displayed as a variable list in ***Phase Properties***, and may be saved in the personal or global phase library for reuse in other methods.

See *Chapter 10 Text edit methods, on page 252* for information about text editing methods.

Note: *Do not mix text-edited and non-text-edited phases unless you clearly understand the implications for the entire method of the instructions in the text-edited phases.*

Method types

UNICORN supplies a number of ***Predefined*** methods for different separation techniques and maintenance applications (e.g., preparation and cleaning of the system and columns). The phase ***Method Settings*** is mandatory in all methods.

See *Chapter 3 Create and edit methods, on page 23* for information about how to create new methods.

The table below gives a general description of the different method types.

Method	Description
Predefined	***Predefined*** methods include a number of relevant phases appropriate for the purification or maintenance to be performed. You may use the predefined methods as they are, or with adjusted settings as needed. See Section for descriptions of the ***Predefined*** methods supplied with the software. ***Note:*** *The **Predefined** methods are included in the instrument configuration files for each specific instrument.*
Wizard generated	Wizard generated methods contain a ***Method Settings*** phase and a ***User Defined*** phase with a variable list.
Empty	***Empty*** methods include the mandatory phase ***Method Settings***. Other phases are then added by the user and settings adjusted as needed.

Predefined phases

UNICORN provides a ***User Defined*** phase which is available for all systems, and a number of other ***Predefined Phases*** which are available for some systems.

Predefined phases (for example ***Equilibration*** and ***Column CIP***) can be used when building or editing methods in the ***Method Editor***. A predefined phase contains all necessary instructions to be run, except ***Method Settings*** which is mandatory in all methods. The ***User Defined*** phase is an empty phase that can be built by adding text instructions in the text editing field.

See for descriptions of the predefined phases supplied with the software. See also *Chapter 3 Create and edit methods, on page 23*.

3 Create and edit methods

About this chapter

This chapter describes how to create, edit and handle chromatography and maintenance methods in UNICORN 7.10 using the ***Phase Properties*** tab. It also describes overall method options, how to sign methods electronically, how to print methods, how to convert and scale methods from one ÄKTA system type to another, and how to import/export methods. Descriptions of the predefined methods and phases supplied with the software are also included.

Note: *It is recommended to work with phases using the* ***Phase Properties*** *tab. This chapter does not cover how to edit methods using the* ***Text Instructions*** *tab. For information about text editing methods, see* *Chapter 10 Text edit methods, on page 252**.*

In this chapter

3.1 Working with methods - Overview

Introduction

In UNICORN 7.10 the predefined methods are built up using phases, where each phase corresponds to a step in a chromatography run with a number of properties associated with that phase. The wizard generated methods consist of a method settings phase and a user defined phase containing all instructions for the method. See *Section 2.2 Methods in UNICORN 7.10, on page 18* for more information about method structure, definitions and concepts of methods in UNICORN 7.10.

There are three different ways of creating and editing methods in UNICORN 7.10:

- Creating and editing methods using phases and the phase properties settings in the ***Phase Properties*** tab.
- Creating methods using the wizard.
- Creating methods by text editing, creating and editing text instructions one-by-one.

Main steps when defining a new method using phases

The main steps when defining a method are:

Stage	Description
1	Create/open a method • Create a ***Predefined*** method (including a set of phases that may be edited) or • Open an existing method that can be edited and saved with a new name or overwritten
2	Build/edit the ***Method Outline*** and/or edit the ***Phase Properties*** for the appropriate phases • ***Predefined*** methods: use as they are, or edit the ***Method Outline*** and/or ***Phase Properties*** • Opened methods: edit the ***Method Outline*** and/or ***Phase Properties***
3	Save the method

Main steps when defining a new method using the wizard

The main steps when defining a method are:

Stage	Description
1	Create/open a method • Create a method using the method wizard or • Open an existing method that can be edited and saved with a new name or overwritten
2	Build/edit the ***Method Outline*** and/or edit the ***Phase Properties*** for the appropriate phases • Wizard generated methods: use as they are, or edit the ***Phase variable list*** in the user defined phase • Opened methods: edit the ***Method Outline*** and/or ***Phase variable list***
3	Save the method

Main steps when defining an empty method

The main steps when defining an empty method are:

Stage	Description
1	Create/open a method • Create a new ***Empty*** method containing the ***Method Settings*** phase.
2	Build/edit the ***Method Outline*** and edit the ***Text instructions*** for the phases • Add user defined or predefined phases to the method (i.e., build the ***Method Outline***) and edit the phases as appropriate.
3	Save the method

Main steps when editing a method

The main steps when editing a method are:

Stage	Description
1	Open the method to be edited

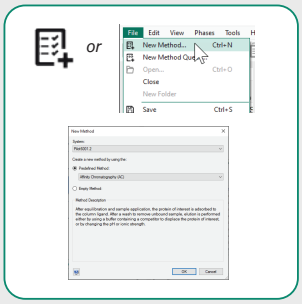
Stage	Description
2	• Edit the ***Method Outline*** and/or • Edit the ***Phase Properties*** for the appropriate phases or the phase variables for a wizard generated or text created method. and/or • Text edit user defined phases
3	Save the method

Illustration of workflow when creating or editing a new method

The illustration below shows the workflow in the ***Method Editor*** when creating or editing a method. The available options depend on the instrument configuration.

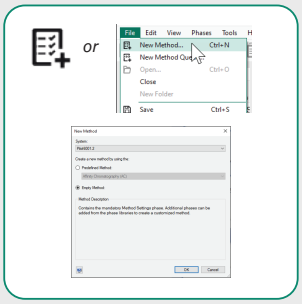
1. Create method

Create new method using phases



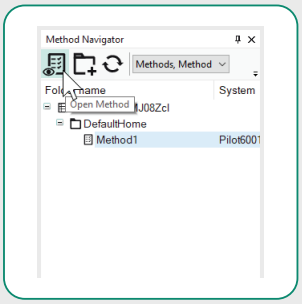
or

Create new method using the wizrd



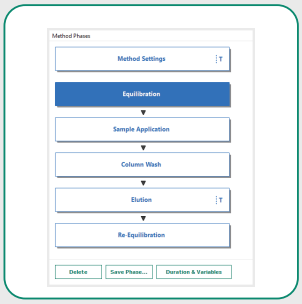
or

Create new method using the wizrd



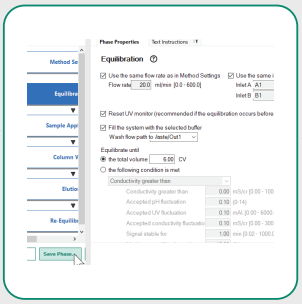
2a. Edit method using phases

Build/edit method outline



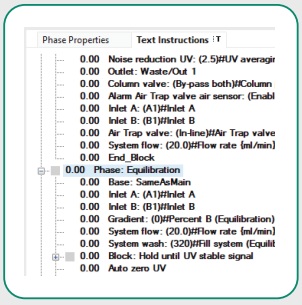
and/ or

Edit Phase Properties



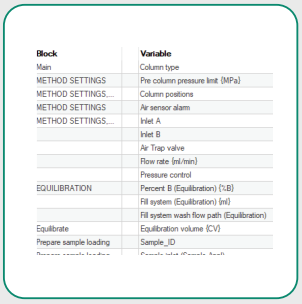
and/ or

Text edit instructions



2b. Edit wizard created methods

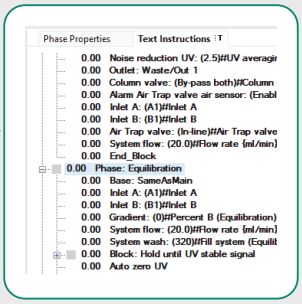
Edit Phase Variables



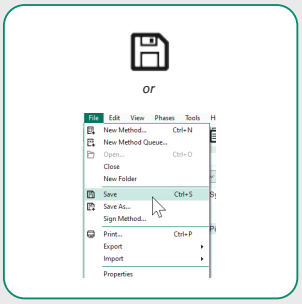
Block	Variable
Main	Column type
METHOD SETTINGS	Pre column pressure limit (MPa)
METHOD SETTINGS,...	Column positions
METHOD SETTINGS	Air sensor alarm
METHOD SETTINGS,...	Inlet A
	Inlet B
	Air Trap valve
	Flow rate (ml/min)
	Pressure control
EQUILIBRATION	Percent B (Equilibration) (%B)
	Fill system (Equilibration) (ml)
	Fill system wash flow path (Equilibration)
Equilibrate	Equilibration volume (CV)
Prepare sample loading	Sample_ID

and/ or

Text edit instructions



3. Save method



Overall method options

In addition to creating, editing and saving the method in the ***Method Editor***, a number of more general method options are available. These are settings for the method and are saved with the method.

Overall method settings can be divided into two groups. The following table shows the different groups.

Method option	Description
General method options	• setting result name and the location of the results • setting up start protocols • adding/editing notes to the method • choosing to include evaluation procedures to be performed after the run • viewing and printing an estimate of the method duration time and the variables in the method See *Section 3.6.2 Set general method options for the method, on page 56* for more information.
Method options intended to assist the user in optimizing runs in UNICORN	• ***Scouting*** See *Chapter 4 Scouting, on page 86* for information. • ***Design of Experiments*** (***DoE***) See *Chapter 5 Design of Experiments, on page 98* for information. • ***BufferPro*** See *Chapter 6 BufferPro, on page 176* for information.

3.2 Open a method

Follow the instructions to open an existing method in the database:

Step	Action
1	In the ***Method Editor***: • Click the ***Open Method Navigator*** button in the toolbar  or • click ***Open*** on the ***File*** menu or • click ***Method Navigator*** on the ***View*** menu *Result:* The ***Method Navigator*** is displayed.
2	Select the method to be opened in the ***Folder name*** column.
3	To open the method, • Click the ***Open*** button located in the toolbar of the ***Method Navigator*** pane  or • double-click the selected method or • right-click on the method name and click ***Open*** *Result:* The method is opened and displayed in the ***Method Outline*** pane with included phases. You can continue to edit the phases of the method using ***Phase Properties***.

3.3 Working with predefined method

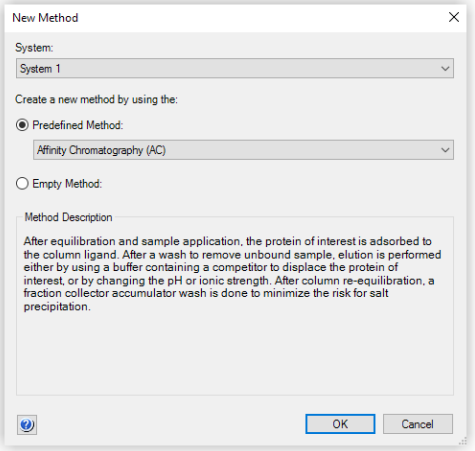
About this section

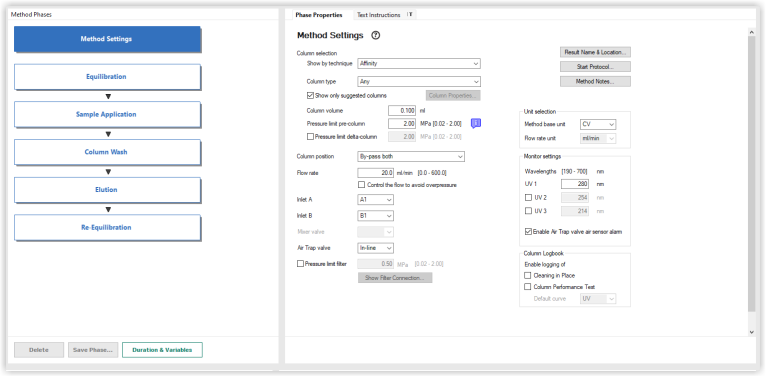
This section describes how to work with methods and phases in systems that have access to the full phase library and predefined methods.

In this section

3.3.1 Create a predefined method

Follow the instructions to create a new method using phases:

Step	Action
1	In the ***Method Editor***: • click the ***Create a new method*** button in the toolbar  or • click ***New Method*** on the ***File*** menu *Result:* The ***New Method*** dialog opens. 
2	In the ***New Method*** dialog box: **a.** select a system in the ***System*** drop-down list **b.** click ***Predefined Method*** and select a method in the drop-down list **c.** click ***OK*** *Result:* The ***Method Outline*** pane shows the included phases for the chosen method and the ***Phase Properties*** tab shows the default settings for the currently highlighted phase.

Step	Action
	

Method Phases
Method Settings
Equilibration
Sample Application
Column Wash
Elution
Re-Equilibration
Delete
Save Phase...
Duration & Variables
Phase Properties
Text Instructions
Method Settings
Column selection
Show by technique
Affinity
Column type
Any
Show only suggested columns
Column Properties...
Column volume
0.100
ml
Pressure limit pre-column
2.00
MPa [0.02 - 2.00]
Pressure limit delta-column
2.00
MPa [0.02 - 2.00]
Column position
By-pass both
Flow rate
20.0
ml/min [0.0 - 600.0]
Control the flow to avoid overpressure
Inlet A
A1
Inlet B
B1
Mixer valve
Air Trap valve
In-line
Pressure limit filter
0.50
MPa [0.02 - 2.00]
Show Filter Connection...
Result Name & Location...
Start Protocol...
Method Notes...
Unit selection
Method base unit
CV
Flow rate unit
ml/min
Monitor settings
Wavelengths [190 - 700]
nm
UV 1
280
nm
UV 2
254
nm
UV 3
214
nm
Enable Air Trap valve air sensor alarm
Column Logbook
Enable logging of
Cleaning in Place
Column Performance Test
Default curve
UV

3.3.2 Edit phase properties

Introduction

When editing ***Phase Properties*** for a phase, the changes affect either:

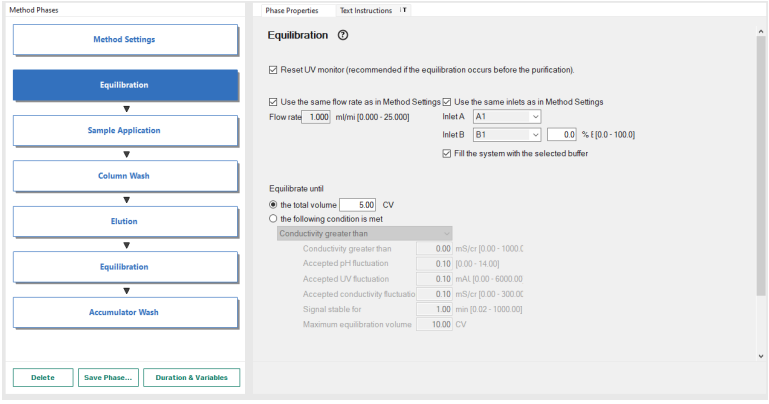
- the whole method, when editing the ***Method Settings*** phase

  or

- only the phase that is being edited, when editing phases other than the ***Method Settings*** phase

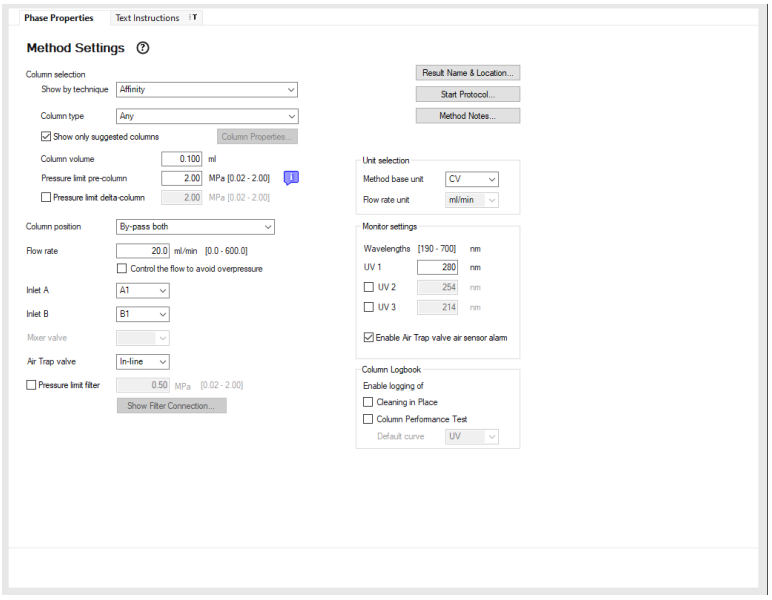
Getting help when editing Phase Properties

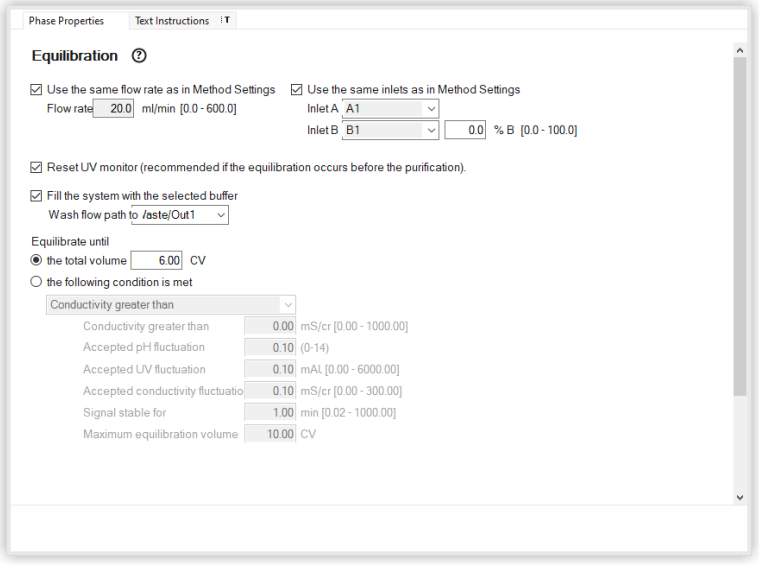
Follow the instructions to get help information for the properties in a phase:

Step	Action
1	Select a phase in the method to be edited, for example, ***Equilibration***. *Result:* The properties for the selected phase are displayed in the ***Phase Properties*** tab. 
2	Click anywhere in the ***Phase Properties*** tab to make it the active area in the software.
3	• Press the **F1** keyboard key. *or* • Click ***Contextual Help*** on the ***Help*** menu *Result:*The online help for the selected phase is displayed.

View and edit phases using Phase Properties

Follow the instructions to edit a method phase in the ***Phase Properties*** tab:

Step	Action
1	Click the ***Phase Properties*** tab.
2	• Select the ***Method Settings*** phase if you want to edit basic settings affecting the whole method (e.g., ***Column type, Flow rate*** and ***Method Base Unit***). Continue with steps 3-4. ***Note:*** *You can also edit the **Result name & Location**, the **Start Protocol** and **Method Notes** from the **Method Settings** phase. These are overall method options that also can be set using the corresponding **Toolbar** options and not described in this section. See Section 3.6.2 Set general method options for the method, on page 56 for information on how to edit these settings.* *or* • Select any other phase to edit the properties for that specific phase. Continue with step 5.
3	To edit the properties for the ***Method Settings*** phase, click ***Method Settings*** in the ***Method Outline***. *Result:* The ***Phase Properties*** of the ***Method Settings*** phase is displayed. 

Step	Action
4	Edit the settings for the ***Method Settings*** phase in the ***Phase Properties*** tab as appropriate. If changing ***Column type***, UNICORN will automatically calculate correct settings for volume, flow rate, and pressure limits. ***Note:*** *Settings in this phase will affect the whole method.* ***Note:*** *Allowed parameter ranges are shown in parentheses beside the text boxes.* *Result:* The method is updated with the new settings.
5	Select a phase in the method to be edited, for example, ***Equilibration***. *Result:* The properties for the selected phase are displayed in the ***Phase Properties*** tab. 
6	**a.** Edit the settings as appropriate. ***Note:*** *If there are, for example, two predefined* ***Equilibration*** *phases in your method, changing settings in one of them will not affect the other. To be able to see that they are different, it is recommended to rename one of them. See* *Section 3.6.1 Edit the method outline, on page 51* *for information about how to rename a phase.* **b.** Repeat steps 5-6 until the appropriate phases have been edited.

Step	Action
	Result: The method is updated with the new settings. The edited settings remain in place while subsequent phases are edited. If the method is closed and not saved, the settings will revert back to the earlier values.
7	Save the method.

3.3.3 Fraction collection

Introduction

For many purification schemes it is convenient to collect fractions of the eluent. Several of the predefined phases and methods include options for fraction collection in the ***Phase Properties*** tab.

This section describes briefly the various options available for fractionation in predefined methods and phases, and how to set up fraction collection when editing a method. More detailed information for individual settings can be found using the online help for the phase, see *Getting help when editing Phase Properties, on page 33*.

Fractionation overview

Fractionation is available in the ***Phase Properties*** tab in the predefined phases ***Sample Application***, ***Column Wash*** and ***Elution***. These three phases are included in many of the predefined methods in UNICORN. This option will also be available in personal or global phases derived from these. See *Section 3.3.2 Edit phase properties, on page 33* for details on how to edit methods and phases.

For each phase, fractions can either be collected using the outlet valve or the fraction collector. If there is no risk of sample loss, the eluate may be sent to the waste and not collected. When fractionating to the outlet valve, a specific outlet valve position is selected. When collecting fractions in the fraction collector a tube or plate type is chosen and the fractions will be collected in the first available tube or plate of that type.

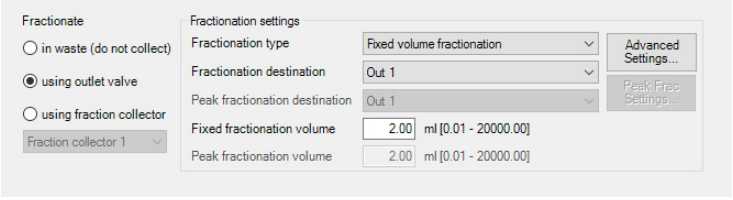
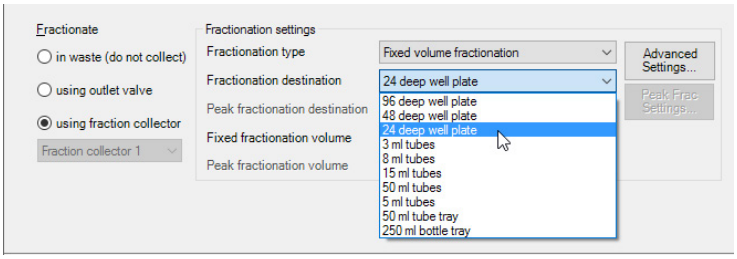
WARNING

Built-in fraction collector. Do *not* fractionate flammable liquids on instruments with built-in fraction collector. Flammable gas might be formed inside the closed cabinet. When running RPC methods, or other procedures using organic solvents, collect fractions through the Outlet valve.

Fractionation setup

The following instruction is an example of how to set up fraction collection in the ***Phase Properties*** tab:

Note: *The setup can vary between different fraction collectors.*

Step	Action
1	Select the phase for which fractionation is required in the method outline and click the ***Phase Properties*** tab. ***Note:*** *Text edited phases will show the fractionation options as variables in the* ***Phase Variables*** *list, see* *Chapter 10 Text edit methods, on page 252**.*
2	Below the heading ***Fractionate***, click the fractionation type required for this phase: **a.** ***using outlet valve*** enables fraction collection using the outlet valve. The ***Fractionation settings*** will change to reflect this choice, and the outlet valve position can be selected as the ***Fractionation destination***.  **b.** ***using fraction collector*** enables fraction collection in the fraction collector. The ***Fractionation settings*** will change to reflect this choice, and the desired ***Fractionation destination*** can be selected in the drop-down list.  **c.** ***in waste (do not collect)*** will direct the eluent to the waste.
3	Edit the ***Fractionation settings*** as appropriate. For detailed information on these settings see the online help for the phase, refer to *Getting help when editing Phase Properties, on page 33*.

3.4 Working with wizard generated methods

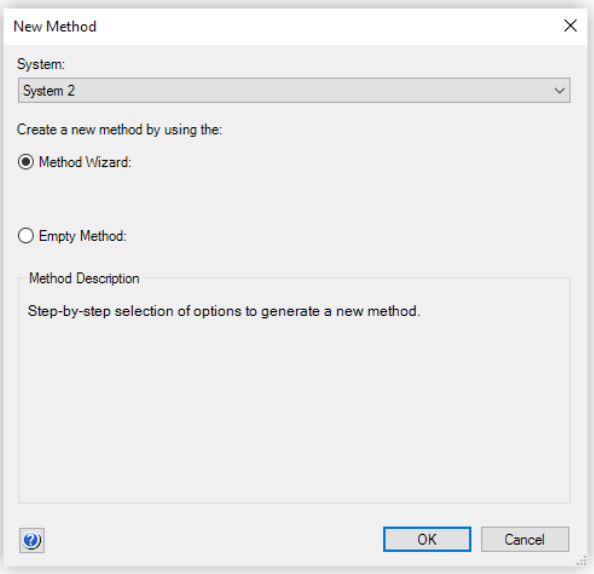
About this section

For instruments with an instrument configuration that includes a method wizard, it is possible to create wizard generated methods. This section describes how to create and edit wizard generated methods. The wizard generated methods only contain a method settings phase and a user defined phase. The sections also describes ***Frac-950*** fractionation.

In this section

3.4.1 Create a wizard generated method

Follow the instructions to create a new method using the wizard:

Step	Action
1	In the ***Method Editor***: • click the ***Create a new method*** button in the toolbar  or • click ***New Method*** on the ***File*** menu *Result:* The ***New Method*** dialog box opens. 
2	In the ***New Method*** dialog box: **a.** select a system in the ***System*** drop-down list. ***Note:*** *The dialog box changes depending on the type of the selected system. For systems that use predefined methods, refer to Section 3.3 Working with predefined method, on page 30.*

Step	Action
	b. click ***Method Wizard***
	Note:
	The dialog box changes depending on the type of the selected system. For systems that use predefined methods, refer to Section 3.3 Working with predefined method, on page 30.
	c. click ***OK***
	Result:
	The ***Method Wizard*** opens.
3	**a.** Follow the instructions in the ***Method Wizard*** until you reach the last page of the Wizard.
	Note:
	Performance tests, CIP and other special methods are also created in the ***Method Wizard****.*
	Press **F1** to open the Wizard help if needed..
	b. Click ***Finish***.

3.4.2 Wizard generated methods

The methods generated by the wizard will consist of two phases.

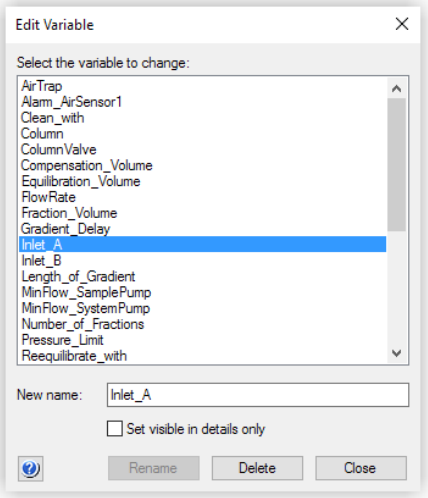
- The ***Method Settings*** phase, which contains Column type, result name and location, start protocol and method notes. Logging of Column performance test and Column CIP is selected in this phase.
- The ***User Defined*** phase, which contains a list of ***Phase variables*** that can be text edited.

Edit phase variables

Editing a variable includes renaming and deleting the variable and choosing whether the variable should be a detailed variable or not.

Follow the instructions to edit a variable for a selected phase:

Step	Action
1	Click the ***Phase Properties*** tab to display the phase variables, select the variable ***Value*** and click ***Edit Variable***. *Result:* The ***Edit Variable*** dialog box opens displaying all the phase variables.
2	Select the variable to be edited (if not already selected). Do one or several of the following as appropriate:



a. Type in a new name in the ***New name*** box and click ***Rename***.

b. Select the ***Set visible in details only*** check box if the variable should be a detailed variable. Clear the check box to set it to a normal variable.

Step	Action
	c. Click ***Delete*** to delete the variable. Confirm that you want to delete the variable in the message box that appears.
3	Click ***Close*** to close the dialog box.

Edit variable values

Follow the instructions to edit default variable values in the ***Phase Variables*** list. For information on how to edit variables from ***Text Instructions***, see *Section 10.2 Working with methods in the Text Instructions tab, on page 260*.

Step	Action
1	Click the ***Phase Properties*** tab to display the ***Phase Variables*** list.
2	Change the variable value for the appropriate variable in the ***Value*** box by selecting a new value in the drop-down list or by entering the value in the box. ***Tip:*** *To show detailed variables, check the* ***Show details*** *check box.* *Result:* The variable value is updated.
3	Repeat this procedure for the appropriate variables.

Changes made in the ***Phase Properties*** tab are automatically updated on the ***Text Instructions*** tab and vice versa.

3.4.3 Frac-950

Introduction

For many purification procedures it is convenient to collect fractions of the eluent. The wizard generated methods include settings for fraction collection using Frac-950 in the ***Phase Properties*** tab if Frac-950 has been chosen as a component for the system.

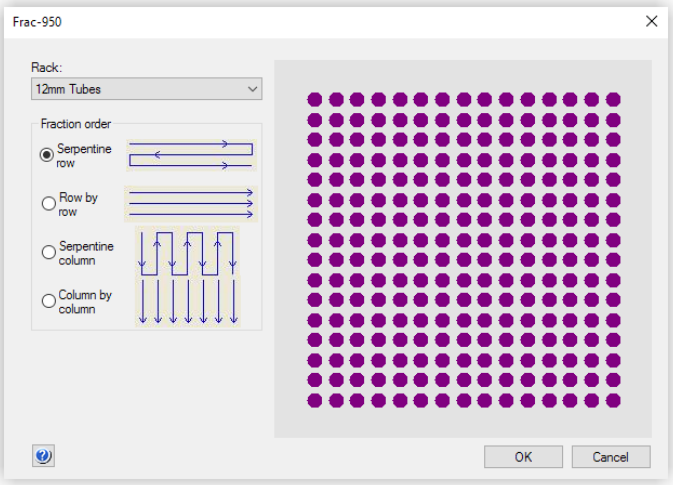
This section describes how to edit a ***Frac-950*** fractionation. More detailed information for individual settings can be found in the online help.

Note: ***Frac-950*** *is a system specific option.*

Fractionation

Follow the instructions to edit a ***Frac-950*** fractionation.

Step	Action
1	• Click ***Frac-950*** in the ***Phase Properties*** tab in the ***Method settings*** phase. *or* • Click ***Frac-950*** on the ***Edit*** menu. ***Note:*** *The* ***Frac-950*** *command is only available if Frac-950 has been chosen as a component for the system.* *Result:*The ***Frac-950*** dialog box opens.



2	**a.** Select rack type in the ***Rack*** drop-down list. **b.** Select what kind of ***Fraction order*** to use in the fractionation.

Step	Action
	c. Click ***OK***. ***Tip:*** *Settings for defining the last tube are available in the* ***Fraction collector*** *page of the* ***Start Protocol*** *when you start a method run.*

3.5 Working with empty methods

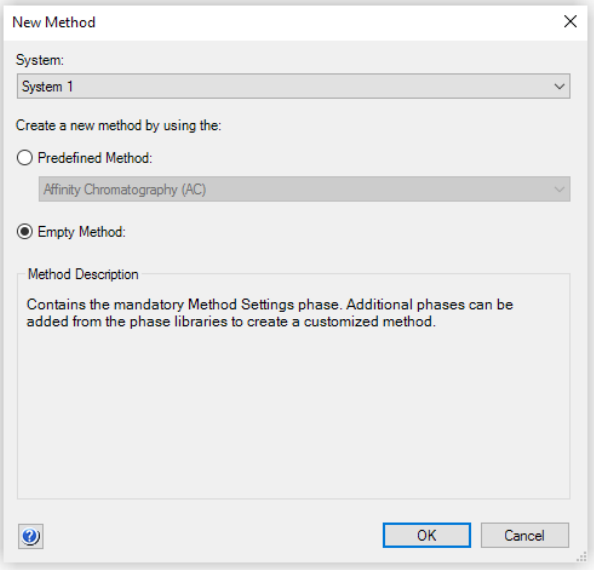
About this section

This section describes how to work with methods and phases in systems that create methods using text instructions. From the start, an empty method only contains a method settings phase.

In this section

3.5.1 Create an empty method

Follow the instructions to create a new empty method:

Step	Action
1	In the ***Method Editor***: • click the ***Create a new method*** button in the toolbar  or • click ***New Method*** on the ***File*** menu. *Result:* The ***New Method*** dialog opens.
2	The ***New Method*** dialog looks different depending on the system.  **a.** select a system in the ***System*** drop-down list **b.** click ***Empty Method*** **c.** click ***OK*** *Result:* An empty method that consists of the mandatory ***Method Settings*** phase is created.

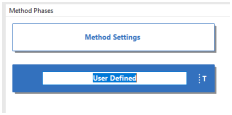
3.5.2 Edit an empty method

Introduction

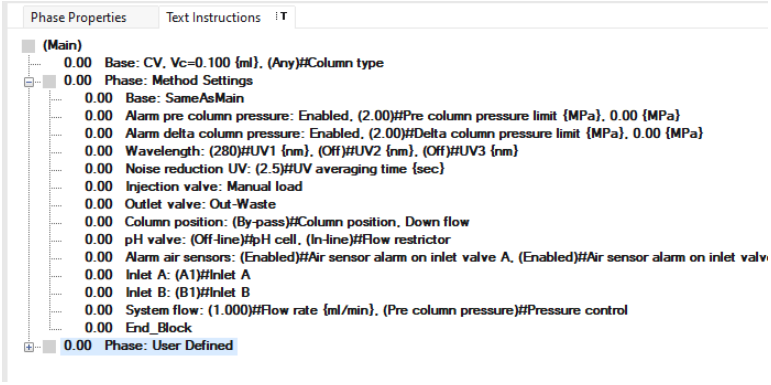
A new empty method only has the ***Method Settings*** phase in the ***Method Outline***. ***User Defined*** and ***Predefined*** phases can be added, rearranged, renamed and deleted from the ***Method Outline***.

Add a phase to the method outline

Follow the instructions to add a phase to the method outline using drag-and-drop:

Step	Action
1	Select the phase in the ***Phase Library*** pane and drag-and-drop the phase to the requested position in the ***Method Outline*** pane. In the example, a ***User Defined*** phase is used. *Result:* The phase is included in the method at the requested position.
2	When the ***User Defined*** phase has been added to the ***Method Outline***, the phase name is enabled for editing.  Type a name for the phase and press the **Return** keyboard key. ***Note:*** *The **User Defined** phase is marked with the letter **T**, meaning that it is text edited. This phase contains only **Base** and **End_Block** instructions, so any functional instructions must be added manually.*

Step	Action
3	To include instructions for the ***User Defined*** phase, select the ***Text Instructions*** tab and text edit the method.



For detailed instructions on how to text edit methods, refer to *Chapter 10 Text edit methods, on page 252*

Note:

*The **Phase Properties** tab will then show a list of variables used in the phase.*

3.6 Working with methods in general

About this section

This section includes descriptions of how to work with methods and phases in general.

In this section

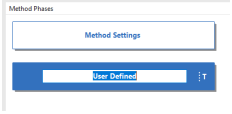
3.6.1 Edit the method outline

Introduction

The ***Method Outline*** shows the phases that are included in the method and the order of the phases in the method. Phases can be added, rearranged, renamed and deleted from the ***Method Outline***.

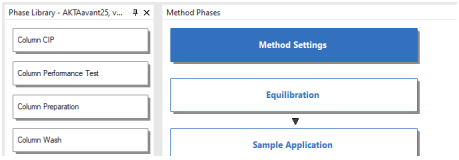
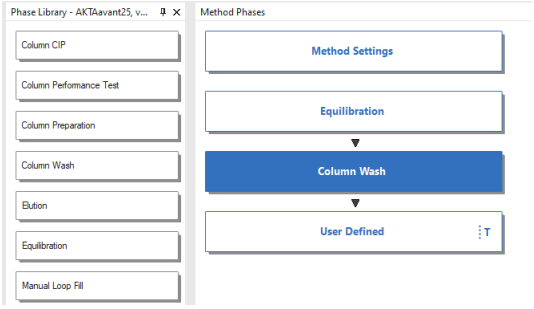
Add a phase to the method outline using drag-and-drop

Follow the instructions to add a phase to the method outline using drag-and-drop:

Step	Action
1	Select the appropriate phase in the ***Phase Library*** pane and drag-and-drop the phase to the requested position in the ***Method Outline*** pane. All systems have a user defined phase available and some system has several predefined phases available in the phase library. *Result:* The phase is included in the method at the requested position. If the ***User Defined*** phase was added, continue with step 2.
2	When the ***User Defined*** phase has been added to the ***Method Outline***, the phase name is enabled for editing.  Type a name for the phase and press the **Return** key. ***Note:*** *The **User Defined** phase is marked with the letter **T**, meaning that it is text edited. This phase contains only **Base** and **End_Block** instructions, so any functional instructions must be added by hand. To include instructions for the **User Defined** phase, select the **Text Instructions** tab. The **Phase Properties** tab will only show the variables used in this phase. See Chapter 10 Text edit methods, on page 252 for information about how to work with instructions in the **Text Instructions** tab.*

Add a phase to the method outline using a button or menu command

Follow the instruction to add a phase to the method outline using a button or a menu command. It is possible to define several phases and store in the library to use in new methods.

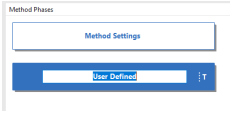
Step	Action
1	**a.** Select the appropriate phase (e.g., ***Equilibration*** or ***User Defined***) in the ***Phase Library***
	b. Select the appropriate phase (e.g., the ***Method Settings*** phase) in the ***Method Outline*** to determine where to place the new phase
	Note:
	*When adding a phase to the **Method Outline** using a button or menu command, the new phase is always inserted below the currently selected phase in the **Method Outline**.*
	Result:
	The selected phase in the ***Phase Library*** is indicated by a dotted frame and the selected phase in the ***Method Outline*** is highlighted.
	
2	• Click ***Insert*** located below the ***Phase Library***
	or
	• double-click the selected phase
	or
	• click ***Insert Phase from Library*** on the ***Phases*** menu
	or
	• right-click the phase in the ***Phase Library*** and click ***Insert***
	*Result:*The phase is included in the method and highlighted. Continue with step 3 if adding the ***User Defined*** phase.
	
3	When the ***User Defined*** phase has been added to the ***Method Outline***, the phase name is enabled for editing.
	Type a name for the phase and press the **Return** key.

Step	Action
	Note: *The **User Defined** phase is marked with the letter **T**, meaning that it is text edited. This phase contains only **Base** and **End_Block** instructions, so any functional instructions must be added by hand. To include instructions for the **User Defined** phase, select the **Text Instructions** tab. The **Phase Properties** tab will show a list of variables used in this phase. See Chapter 10 Text edit methods, on page 252 for information about how to work with instructions in the **Text Instructions** tab.*

Rename phases

Note: *It is only possible to rename phases in the **Method Outline** pane, not in the **Phase Library**.*

Follow the instruction to rename a phase in the method:

Step	Action
1	Select the phase to be renamed in the ***Method Outline*** pane.
2	• right-click the phase and click ***Rename*** *or* • press the **F2** key *or* • click ***Rename*** on the ***Edit*** menu *Result:*The name in the phase becomes editable. 
3	Type an appropriate name and press the **Return** keyboard key. *Result:* The name of the phase is updated.

Re-arrange phases within a method

Follow the instruction to re-arrange phases within a method:

Step	Action
1	Select the phase to be moved in the ***Method Outline*** pane.
2	• Drag-and-drop the phase to the requested position in the ***Method Outline*** pane.

Step	Action
	*Result:*The phase is moved to the requested position. *or* • Right-click the phase and click ***Move up*** or ***Move down***. *Result:*The phase is moved one step up or down in the ***Method Outline***.

Delete a phase from the Method Outline

Follow the instruction to delete a phase from the ***Method Outline***:

Step	Action
1	Select the phase to delete from the method in the ***Method Outline***.
2	• Click ***Delete*** below the ***Method Outline*** pane. *or* • Press the **Delete** key on the keyboard. *or* • click ***Delete*** on the ***Edit*** menu *or* • Right-click on the phase and click ***Delete***. *Result*The phase is removed from the method.

Copy, cut, and paste phases in a method

Note: *It is only possible to copy, cut and paste phases in the* ***Method Outline*** *pane, not in the* ***Phase Library****.*

The features copy, cut and paste phases can be used to add/delete and rearrange phases in the ***Method Outline***. Follow the instructions to copy, cut or paste a phase features.

To...	then...
copy a phase	select the phase and: • right-click the phase and click ***Copy*** *or* • press ***Ctrl + C*** *or* • click ***Copy*** on the ***Edit*** menu.

To...	then...
cut a phase	select the phase and: • right-click the phase and click ***Cut*** *or* • press ***Ctrl + X*** *or* • click ***Cut*** on the ***Edit*** menu.
paste a phase	***Note:*** *The phase to be pasted will be pasted below the phase highlighted in the* ***Method Outline****.* Select the appropriate phase in the ***Method Outline***. Then: • right-click the phase in the ***Method Outline*** and click ***Paste*** *or* • press ***Ctrl + V*** *or* • click ***Paste*** on the ***Edit*** menu.

3.6.2 Set general method options for the method

Introduction

This section describes how to set and view options for an entire method. The following are covered in this section:

- Defining the name and location for the results.
- How to set up a ***Start Protocol*** that will be displayed before each method run.
- Adding or changing method notes.
- How to include evaluation procedures which can be executed during the run.
- Viewing the method duration time and volume.
- Viewing the variables used in the method.

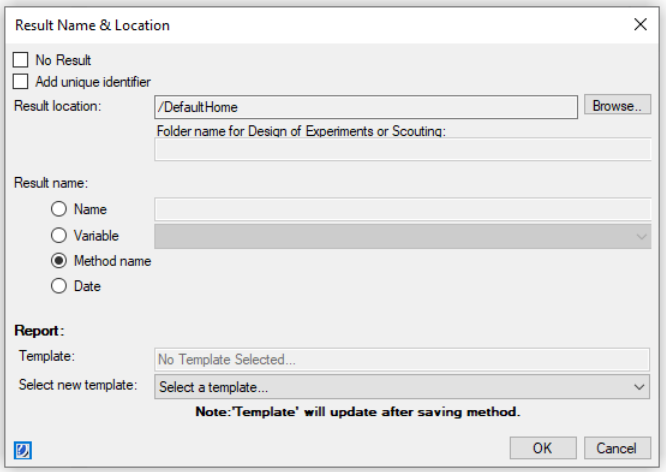
Define Result Name & Location

Follow the instruction to define the name of the result file created after the run and how to specify the folder in which to save the result file.

Step	Action
1	In the ***Method Editor***: • click the ***Result Name and Location*** button  or • click ***Result Name and Location*** on the ***Edit*** menu or • click the ***Method Settings*** phase and click the ***Result Name and Location*** in the ***Phase Properties*** tab *Result:* The ***Result Name and Location*** dialog box opens.

Step	Action

2



In the ***Result Name and Location*** dialog box:

a. Select the ***Add unique identifier*** check box if you want to include a unique identifier number to the file name. The number will be generated by UNICORN based on the run time of the method.

b. Set ***Result location*** by clicking ***Browse*** and select a folder in which to save the results. By default, the results will be saved in your home folder.

c. Choose the result name by clicking:

- ***Name***. The result name can be typed in manually
- ***Variable***. The result name will be generated from the chosen variable (see *Section 10.2.4 Method variables, on page 277*)
- ***Method name*** (default). The result name will be generated from the name of the method
- ***Date***. The result name will be generated from the date of the run

3 Select a template for your Report:

a. Select a template from the ***Select new template*** drop-down list.

The selected method will be shown in the ***Template*** field and will be used for auto-print after the method run.

b. Click ***OK*** to confirm and close the dialog box.

Set up a Start Protocol

Follow the instruction to set up a ***Start Protocol*** to be displayed before the run starts.

Step	Action
1	In the ***Method Editor***: • click the ***Start Protocol*** button  or • click ***Start Protocol*** on ***Tools*** menu or • click the ***Method Settings*** phase and click ***Start Protocol*** in the ***Phase Properties*** tab *Result:* The ***Start Protocol*** dialog box opens.
2	In the ***Start Protocol*** dialog: **a.** Select items to display at method start. When selecting a method item, a description is shown to the right. ***Result Name and Location*** is selected by default. **b.** Click ***OK*** to confirm and close the dialog.

Add/edit Method Notes

Follow the instruction to add/edit notes to a method.

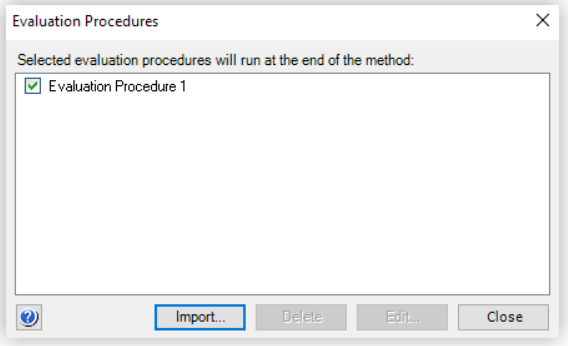
Step	Action
1	In the ***Method Editor***: • click the ***Method Notes*** button  or • click ***Method Notes*** on the ***Edit*** menu or • click the ***Method Settings*** phase and click ***Method Notes*** in the ***Phase Properties*** tab *Result:* The ***Method Notes*** dialog box opens.

Step	Action
2	In the ***Method Notes*** dialog box: **a.** Enter/edit notes about the method. If notes already have been entered, it is possible to search for specific words by clicking ***Find***. **b.** Click ***OK*** to confirm and close the dialog box. ***Note:*** *For some systems information will automatically be added to the* ***Method Notes*** *if the method has been converted for use with another system type than it was originally created for, or scaled for another Column type than was originally selected, or if it is a wizard generated intelligent packing method.*

Include Evaluation Procedures after the run

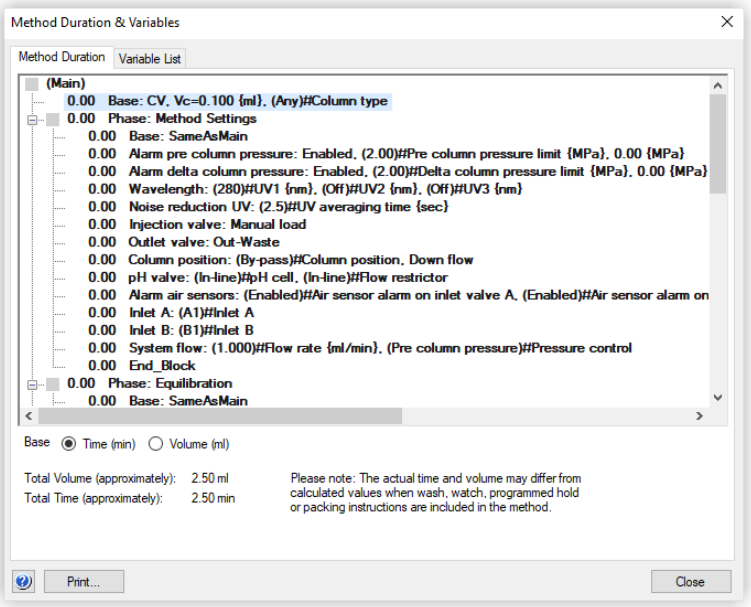
Follow the instruction to include an evaluation procedure in the method. The evaluation procedure will be performed automatically after the run has finished. The evaluation procedures must have been defined in the ***Evaluation*** module, see the UNICORN Evaluation Manual.

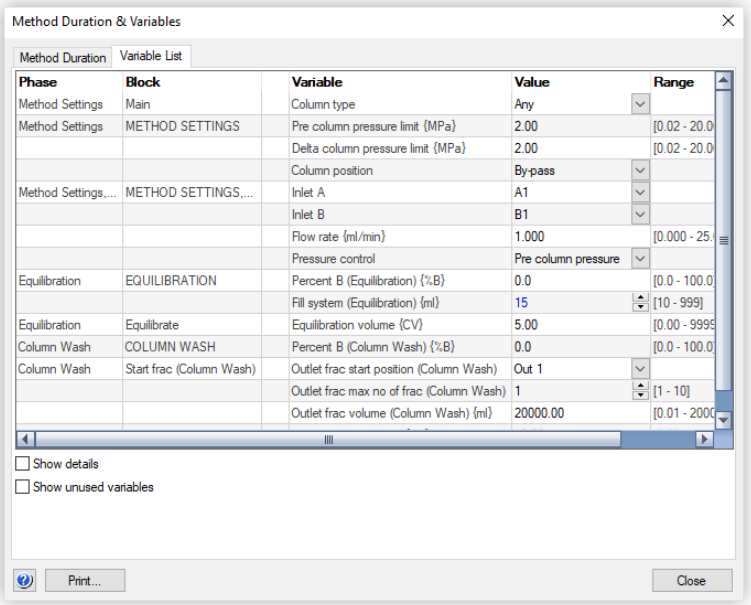
Step	Action
1	In the ***Method Editor***: • click the ***Evaluation Procedures*** button  or • click ***Evaluation Procedures*** on the ***Tools*** menu *Result:* The ***Evaluation Procedures*** dialog box opens.
2	**a.** If there are no evaluation procedures listed in the ***Evaluation Procedures*** dialog box, click ***Import*** to import an evaluation procedure. *Result:* The ***Import Evaluation Procedure*** dialog box opens. Continue with step 3. **b.** If an evaluation procedure that should be used in the run has been saved in the method earlier, it is shown in the ***Evaluation Procedures*** dialog. Continue with step 4.
3	**a.** Select the appropriate procedure to import in the ***Select procedure to import*** field.

Step	Action
	b. It is possible to change the name of the procedure to be displayed in your method by changing the name in the ***Import as*** box. **c.** Click ***Import*** to import the procedure In the ***Import Evaluation Procedures*** dialog box: ***Note:*** *Only **Global** procedures and your own **Personal** procedures are shown in the list.* ***Note:*** *It is also possible to import a procedure saved in another method by browsing to the appropriate folder and selecting the method containing the procedure. The procedure will be listed in the **Select procedure to import** field and can be imported as described above.* *Result:* The evaluation procedure is listed in the ***Evaluation Procedures*** dialog box.
4	 Select the check box in front of the evaluation procedure to include it in the method, and then click ***Close***. *Result:* The evaluation procedure is included in the method. ***Note:*** *It is possible to edit an existing evaluation procedure by selecting it and clicking **Edit**. The edits will only change the procedure that is included in the method. See the UNICORN Evaluation Manual for information about how to edit an evaluation procedure.*

View and print the method duration time and variables

Follow the instruction to view and print an estimation of the method duration time and the variables in the method:

Step	Action
1	In the ***Method Editor***, click ***Duration & Variables*** on the ***View*** menu. *Result:* The ***Method Duration & Variables*** dialog box opens displaying the ***Method Duration*** tab.
2	 The ***Method Duration*** tab shows an estimation of the accumulated method time and volume for the current method below the text method. If the method includes a scouting series, an estimation of the accumulated method time and volume for the total series of runs is displayed below the text method. ***Note:*** *Click the arrow buttons to display the different scouting runs.* *The accumulated time/volume is an approximation and does not take into account time or volume for* ***Watch*** *blocks,* ***Wash*** *commands or programmed* ***Hold****.*
3	**a.** Click ***Time*** as base to show the time in minutes in the text method. **b.** Click ***Volume*** as base to show the volume in the text method.

Step	Action
4	To view the variables in the method, click the ***Variable List*** tab. *Result:* The ***Variable List*** is displayed.
5	 The ***Variable List*** shows the variables in the method. It is also possible to see in which phases the variables are included and the different values. Variables with an ellipsis (`...`) after their name are used in multiple phases or blocks. It is not possible to change any values in this dialog. **a.** Select the ***Show details*** check box to view variables classified as detailed. The letter `D` will be shown to the left of the detailed variables. **b.** Select the ***Show unused variables*** check box to view unused variables in the method. The letter `U` will be shown to the left of unused variables.
6	To print the information in the ***Method Duration & Variables*** dialog box, click ***Print***. *Result:* The ***Print*** dialog box opens.
7	Select a ***Printer*** from the drop-down list and click ***OK***. *Result:* The information is printed.

3.6.3 Print a method

Introduction

This section describes how to print a method's text instructions and variables. UNICORN uses the printers and printer settings that are installed on your computer.

Print a method

Follow the instructions to print an opened method:

Step	Action
1	• Click the ***Print*** button  or • click ***Print*** on the ***File*** menu. *Result:*The ***Print*** dialog box opens.
2	In the ***Print*** dialog box: **a.** select printer in the ***Printer*** drop-down list **b.** select check boxes for the ***Print items*** to be printed **c.** click to print ***All phases*** in the method or a specific ***Phase*** in the ***Phase range*** field
3	By default, information about the overall method settings as well as any signatures and specific columns used in the method are printed. To exclude or add information, click ***Options*** and select or clear the appropriate check boxes. ***Note:*** *Only options that are used in a method can be printed. Options that are not available are dimmed in the* ***Print*** *dialog box.*
4	Click ***OK***. *Result:* The method is printed.

3.6.4 Sign methods electronically

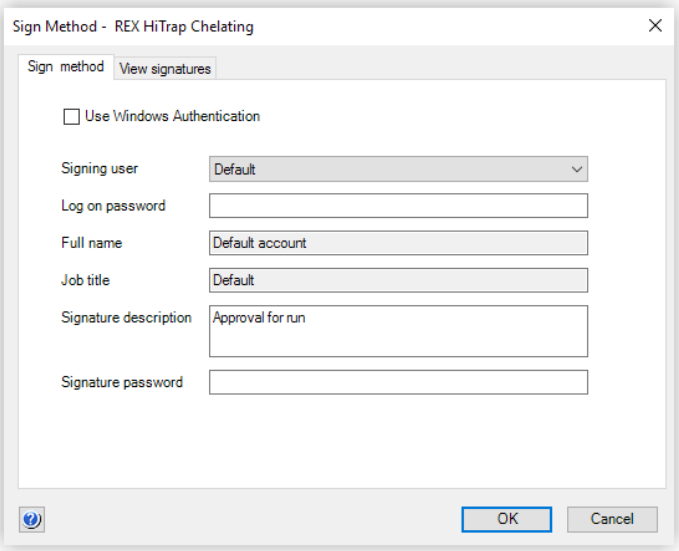
Introduction

Methods can be signed electronically to enhance data file security. Once a method has been signed, it is not possible to edit the method.

Tip: *To edit a signed method create a new method using the settings in the signed method by clicking **Save As** on the **File** menu and save the method with a new name.*

Sign a method electronically in the Method Editor

Follow the instruction to sign a method electronically in the ***Method Editor***:

Step	Action
1	Click ***Sign Method...*** on the ***File*** menu. *Result:* The ***Sign Method*** dialog box opens.
2	

a. The ***Signing user*** field shows the user currently logged on.

Note:

*If you want to **Use Windows Authentication**, select the check box, and log in as a network user.*

Step	Action
	If you want to sign the method but are not logged on to UNICORN, select your user name in the ***Signing user*** drop-down list. Your ***Full name*** and ***Job title*** are displayed.
	b. Type your ***Log on password*** to UNICORN.
	c. Type a ***Signature description*** if appropriate.
	d. Type your ***Signature password*** and click ***OK***.
	Result:
	The method has been signed.

3.6.5 Save methods and phases

Introduction

Methods and phases are saved in the UNICORN database.

Individual, edited phases may be saved to the ***Phase Library*** for later use in other methods on systems having the same instrument configuration and component configuration.

Note: *You cannot save an edited method/phase to replace a predefined method or phase. If you want to save an edited variant of a predefined method or phase with specific settings, you must save it under another name. A predefined method or phase cannot be overwritten.*

Save a method

Follow the instruction to save a method in UNICORN.

Step	Action
1	• Click the ***Save the Method*** button  or • click ***Save*** or ***Save As*** on the ***File*** menu. *Result* • If the method has been named and saved previously, the changes are saved immediately. *If not* • The ***Save As*** dialog box opens. Proceed with steps 2-4 below.
2	Browse for an appropriate folder, or create a new one.
3	**a.** Select the folder in which to save the results. **b.** Enter a method ***Name***. **c.** Select for which ***System*** to save the method.
4	Click ***Save***. *Result:* The method is saved in the database. ***Note:*** *For some systems an error message will appear if you are trying to save the method for:*

Step	Action
	• *a system using another instrument configuration and/or another component configuration than the method originally was created for* *and* • *the settings in the method depend on the component configuration (e.g., if an extra inlet A valve is used in the method, this setting cannot be used in a system lacking the extra inlet A valve.)* *It will still be possible to save the method but the phases in the method will be marked with an error symbol. In order to be able to subsequently run the method, either the method must be text-edited or the component configuration of the system changed in the* ***Administration*** *module.*

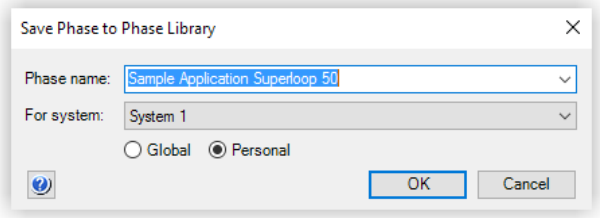
Adapt a method

For ÄKTA pure™ it is possible to open and save methods created with systems with another set of components than the currently used system. When a method is opened for a system that has been changed or if a method is saved for a different system, a dialog box is opened where the user can choose to either adapt the method or to keep the text method unchanged.

If you select..	Then..
Adapt method	the method will be adapted to a new set of components. All functions and positions that are still available will be unchanged. For example, valve positions present on both the original and the new system will be unchanged in the adapted method. Non compatible settings will be adjusted in order to properly adapt the method for new components. Settings that have been changed are described and saved in Method Notes, which are displayed after the method has been adapted. Change the settings in ***Phase Properties*** if required.
Keep text method	the text method will be kept and nothing in the text method will be changed. Non-compatible settings will still be present in the method but they will not be functional. All ***Phase Properties*** of the method will be replaced by tables of ***Phase Variables***. ***Note:*** *Some predefined methods require certain components to be functional. Adapting these methods to systems that do not include the required components is not possible.*

Save a phase

Follow the instruction to save a phase to the Phase Library:

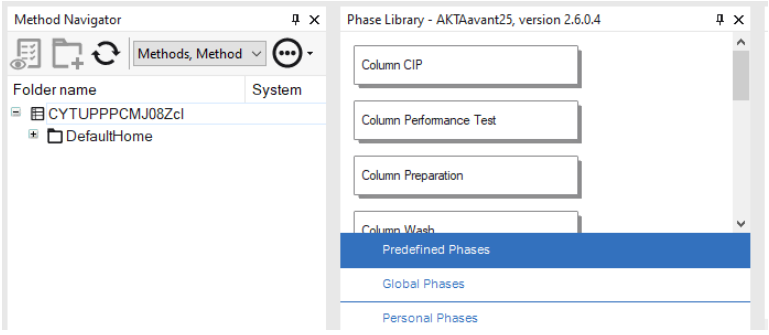
Step	Action
1	Select the phase to be saved in the method outline. ***Note:*** *A **Method Settings** phase cannot be saved as a separate phase with a new name. If properties for the **Method Settings** phase are changed, the changes will be saved with the method.*
2	• click ***Save Phase*** below the ***Method Outline*** pane *or* • click ***Save Phase*** on the ***Phases*** menu *or* • right-click the phase and click ***Save Phase*** *Result*The ***Save Phase to Phase Library*** dialog box opens. 
3	• Type a ***Phase name*** *or* • Select a phase from the ***Phase name*** drop-down list. This phase will be replaced by the phase with the new settings.
4	In the ***For system*** box, the system that was selected when the current method was set up will be displayed by default. To save the phase for another system, select the appropriate system from the ***For system*** drop-down list. ***Note:*** *Only systems using the same instrument configuration and component configuration as the system that was selected when the current method was set up will be displayed in the **For system** box.*
5	**a.** Select if the phase shall be ***Global*** (available for all users) or ***Personal*** (for your own use only). **b.** Click ***OK***. ***Note:*** *It is not possible to replace a predefined phase by saving an existing phase.*

Step	Action
	Result: The phase is saved and is available in the ***Global Phases*** or ***Personal Phases*** panel of the ***Phase Library***.

Delete a phase from the Phase Library

It is possible to delete personal and global phases from the ***Phase Library***. Predefined phases cannot be deleted.

Follow the instruction to delete a personal or global phase from the phase library:

Step	Action
1	Select the appropriate phase library: ***Personal Phases*** or ***Global Phases*** at the bottom of the ***Phase Library*** pane.  *Result:* The phases in that phase library are displayed.
2	Select the phase to delete from the library.
3	• Click ***Delete*** at the bottom of the ***Phase Library*** pane. *or* • Right-click the phase and click ***Delete***. *Result*The phase is removed from the ***Phase Library***.

3.6.6 Scale or convert methods

Introduction

UNICORN methods are always created specifically for a designated system and thus also for a specific system type. However, it is often useful to convert a method that was originally created for a system of one type, for use with a system of another type. The converted method is created as a copy of the original method. The original method remains unchanged.

The possibility to scale or convert methods is only available for some systems, and only between systems that use the same instructions.

A method can also be scaled for use with a different Column type than it was originally created for, when the method is converted. Conversion and scaling is described in this section.

Tip: *If you wish to use a method for another system of the same system type that it was originally created for, you only need to click **Save As** on the **File** menu, select the new system and save the method with another name. This is possible for all systems.*

Instructions marked with a red cross must be edited.

To only change the selected Column type, you should edit the method, change the column selection and save the edited method.

Prerequisites

The following items should be considered to ensure that the method conversion and scaling is successful:

- The original method should use column volume (CV) as base
- All parameters that will require scaling should be defined as variables
- If linear flow is to be maintained in the scaled method, it must have been applied in the original method as well
- Scaling of the Column type is not possible if the ***Any*** Column type was selected in the original method
- Text edited phases will not be automatically updated during the conversion

Convert a method to another system type

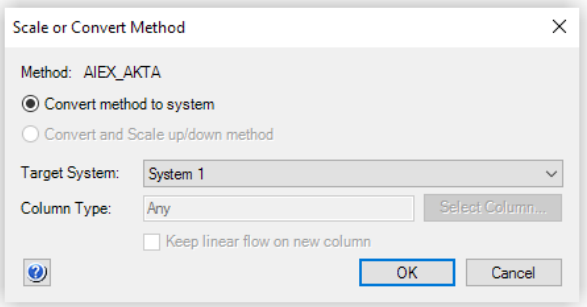
Follow the instructions to convert a method to be used with another system type.

Step	Action
1	Open the method you want to convert in the ***Method Editor***.
2	Click ***Scale or Convert Method*** on the ***File*** menu.

Step	Action

Result:

The ***Scale or Convert Method*** dialog box opens.



3 Click ***Convert method to system***.

4 Select the system to which the method should be converted to in the ***Target System*** drop-down list.

The list will show all available, active systems. Deactivated systems are not shown.

Tip:

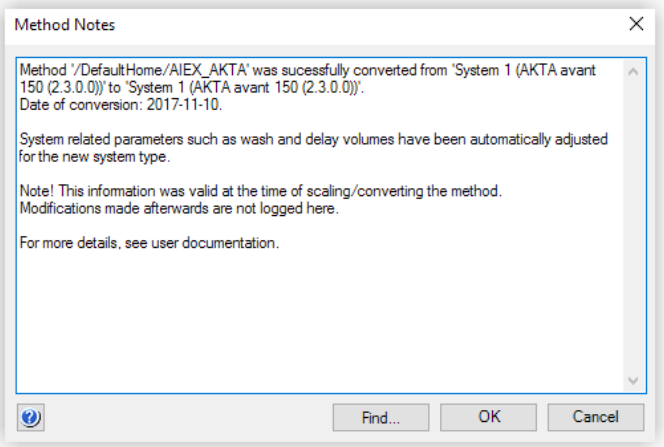
You can convert methods that originally were created for systems that now are deactivated.

5 Click ***OK***.

Result:

The method is converted as an untitled copy (***UNTITLED converted****). The ***Method Notes*** dialog box opens, showing basic information about the conversion.

Step	Action



Note:

*The information shown in the **Method Notes** will not include notes concerning method instructions that may have become invalid as a result of conversion between systems with different components and instrument configurations. You must verify that there are no phases with invalid instructions (i.e. phases marked with a red cross) in the new method before it can be used. See note below this instruction.*

6 **a.** Type any additional notes you wish to add in the text field

and

b. click ***OK*** to close the ***Method Notes*** dialog box.

7 • Click ***Save As*** on the ***File*** menu command to save the converted method

or

• click the ***Save*** button

*Result:*The ***Save As*** dialog box opens, with the folder where the original method is saved open by default.

8 **a.** Select the desired target folder,

b. type a new method name in the ***Name*** box

and

c. click ***Save***.

Note: *The original method remains after the converted method is saved. However, the converted method will replace the original if you choose to save the converted method with the same name in the same folder as the original.*

Note: *The flow rate and/or pressure settings in the method will automatically be adjusted if the maximum flow rate and/or pressure values for the target system is exceeded after the conversion. The maximum settings for the target system will be used by default.*

Note: *In case the original method contains instructions that are not supported by the new system, this will be indicated in the method outline of the converted method as a red cross on the phase containing these instructions.*

To be able to run the method on the new system you need to replace or remove the invalid instructions in the ***Text Instructions*** *tab. Invalid instructions are indicated with red square symbols in the text instructions.*

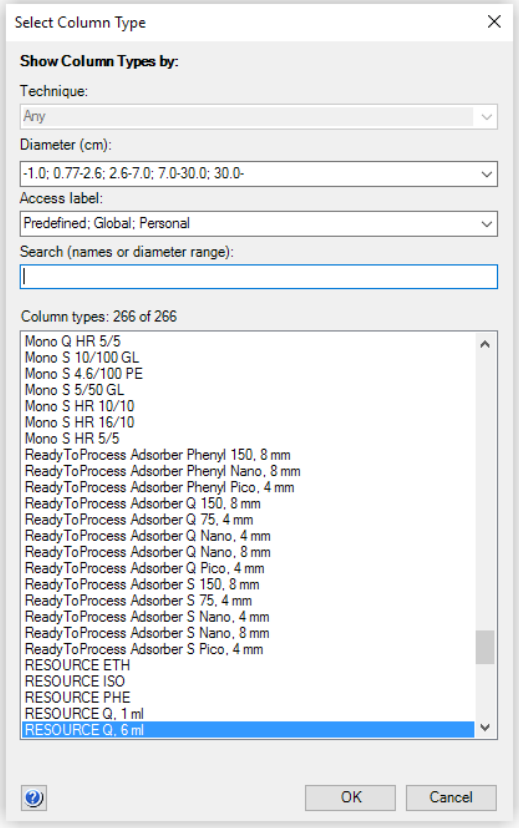
You can also replace the phase with a predefined phase from the phase library.

Convert and scale a method up or down

Follow the instructions to convert a method to be used with another system type, and at the same time scale the method to be used with another column size.

Step	Action
1	Open the method you want to convert and scale in the ***Method Editor***.
2	Click ***Scale or Convert Method*** on the ***File*** menu. *Result:* The ***Scale or Convert Method*** dialog box opens.
3	Click ***Convert and Scale up/down method***.
4	Select the system to which the method should be converted in the ***Target System*** list.
5	By default, the same Column type that was selected in the original method is shown in the ***Column Type*** field. **a.** Click ***Select Column*** to select a new Column type. *Result:* The ***Select Column Type*** dialog box opens.

Step	Action

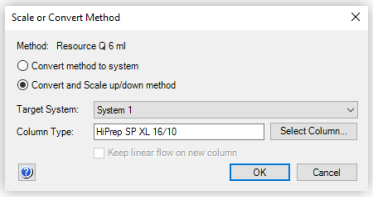


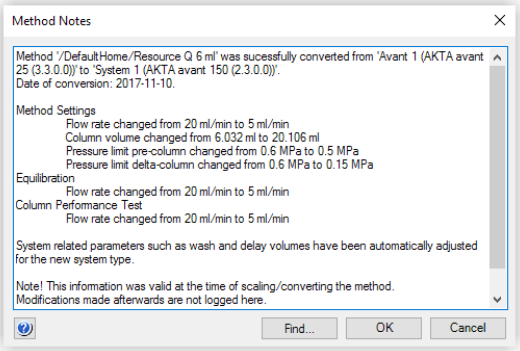
6 **a.** Select the Column type from the list of available Column types and

b. click ***OK***.

Result:

The ***Select Column Type*** dialog box closes and the selected column is shown in the ***Column Type*** box of the ***Scale or Convert Method*** dialog box.



Step	Action
7	If desired, select the ***Keep linear flow on new column*** check box. ***Note:*** *This check box is applicable only if linear flow was selected in the original method. If linear flow is not selected, the default flow settings for the selected Column type is used.*
8	Click ***OK***. *Result:* The method is scaled as an untitled copy (***UNTITLED converted****). The ***Method Notes*** dialog box opens, showing basic information about the conversion.  ***Note:*** *The information shown in the **Method Notes** will not include details about scaled system related parameters (e.g. wash or delay volumes) or notes concerning method instructions that may have become invalid as a result of conversion between systems with different components and instrument configurations. You must verify that there are no phases with invalid instructions (i.e. phases marked with a red cross) in the new method before it can be used. See note below this instruction.*
9	**a.** Type any additional notes you wish to add in the text field and **b.** click ***OK*** to close the ***Method Notes*** dialog box.
10	• Click ***Save As*** on the ***File*** menu *or* • click the ***Save*** button *Result:*The ***Save As*** dialog box opens, with the folder where the original method is saved open by default.

Step	Action
11	**a.** Select the desired target folder, **b.** type a new method name in the ***Name*** box and **c.** click ***Save***.

Note: *The original method remains after the converted method is saved. However, the converted method will replace the original if you choose to save the converted method with the same name in the same folder as the original.*

Note: *The flow rate and/or pressure settings in the method will automatically be adjusted if the maximum flow rate and/or pressure values for the target system is exceeded after the conversion. The maximum settings for the target system will be used by default.*

Note: *In case the original method contains instructions that are not supported by the new system, this will be indicated in the method outline of the converted method as a red cross on the phase containing these instructions.*

To be able to run the method on the new system you need to replace or remove the invalid instructions in the ***Text Instructions*** *tab. Invalid instructions are indicated with red square symbols in the text instructions.*

You can also replace the phase with a predefined phase from the phase library.

The method after conversion

The conversion will adjust the following settings to the appropriate values for the selected new system:

- Gradient delay
- System wash volume (i.e. ***Fill system*** value)
- Volume for finalization of sample application (only applicable when ***Inject sample directly onto column*** and then ***Inject all sample using air sensor*** is selected)

The following settings may require manual adjustment after the conversion:

- Sample volume
- Fractionation volumes
- User defined volumes in System CIP and System preparation
- System related settings in text edited phases

  (However, column related settings will be scaled also in text edited phases, provided they have been defined as variables)

Converting a method for use in a different database

The following table describes the necessary steps to be performed if you wish to convert a method for use with another system type, in another database than where the original method was created.

Tip: *This procedure must be followed in order to convert methods from one stand-alone system to another stand-alone system.*

Stage	Description
1	Set up a new system in the target database. Use the same instrument configuration as the system for which the method was originally created. ***Tip:*** *This system is created for the conversion only, and should be set up inactivated.* Since the new system will not be used for anything other than the method conversion, the system name, IP address and serial number can be selected at random. For example: • ***System name***: `Method conversion system` • ***Instrument serial number***: `123456789` • ***Fixed IP address***: `10.1.1.1` Refer to UNICORN Administration and Technical Manual, *Define a new system* for more information on how to set up a new system.
2	Export the method from the original database.
3	Import the method into the target database, for use with the new, inactivated system.
4	Convert the method from the inactivated system to be used with the target system, as described in the applicable instruction above (i.e. with or without scaling of the column size).

3.6.7 Import and export methods

Introduction

UNICORN methods are stored internally in the UNICORN database. It is however possible to export entire methods or individual phases to a zip file on the local computer so that they can be imported again later into the same database installation, or imported into another database installation.

Alternatively methods or phases can be exported as plain text files or Excel files, which may be useful for documentation purposes.

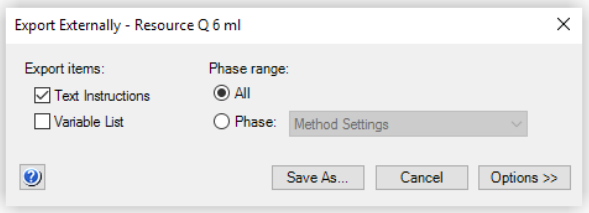
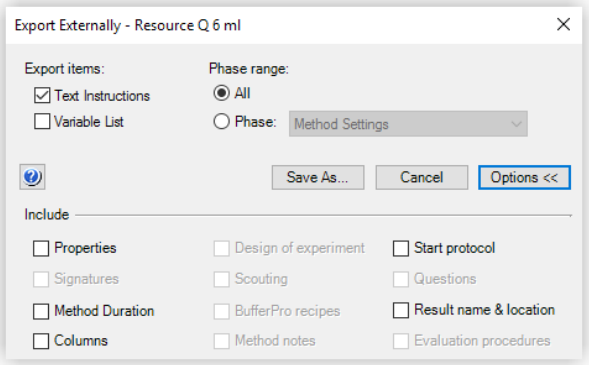
Export a phase or method to UNICORN

Follow the instruction to export a method or a phase for later import into UNICORN.

Step	Action
1	In the Method Navigator, select the method to be exported. ***Note:*** *Several methods in the same folder can be selected and exported at the same time. You can select several results at once by pressing and holding the* **Shift** *or* **Ctrl** *key when selecting.* *or* In the ***Phase Library*** pane, select the phase to be exported. ***Note:*** *Only* ***Personal*** *or* ***Global*** *phases may be exported.* ***Note:*** *Only single phases may be exported.*
2	On the ***File*** menu, click ***Export***, then click ***to UNICORN*** and then click ***Method(s)***. ***Note:*** *To export phases, right-click on the phase name and click* ***Export****.*
3	Choose a file name and location and click ***Save*** to save the zip file.

Export a method to a plain text file

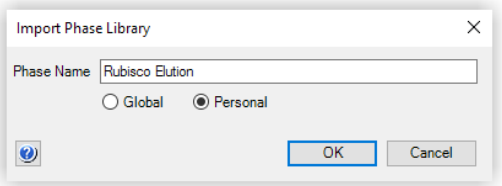
Follow the instructions to export a method as a plain text file or to Excel.

Step	Action
1	In the Method Navigator, select the method to be exported. It is only possible to export one method at a time to an external file.
2	On the ***File*** menu, click ***Export*** and then click ***Export Method Externally***. *Result:* The ***Export Externally*** dialog box opens. 
3	**a.** Choose which ***Export items*** to include by selecting the appropriate check boxes. **b.** Choose whether to include all phases or only a specific phase by clicking the appropriate ***Phase range*** setting.
4	To add further information to export, click ***Options***. *Result:* The ***Include*** options will be expanded. 
5	Select information to add to the text file by selecting the appropriate check boxes. ***Note:*** *Information that is not included in the method will appear dimmed and cannot be selected.*
6	To save the text file with the selected information included click ***Save As***.

Step	Action
	Result: The ***Export*** dialog box opens.
7	Select whether to save to an ***ASCII*** file or to an ***Excel*** file from the ***Save as type*** drop-down list.
8	Choose a file name and location and click ***Save*** to save the zip file.

Import a phase into UNICORN

Phases that have previously been exported as zip files can be imported back into UNICORN. Follow the instructions to import a phase.

Step	Action
1	On the ***File*** menu, click ***Import*** and then click ***Phase(s)***. *Result:* The ***Import*** dialog box opens.
2	Browse to the required zip file in the ***Import*** dialog box.
3	Open the file by selecting it and clicking ***Open***, or by double-clicking on the file name. *Result:* The ***Import Phase Library*** dialog box opens. 
4	Type a new ***Phase Name*** if required, and select whether the phase should be imported as ***Global*** or ***Personal***. ***Global*** phases are available to all users, ***Personal*** phases only to the currently logged-on user.
5	Click ***OK*** to import the phase.

Import a method into UNICORN

Methods that have previously been exported as zip files can be imported back into UNICORN. Plain text files or Excel files cannot be imported since there is no guarantee that they contain all the information UNICORN needs to recreate the method. Follow the instructions to import a method.

Step	Action
1	On the ***File*** menu click ***Import*** and then click ***Method(s)***. *Result:* The ***Import*** dialog box opens. ***Note:*** *When importing methods from UNICORN 5, choose format* ***UNICORN 5 Method Files*** *in the* ***Files of type*** *field.*
2	Browse to the required zip file in the ***Import*** dialog box. ***Note:*** *For methods imported from UNICORN 5, browse to the required method file.*
3	Open the file by selecting it and clicking ***Open***, or by double-clicking on the file name. ***Note:*** *Several methods in the same folder can be selected and imported at the same time. You can select several results at once by pressing and holding the* **Shift** *or* **Ctrl** *key when selecting.* Result: The ***Import Method(s)*** dialog box opens.
4	Browse to the required folder in the database and type in a new ***Name*** if necessary.
5	Select a ***System*** for the method from the drop-down menu.
6	Click ***Import*** to import the method. *Result:* The imported method will be opened in the ***Method Editor***. If several methods were chosen in the first step, repeat the procedure in the ***Import Method(s)*** dialog box for each method. ***Note:*** *If the imported method contains instructions that are not supported by the selected system, the phases containing these instructions will be marked with an error symbol.*

Import a method from UNICORN 5

Methods that have been created in UNICORN 5.11 and subsequent versions can be imported into UNICORN 7.10 in the same way as described above. However, some information will not be imported.

Note: *The Data collection utility can only be used if UNICORN 5 is installed on Windows XP.*

Changes in data after import

When methods are imported from UNICORN 5, there is some information that will not be included and some information that will be added. It is important to review the method after import, to verify that all essential information is there. The method can be manually edited, after the import. All functionality in UNICORN 7.10 is not available for imported methods, since necessary data might be missing.

Note: *Some instructions from a UNICORN 5 method are moved to the* ***Methods settings*** *phase and other instructions are moved to the* ***User Defined*** *phase when imported into UNICORN 7.10. Make sure to check both the* ***Phase properties*** *tab and* ***Text instructions*** *tab when searching for instructions and method settings.*

Affected information	Description
Installation test methods	Installation test methods that were generated by the method wizard in UNICORN 5 do not work in UNICORN 7.10. These methods must be regenerated by the method wizard in UNICORN 7.10.
Reference curves	Reference curves are not migrated since this functionality is not available in UNICORN 7.10.

Affected information	Description
Column data	**Delta pressure** A UNICORN 5 method with an included column can be migrated, but where UNICORN 5 uses ***Max pressure*** in the method, UNICORN 7.10 uses the terms ***Max delta column pressure*** and ***Max pre column pressure***. UNICORN 7.10 will set both ***Max delta column pressure*** and ***Max pre column pressure*** to be the ***Max pressure*** value from UNICORN 5 when a method without these parameters is imported. **Max flow rate** A UNICORN 5 method with an included column can be migrated, but ***Max flow rate*** is not mandatory in UNICORN 5 and may not have been set. When a method without ***Max flow rate*** is migrated UNICORN 7.10 will set the ***Max flow rate*** to be the same as the default flow rate for the column. **Column type** The UNICORN 7.10 Column list differs from the UNICORN 5 Column list. Any Column type included in a method is migrated and the values are not changed, but the Column type will not be imported to the UNICORN 7.10 Column list. The Column type of pre-packed columns from Cytiva from the UNICORN 5 Column list should be exchanged for the corresponding Column type in the UNICORN 7.10 Column list. Custom packed columns from the UNICORN 5 Column list have to be redefined in Column handling in UNICORN 7.10, in order to get the full use of the Column handling tool.

Affected information	Description
Evaluation procedures	If an evaluation procedure instruction is invalid in UNICORN 5 the instruction is removed from the method during migration. **Non-imported instructions** Some evaluation procedure instructions are not imported. A removed instruction is replaced in UNICORN 7.10 by a comment that informs the user that an instruction is missing. The following evaluation procedure instructions will not be imported: • ***CURVE_OPEN*** • ***PEAKTABLE_OPEN*** • ***EXPORT_DOC_ASCII*** • ***EXPORT_DOC_WKS*** • ***EXPORT_DOC_XLS*** • ***COPY_CHROM*** • ***CREATE_NEW_CHROM*** • ***OPEN_CHROM*** • ***DELETE_CHROM*** • ***RENAME_CHROM*** • ***SIMULATE_PEAK_FRAC*** • ***POOL_FRACTIONS*** • ***RET_MUL*** • ***EXPORT_DOC_400_xxx*** • ***EXPORT_NORMALISE_RETENTION*** • Export of curves in AIA • Quantitate functions • All instructions for exporting to WKS format **Changed instruction** The following evaluation procedure instructions are changed when imported: • The evaluation procedure instruction ***EXPORT_MULTI_CURVES_XLS*** is migrated from UNICORN 5 to the equivalent instruction ***EXPORT_MULTI_CURVES_CSV*** in UNICORN 7.10.

Affected information	Description
	• The instruction ***RUN_PROGRAM*** is implemented in UNICORN 7.10 but programs that have a user interface cannot be launched.
Report layouts	Report layouts are not migrated. If a non-imported report layout is referred to in an evaluation procedure, the layout will be replaced with the default layout.

4 Scouting

In this chapter

About this chapter

Scouting is used to repeat a series of method runs automatically using different settings or with predetermined changes in the values for one or more ***Variables***. A ***Scouting scheme*** is defined as part of the method. This chapter gives an overview of scouting and the scouting workflow and describes how to set up and edit a ***Scouting scheme***.

Scouting is ideal for relatively simple variable combinations. When designing experiments to analyse several variables at the same time, it is advantageous to use the ***Design of Experiments*** (***DoE***) tool. This tool applies statistical methods for generating scouting runs that provide the most information with as few runs as possible, thus economizing on time and sample amounts. For details on ***DoE***, see *Chapter 5 Design of Experiments, on page 98*.

4.1 Overview

Introduction

Scouting can be used to generate a series of method runs where one or more ***Variable*** parameters are varied in the same method. The resulting ***Scouting scheme*** is defined and saved in the method.

This section gives an overview of scouting and the scouting workflow.

When to use scouting

Some typical situations where scouting is useful are for instance when the objective is to:

- screen for the best column
- find the optimal pH
- test column capacity (sample volume)
- find the optimal flow rate for binding and elution
- optimize gradient length and slope
- optimize step gradients.

Scouting workflow overview

This is an overview of a scouting experiment:

Stage	Description
1	Create a method and decide appropriate run parameters (i.e.,variables) to be varied in the experiment. See *Chapter 3 Create and edit methods, on page 23* for information about how to create methods.
2	Set up a scouting scheme. This includes selecting variables, inserting runs/series of runs with different variable settings. To define new variables for a method, see *Section 10.2.4 Method variables, on page 277* for information.
3	Start and monitor the scouting run. This is performed in ***System Control***. See *UNICORN System Control Manual* for information. ***Note:*** *The **Start protocol** will only be displayed before the first run in the **Scouting** experiment.*

Stage	Description
4	Evaluate the results of the scouting run. This is performed in the ***Evaluation*** module. All results from ***Scouting*** runs performed at any one time are stored in the same folder. See *UNICORN Evaluation Manual* for information.

4.2 Set up and edit a Scouting scheme

Introduction

Any parameter can be scouted, provided that it can be defined as a variable in the method.

This section describes how to set up and edit a ***Scouting scheme***.

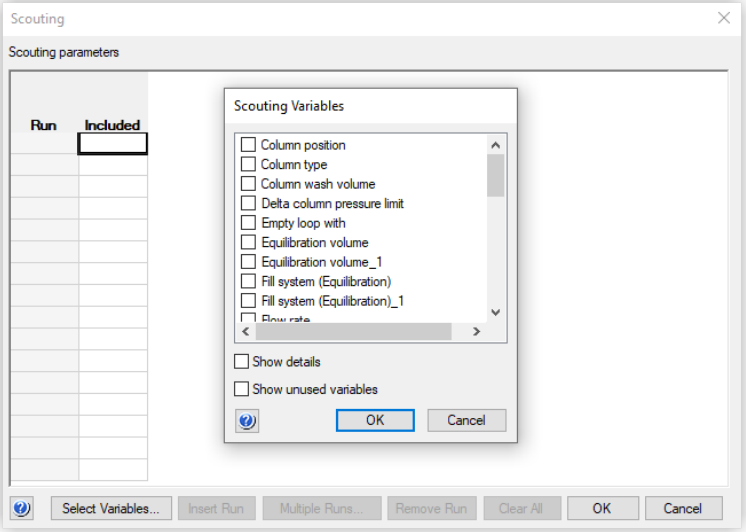
Set up a scouting scheme

Follow the instructions to set up a ***Scouting*** scheme where the flow rate is varied. In this example, the flow rate is varied between 0.5 and 3 ml/min.

Note: *The **Start protocol** will only be displayed before the first run in the **Scouting** experiment.*

Step	Action
1	Create a method and decide appropriate run parameters to be varied in the experiment. The run parameters to be varied should be defined as ***Variables*** in your method. See *Chapter 3 Create and edit methods, on page 23* for information about how to create methods. See *Section 10.2.4 Method variables, on page 277* for information about how to define new variables. ***Tip:*** *Many variables that can be used for scouting are already defined in either the Method Settings phase or the predefined phases. Note that some variables may be hidden or unused in the method. New variables often do not need to be defined.*
2	In the ***Method Editor***: • Click the ***Scouting*** button in the toolbar  or • Click ***Scouting*** on the ***Tools*** menu *Result:* The ***Scouting*** dialog box opens with the ***Scouting Variables*** dialog box displayed on top.

Step	Action



Note:

*When editing a scouting scheme, only the **Scouting** dialog box is displayed.*

3 **a.** In the ***Scouting Variables*** dialog box, select the appropriate variable(s) to be varied by selecting a the appropriate check box(es).

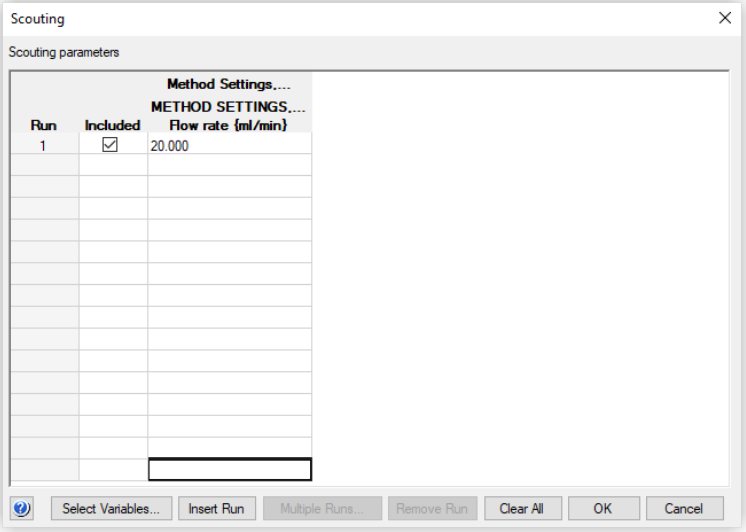
- Select the ***Show details*** check box if you want to display variables defined as detailed variables in your method. These are rarely used as scouting variables.
- Select the ***Show unused variables*** check box if you want to display variables currently not used in the method.

b. Click ***OK***.

Result:

The ***Scouting*** dialog box is updated with the selected variable(s) and their default value(s).

Step	Action



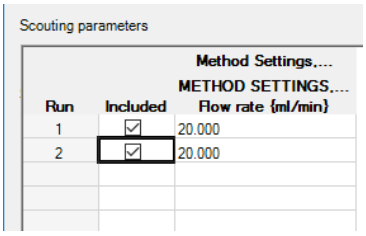
It is possible to insert runs one by one (see step 4)
or insert multiple runs at once (see step 5).

4 To insert runs one by one:

a. In the ***Scouting*** dialog box, select a row in the ***Scouting parameters*** table and click ***Insert Run***.

Result:

A new row is added below the selected run.
The variable value from the selected row is copied to the new run. Each chosen variable is displayed in a separate column.



Step	Action

b. In this example, click in the ***Flow rate {ml/min}*** column for the appropriate run and edit the flow rate value.

Note:

*Changing variable values in the scouting scheme does not change the values in the **Variable List** in the **Duration and Variables** dialog or in the text instructions. The actual variable values used for each run in the scouting scheme are saved in the result file. To change the default values, the variable values must be edited in the **Phase Properties** tab.*

c. Repeat until all runs are included using the correct variable values.

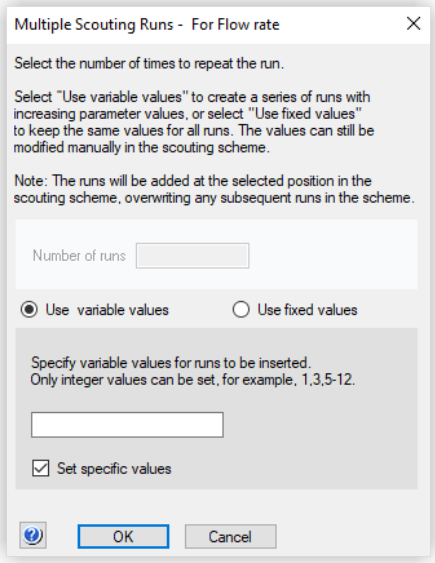
Note:

The scouting scheme can also be edited just prior to starting the method run in the Start Protocol. Here variable values can be changed and individual runs included or excluded.

5 To insert multiple runs or a series of runs, click in the appropriate variable column in the ***Scouting parameters*** table and click ***Multiple Runs***.

Result:

The ***Multiple Scouting Runs*** dialog box opens.



6 In the ***Multiple Scouting Runs*** dialog box, follow these steps to insert a series with increasing parameter values:

a. Enter the number of runs to insert in the scouting scheme in the ***Number of runs*** text box. In this example, 6.

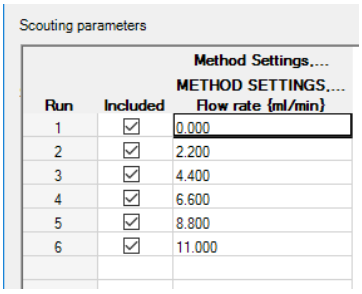
Step	Action

b. Enter ***Start value:*** and ***Step by:***. In this example, 0 and 2.2.

c. Click ***OK***.

Result:

The ***Scouting parameters*** table is updated.

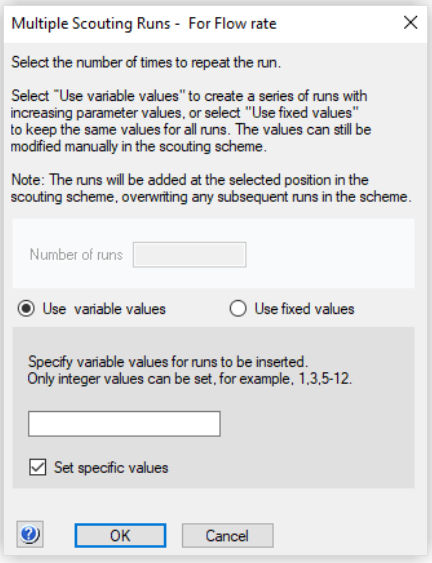


7 Alternatively, to enter either consecutive or non-consecutive integer values:

a. Select the ***Set specific values*** box in the ***Multiple Scounting Runs*** dialog box.

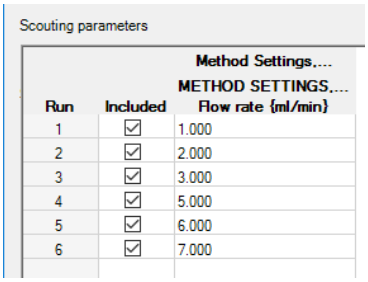
Result:

The following alternative ***Multiple Scouting Runs*** dialog box for the selected variable opens.



b. Enter the appropriate range, for example: 1-3,5-7

c. Click ***OK***.

Step	Action
	Result: The ***Scouting parameters*** table is updated. 
8	The ***Multiple Scouting Runs*** dialog box can also be used to enter multiple runs in the ***Scouting scheme***, while keeping the parameter values fixed. To do this: **a.** Select ***Use fixed values***. **b.** Enter the number of runs to insert in the scouting scheme in the ***Number of runs*** text box. **c.** Click ***OK***.
9	Click ***OK*** in the ***Scouting*** dialog box to save the scouting scheme.
10	Save the method.

Scouting parameters

Run	Included	Method Settings,... METHOD SETTINGS,... Flow rate (ml/min)
1	☑	1.000
2	☑	2.000
3	☑	3.000
4	☑	5.000
5	☑	6.000
6	☑	7.000

Add, delete, or edit variables in the Scouting scheme

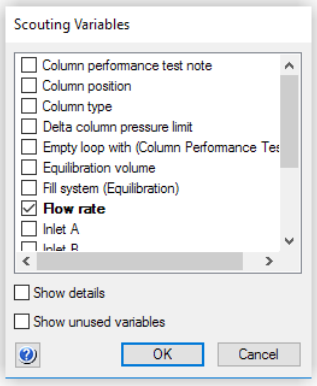
Follow the instructions to add, delete or edit variables in the ***Scouting scheme***.

Step	Action
1	Open the ***Scouting scheme*** (see *Set up a scouting scheme, on page 89*).
2	To add or delete variables in the ***Scouting scheme***, click ***Select Variables*** in the ***Scouting*** dialog box.

Step	Action

Result:

The ***Scouting Variables*** dialog box opens.



3

- To add a variable to the ***Scouting scheme***, select the appropriate check box in front of the variable.
- To delete a variable from the ***Scouting scheme***, clear the check box in front of the variable.

If you cannot find the appropriate variable:

- Select the ***Show details*** check box to display variables defined as detailed variables in your method.
- Select the ***Show unused variables*** check box to display variables currently not used in the method.

To define a new variable, see *Section 10.2.4 Method variables, on page 277* for information.

Click ***OK***.

*Result:*The ***Scouting parameters*** table is updated with the changes.

4

To edit a variable value for a run:

a. Select the appropriate row and the variable value cell in the ***Scouting parameters*** table.

b. Type a new value for the variable.

Result:

The variable value is updated.

Note:

Changing variable values in the scouting scheme does not change the values in the ***Variable List*** *in the* ***Duration and Variables*** *dialog box or in the text instructions.*

Step	Action
	The actual variable values used for each run in the scouting scheme are saved in the result file. *To change the default values, the variable values must be edited in the* ***Phase Properties*** *tab.*
5	Click ***OK***. *Result:* The ***Scouting parameters*** table is updated with the changes.
6	Add new scouting runs to the scouting scheme as required.
7	Click ***OK*** in the ***Scouting*** dialog box to save the scouting scheme.
8	Save the method.

Add/delete runs in the Scouting scheme

Follow the instructions to add runs and series of runs to the ***Scouting scheme*** and how to delete runs.

Step	Action
1	Open the ***Scouting scheme*** (see *Set up a scouting scheme, on page 89*).
2	• To insert a run after an existing run: Select the appropriate row in the ***Scouting parameters*** table and click ***Insert Run***. *Result:*A new row is added below the selected run to the ***Scouting parameters*** table. The variable value from the selected row is copied to the new run. Edit the variable value as appropriate. • To insert multiple runs: - Click in the appropriate variable column in the ***Scouting parameters*** table and click ***Multiple Runs***. - Set up a series or enter several identical runs in the ***Multiple Scouting Runs*** dialog box and click ***OK*** (see *Set up a scouting scheme, on page 89*). *Result:* The new set of runs are inserted in the ***Scouting scheme*** with the values provided. • To delete runs from the Scouting scheme: - Select the row(s) in the ***Scouting parameters*** table and click ***Remove Run***. *Result:*The selected runs are removed from the ***Scouting scheme***.

Step	Action
	or - Click ***Clear All***. *Result:*All runs are removed from the ***Scouting scheme***. No scouting will be performed when starting the run. • To exclude a run from being used in the ***Scouting*** experiment but keep it in the ***Scouting scheme***: Clear the ***Included*** check box in front of the appropriate run.
3	Click ***OK*** in the ***Scouting*** dialog box to save the changes to the scouting scheme.
4	Save the method.

5 Design of Experiments

About this chapter

This chapter gives a brief overview of ***Design of Experiments*** and describes some basic terms and concepts used in the ***Design of Experiments*** tool in UNICORN. It also describes how to set up an experimental design plan using the ***Design of Experiments*** (***DoE***) tool in the ***Method Editor*** and how to evaluate the results of the runs in the ***Evaluation*** module.

Design of Experiments is only available for some systems.

In this chapter

5.1 Introduction to Design of Experiments

Introduction

This section gives a brief introduction to the basic terms and concepts used in ***Design of Experiments*** (***DoE***).

What is Design of Experiments?

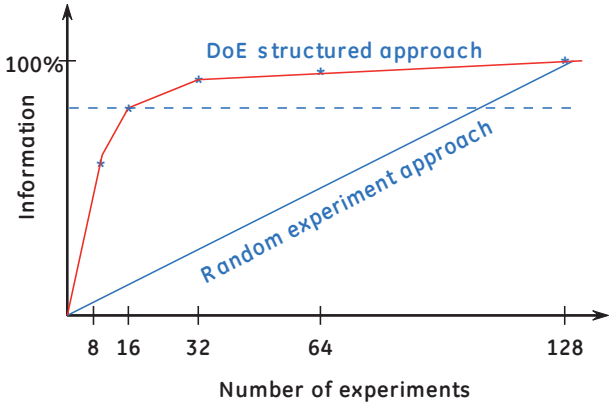
Design of Experiments is a way to systematically vary several parameters simultaneously to obtain as much information about a process with as few experiments as possible.

Why use DoE?

Maximize the amount of information using a minimum number of runs

When trying to find optimal conditions for a process to obtain the best results, it is usually not possible to perform all experiments needed due to time or cost using a random experiment approach. The number of runs to be performed needs to be minimized at the same time as the information from the runs are maximized. ***DoE*** facilitates this by using a systematic approach for experimental set-up and statistical modelling for the results.

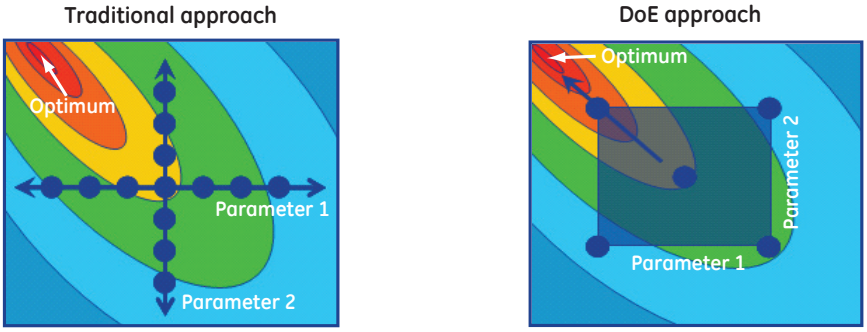
For example, it could be enough to obtain 80% information about a process. This level of information can be reached using a significantly lower number of experiments using ***DoE*** than using a random experiment approach as illustrated in the figure below.



Estimate parameter interactions

In the simplest traditional approach to optimization experiments, one parameter is varied while all others are fixed. In this way optimal values may be found for each parameter. Using this approach, interaction effects between parameters might be missed that could lead to better optimization of a process.

In the DoE approach, process parameters are allowed to vary simultaneously, thus allowing the effect of each parameter individually as well as the combined effect of parameters to be estimated. Each parameter may have an optimum, but when combining the parameters, values may be found that together give a new optimum, even better than the optima for the separate parameters. The illustration below shows the different approaches in a graphical way.



Obtain reliable maps of the system

Experiments are performed to assess the conditions for best processes or to obtain the product characteristics required. In order to make the necessary decisions we need tools or ways to make this as intuitive and easy as possible. In the evaluation of ***DoE*** results, different plots are created from the model. Decision making becomes more reliable when using tools that benefit from the created model. This "map" of the process helps to decide on, for example, how to progress, or whether the process is already optimized. Is the process robust? What experiments can be performed to verify the process?

DoE in UNICORN

In UNICORN, ***DoE*** is used to systematically create an optimized set of experiments to be run. Depending on the objective and the number of parameters, a suitable design is suggested. An experimental plan is presented and a ***Scouting scheme*** is generated as a result from ***DoE*** containing the method runs to be performed. When the runs have been performed, the results can be analyzed in the ***Evaluation*** module. A model is created and a number of plots are generated to aid evaluation of the results. The model can be used to predict responses for new parameter settings and to optimize the parameter settings for a desired combination of responses (e.g., optimize the response combination "minimize the level of impurities and maximize the yield").

Factors and responses

The table below lists the definitions of the ***DoE*** terms factors and responses and how to use them in UNICORN.

Term	Definition	In UNICORN
Factor	The different parameters that may affect the process to be run. Factors may be either quantitative or qualitative. • Quantitative factors are characterized by being found on a continuous scale for example, pH, flow rate and conductivity. • Qualitative factors are characterized by being discrete (discontinuous), for example, Column type, resins type and buffer substance.	The factors are connected to a variable in the method. For example, the factor pH may be connected to the variable BufferPro pH. In predefined methods, most useful parameters are already defined as variables. ***Note:*** *To be able to vary a value for a process parameter in the method it must be defined as a variable.* Low and high values are entered for the quantitative factors. The factors will be varied within this range.
Response	The output parameter(s) from a process. For example, capacity, yield and purity.	When evaluating the ***DoE*** runs, the measured response values for each experiment are entered in UNICORN.

DoE design

The design is the setup of experiments with different combinations of factor settings resulting in a minimum number of experiments to be run to obtain as much information as possible.

UNICORN suggest a suitable design to be used in the experiment based on the:

- number of factors
- type of factors (quantitative or qualitative)
- experimental objective (screening, optimization or robustness testing)

There are different objectives and design types included in the ***DoE*** tool in UNICORN. See the following blocks for more information about design objectives and design types.

Design objectives

The table below describes the different design objectives:

Design objective	Used when you want to...
Screening	Determine which factors are important in your process and the appropriate ranges for these factors.
Optimization	Find the optimal factor settings for your process, that is, factor settings that give the desired responses.

Design objective	Used when you want to...
Robustness Testing	Determine the process robustness by making minor adjustments of the factor settings and see if the responses are within the set specification limits. If the responses do not vary significantly due to the factor changes, the process is considered to be robust.

Example of how to use DoE for different objectives

To obtain maximum amount of protein after purification of a sample using a minimum number of runs, use ***DoE*** to find:

- important parameters (e.g., pH, conductivity) and the appropriate parameter ranges affecting this process
- the optimal parameter settings and any dependencies (interaction) between the parameters affecting the response of the product or process (e.g., yield or impurity level)

When the parameters affecting the process as well as their settings have been determined it is appropriate to test if the process is robust, that is, not affected by minor changes in the parameter settings. Neither the parameter settings selected or their interactions should affect the process if the process is to be considered robust.

A specific ***DoE*** setup is required for each step (i.e., screening for parameters or parameter settings, for optimization of parameter settings and for robustness testing). Each setup is a balance between the amount of information obtainable and the number of experiments that can be afforded. The process can be iterated and the initial screening results from one ***DoE*** can logically be used as input for the next ***DoE*** and so on.

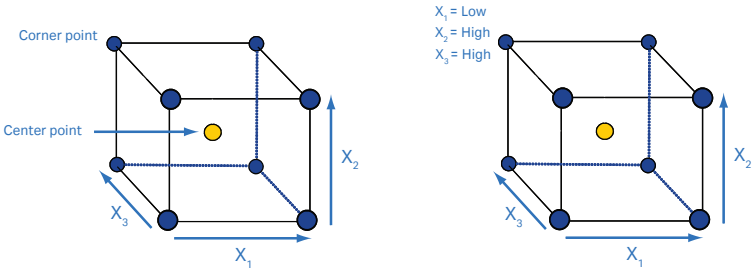
Design types

A design can be graphically illustrated by a box. The design box in the following examples illustrates designs where three different factors (X_1, X_2 and X_3) are varied simultaneously. Each corner point is the experiment for a specific combination of the settings of the three factors (e.g., low value for X_1 and X_2 and high value for X_3). The center point is the experiment where the different factors have the closest distance to all other factor settings, that is, the mean value.

The corner points are used to assess factor interaction effects. The center points are used to estimate the pure error and detect curvature. See *Model, on page 105* for detailed information about the terms interaction, curvature and pure error.

Illustration of the design box

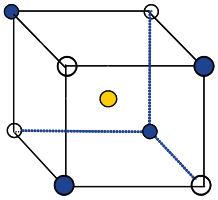
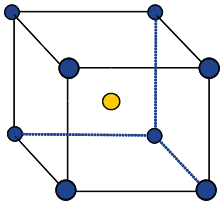
The illustration below (to the left) shows a design box with corner- and center points for the different factors X_1, X_2 and X_3. The illustration below (to the right) shows the factor values for one of the corner points. The arrow directions along the box edges denote the parameter change from low to high.

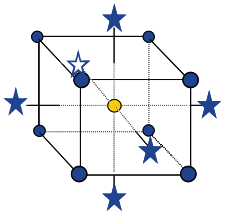


Different designs in UNICORN

There are several types of design available in UNICORN.

The table below describes three design variants illustrated by the design box. Different designs are used based on the objective and the experiment setup. The center point experiments are always included in all designs.

Design type	Description
Fractional Factorial 	In the Fractional Factorial design, some of the corner point experiments are excluded as illustrated to the left (i.e., the white circles). This design will not give as much information as when using all corner point experiments but by excluding the corner points as shown in the illustration, the information loss is minimized. Information about which factors are important (main effects) and some information about factor interaction effects are obtained. This design type is good to use when you need to obtain information about the parameter settings and reduce the number of factors in your experiment before optimization. Fractional designs are suggested when performing: • ***Screening*** (because the information provided using this design is often enough to find the factors affecting the process) • ***Robustness Testing*** (because then optimal factor settings have already been found and only minor changes in the factor settings are studied) Less experiments are needed compared to using Full Factorial and Optimization designs.
Full Factorial 	The Full Factorial design uses all corner point experiments. This design is often suggested when performing ***Screening***. Information about which factors are important (main effects) and more information about factor interaction effects are obtained.

Design type	Description
Optimization designs 	For optimization studies and especially if curvatures are detected, the Full Factorial design can be extended with additional experiments outside the box, called star point experiments. The design box illustrates the experimental space (the low and high values of the different factors) and experiments outside the box, that is, star points enhancing the detection capability for curvatures. The default star point distance (CCC design, see below) can be edited in UNICORN. This design results in a higher number of experiments but more information can be obtained. It may be suggested when performing ***Optimization*** but often not as the first choice because a higher number of experiments must be performed. Information about which factors are important (main effects), information about factor interaction effects and curvature are obtained. See also *Model, on page 105* for information.

Designs supported by UNICORN

The table below briefly describes the design types that are supported by UNICORN.

Design type	Description
L-designs	L-designs are a type of Fractional Factorial design. Different variants are available in UNICORN. The table below gives a short description of the designs.

L-design	Description
L9	Fractional design at three levels for up to four factors. You can estimate quadratic terms but not all interactions.
L18	Fractional design with one factor at two levels and with up to 7 factors at three levels.
L27	Fractional design at three levels for up to 13 factors. You can estimate square terms but not all interactions.
L36	Fractional design at three levels for up to 13 factors. You can estimate square terms but not all interactions.

L-designs are useful when performing ***Screening*** or ***Robustness Testing***.

Design type	Description
Plackett Burman	***Plackett Burman*** is a type of Fractional Factorial design with a lower resolution. This means that it is not possible to estimate any two-factor interactions using this design. ***Plackett Burman*** designs are useful when performing ***Screening*** or ***Robustness Testing***.
Rechtschaffner	***Rechtschaffner*** is a saturated fraction of the 2^n and 3^n factorial designs that supports all the first order interactions and quadratic terms. ***Rechtschaffner*** is useful when performing ***Optimization*** and you have at least three factors in your experimental plan.
Full Factorial 2 levels	***Full Factorial 2 levels*** is an orthogonal (balanced) design with all combinations of the factor levels. Main effects and all interactions are clear of each other (not confounded). Full Factorial 2 levels designs are useful when performing ***Screening*** or ***Robustness Testing***.
Full Factorial 3 levels	***Full Factorial 3 levels*** is a full factorial design with every factor varied at three levels. You can estimate the full quadratic model. ***Full Factorial 3 levels*** designs are useful when performing ***Screening***, ***Optimization*** or ***Robustness Testing***. They are however not the primary choice for ***Screening*** or ***Robustness Testing***.
CCC	The ***Central Composite Circumscribed*** (***CCC***) design is composed of a full or fractional factorial design and star points. ***CCC*** designs are useful when performing ***Optimization***.
CCF	The ***Central Composite Face*** (***CCF***) design is composed of a full or fractional factorial design and star points placed on the faces of the sides. ***CCF*** designs are useful when performing ***Optimization***.
Box Behnken	***Box Behnken*** is a three level Response Surface Modelling (RSM) design. All design points, except the center points, are located at the center of the edges of the hypercube, and are also on the surface of a sphere. You can estimate the full quadratic model. ***Box Behnken*** is useful when performing ***Optimization*** and you have at least three factors in your experimental plan.
Doehlert	***Doehlert*** is a RSM design constructed from regular simplexes. ***Doehlert*** designs are useful when performing ***Optimization***.

Model

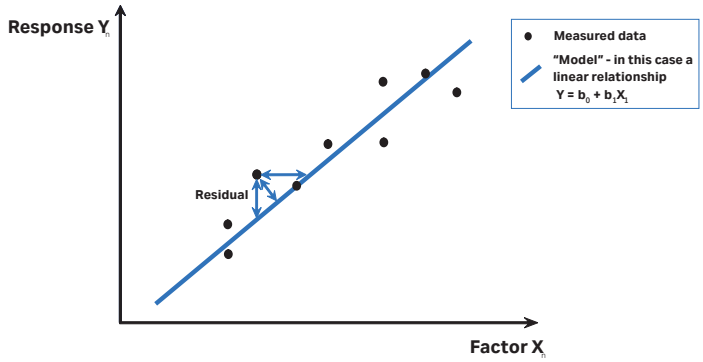
A model is created in the ***Evaluation*** module based on the response values measured or entered for each experiment in the ***DoE*** setup.

The model is a mathematical fit to all data (Multiple Linear Regression, MLR) and can be expressed as:

$\mathbf{Y_n = f(X_1, X_2, ..., X_n)}$

where **Y** is response and **X** is factor

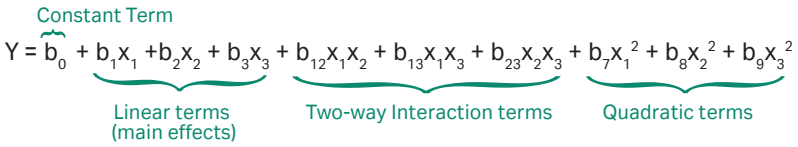
The model can be explained in a graphical way as shown in the illustration below. In this case the "model" is a linear relationship. The residual (error) between the measured data and theoretical data according to the model is minimized.



Model details

A more detailed description of the model is provided by the following formula as shown in the illustration below. The example is valid for three factors, X_1, X_2 and X_3 respectively.

Model



Description of the variables and constants in the model

Known variables (entered in UNICORN)

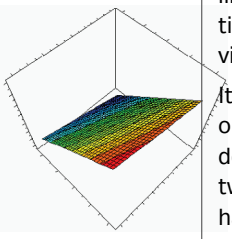
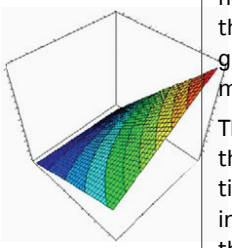
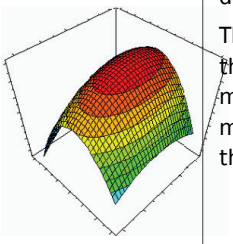
$x_1, x_2, \ldots\ldots, x_M$	are the input variables to the equation They correspond to the low and high factor values entered in the Method Editor.
$y_1, y_2, \ldots\ldots, y_M$	are the output variables to the equation They correspond to the measured response values entered in the Evaluation module.

Unknown constants determined by the model selected in UNICORN

b_0	is the unknown Constant Term. It is the response of Y when the main effects are 0.
$b_1, b_2, \ldots\ldots, b_M$	are unknown constants determined by the model in UNICORN.

As seen in the illustration above, the model can be divided into four parts, the Constant Term, the Linear Terms (main effects), the Two-way interaction terms and the Quadratic terms. The b-values are determined by the selected model. The Y-values are the response values that are entered in UNICORN.

The table below gives a brief description on how to interpret the different terms in the model.

Term	Graphical illustration	Description
Constant b_0	N/A	b_0 is the unknown constant term. It is the response of **Y** when the main effects are 0.
Linear (main effects) $b_1X_1 + b_2X_2 + b_3X_3$	Undistorted plane 	The main effects are described by the linear terms. In the graphical illustration, this part of the model can be viewed as an undistorted plane. It will give an overall idea of where the optimum for your process is but not details on how the sampling plane is twisted or which curvature the plane has. This part of the model usually gives sufficient information when the objective is screening or robustness testing. The fractional factorial designs will give enough input to create the linear part of the model.
Two-way interaction $b_{12}X_1X_2 + b_{13}X_1X_3 + b_{23}X_2X_3$	Twisted plane 	The two-way interaction terms describe how the effect of one factor depends on the level of a second factor. In the graphical illustration, this part of the model can be viewed as twisted plane. This part is added to the model when the objective is screening. The fractional and full factorial designs will give input to the two-way interaction part of the model.
Quadratic $b_7X_1^2 + b_8X_2^2 + b_9X_3^2$	Curved plane 	The curvature of the sampling plane is described by the quadratic terms. This part is added to the model when the objective is optimization. The optimization designs will give enough information to create the quadratic part of the model.

DoE workflow in UNICORN

The main steps when performing a ***Design of Experiments*** in UNICORN are:

Step	Action
1	**Create a method for your process to be screened, optimized or tested for robustness** This includes defining the appropriate variables (if not already defined) that should be connected to the factors in ***DoE***. See *Chapter 3 Create and edit methods, on page 23* for information about how to create a method.
2	**Set up an experimental design** This is performed in the ***Method Editor*** in the ***Design of Experiments*** tool. See *Section 5.2 Create an experimental design, on page 110* for more information.
3	**Perform the runs in the Scouting scheme generated from DoE** See UNICORN System Control Manual for information about starting scouting runs. ***Note:*** *The **Scouting scheme** generated from **DoE** does not normally need to be edited. If for some reason this is absolutely necessary, care must be taken so that the results can be used during evalulation of the **DoE** results.*
4	**Perform statistical evaluation of a DoE scouting** This is performed in the ***Evaluation*** module. See *Section 5.4 Evaluation of Design of Experiments, on page 133* for more information.

5.2 Create an experimental design

About this section

This section describes how to set up a ***DoE*** in the ***Method Editor***. A ***Scouting*** scheme is generated as a result containing the method runs to be run in ***System Control***.

In this section

5.2.1 Set up an experimental design

Introduction

This section describes how to set up a ***Design of Experiments*** in the ***Method Editor***.

Create a method

Create a method for the process to be optimized. The table below briefly describes how to create a method.

Step	Action
1	Create a method for the process to be optimized. See *Chapter 3 Create and edit methods, on page 23* for detailed information about how to create methods.
2	Decide which run parameters that should be screened for or optimized in the experiment. If the run parameters are not already defined as variables, define the parameters as variables to be able to vary the values i the ***DoE*** setup and to connect them to the appropriate factors. See *Section 10.2.4 Method variables, on page 277* for information about how to define new variables. ***Note:*** *In the **DoE** setup factors are connected to the variables in the method.*
3	Save the method.

Set up a new Design of Experiments

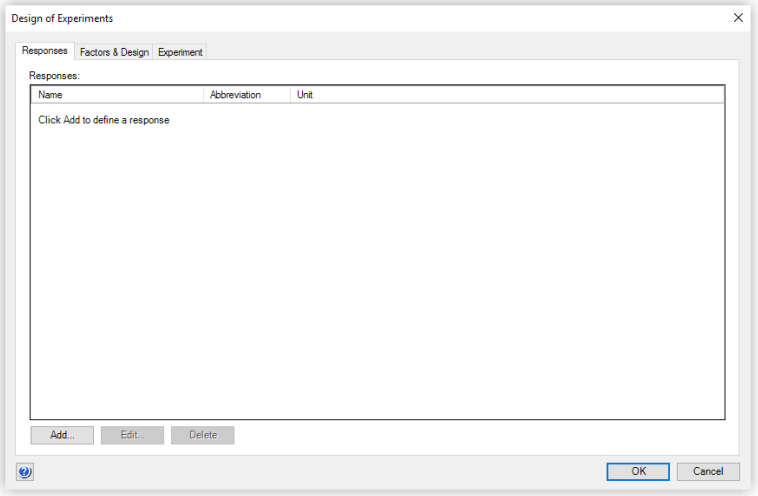
The table below describes how to set up a new ***Design of Experiments*** in the ***Method Editor***.

Step	Action
1	In the ***Method Editor***: • click the ***Design of Experiments*** icon in the ***Toolbar***  or • select ***Tools*** →***Design of Experiments***

Step	Action

Result:

The ***Design of Experiments*** dialog opens.

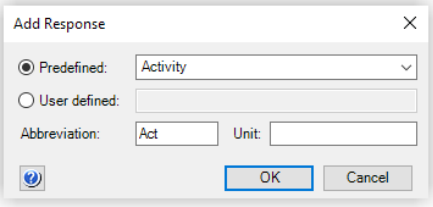


2 To add responses to the ***Design of Experiments***, click ***Add...***

Result:

The ***Add Response*** dialog opens.

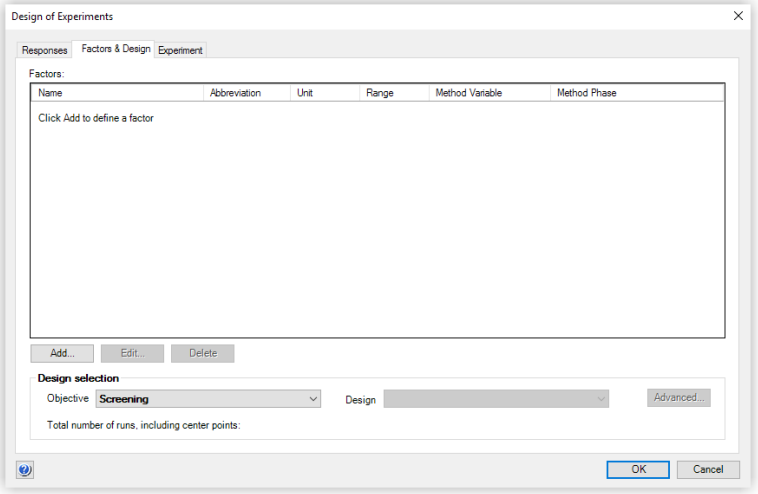
For detailed information about how to define and add a response, see *Section 5.2.2 Add responses and factors to an experimental design, on page 119*.



Note:

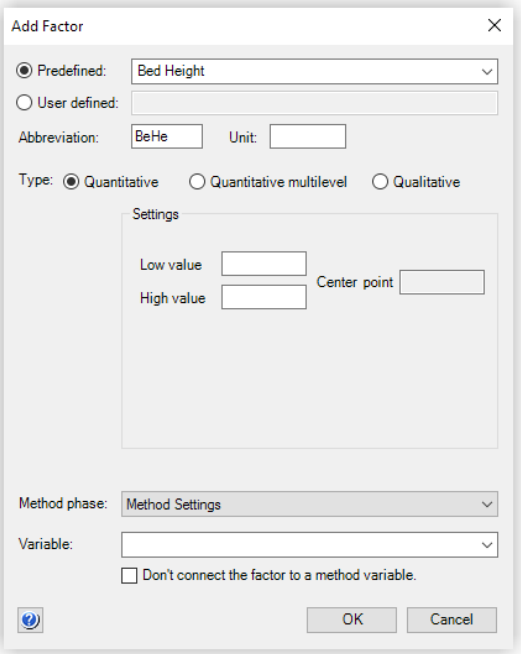
It is possible to add new responses to the experimental design in ***Evaluation****. These new responses will not be added to the method file as opposed to responses added in the* ***Method Editor****.*

Step	Action
3	When all responses are defined, select the ***Factors & Design*** tab.



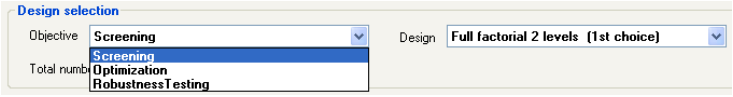
4	To add factors to the ***Design of Experiments***, click ***Add...*** *Result:* The ***Add Factor*** dialog opens. For detailed information about how to define and add a factor, see *Section 5.2.2 Add responses and factors to an experimental design, on page 119*.

Step	Action



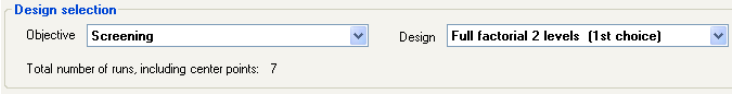
5 When all responses and factors have been defined, select the objective for the ***Design of Experiments***:

a. In the ***Design selection*** area in the ***Factors & Design*** tab, select the appropriate objective from the ***Objective*** drop-down list.



Result:

Depending on the selected objective, UNICORN suggests a suitable design to obtain sufficient resolution with as few experiments as possible in the ***Design*** list. The total number of runs are displayed for the suggested design.



Step	Action

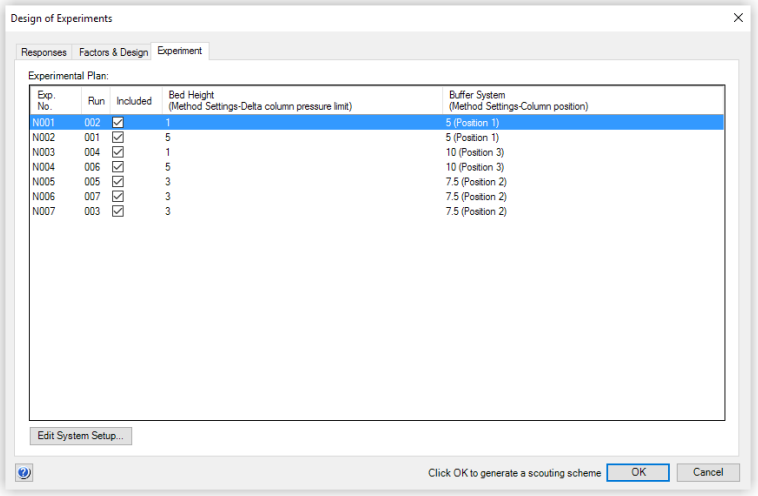
b. It is possible to select the ***2nd choice*** design in the ***Design*** drop-down list if appropriate. The ***2nd choice*** design usually either requires a higher number of runs to be performed, or the resolution of the design is lower.

For information about how to view details for the selected design and/or to change to another design than the 1st or 2nd choice designs, see *Section 5.2.3 Change design and design settings in a Design of Experiments setup, on page 124*.

6 Click the ***Experiment*** tab.

Result:

The ***Experimental Plan*** is displayed.

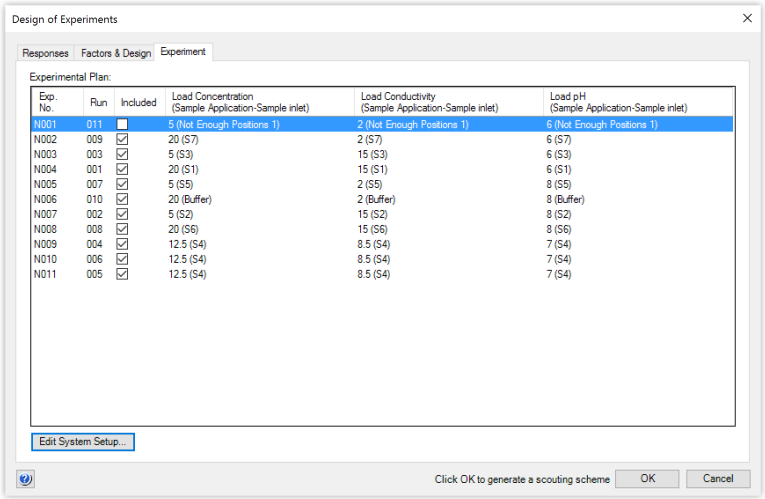


The ***Run*** column shows the run order for the optimized ***Scouting scheme*** that is generated from the ***DoE*** when clicking ***OK***.

Note:

*If any of the runs in the **Experimental Plan** are excluded, the results may not be reliable for use in the **DoE** evaluation.*

Step	Action
7	In some cases it may be necessary to divide the ***DoE*** runs into two or more scouting runs, for example if there are too few sample inlet valves. Some of the runs can be excluded the first time and run during further rounds. Limitations in the hardware are indicated in the ***Experimental Plan*** on the ***Experiment*** tab by the text ***Not Enough Positions***. For information about how to proceed when, for example, the available sample inlet positions are not sufficient, see *Section 5.2.4 Divide the DoE runs into several scouting runs, on page 127*.

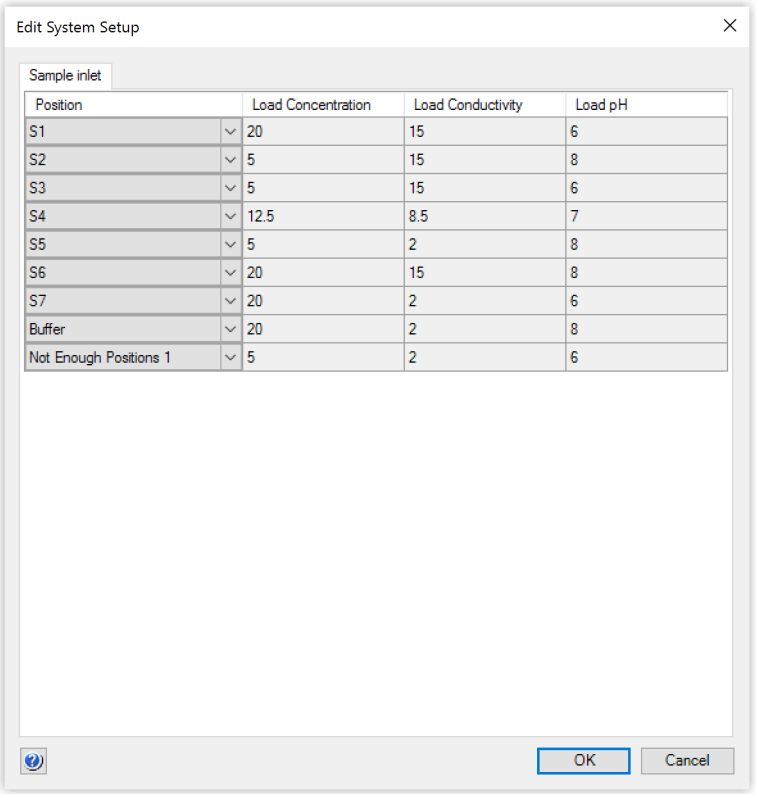


8	To view the system setup, click ***Edit System Setup...***.

Step	Action

Result:

The ***Edit System Setup*** dialog opens.



In this example the variable connected to both factors is the ***Sample inlet*** valve.

The ***Load pH*** and ***Load Conductivity*** values are set for each sample inlet.

To change the position for a certain combination of ***Load pH*** and ***Load Conductivity***:

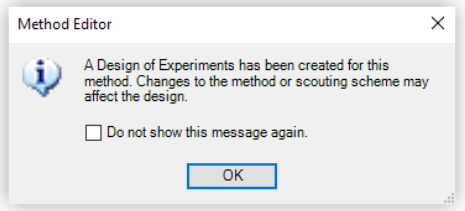
- Select the appropriate position in the corresponding ***Position*** drop-down list.

  Note:

  It is not possible to change to a position already used.

- Click ***OK***.

*Result:*The changes are saved and you return to the ***Design of Experiments*** dialog.

Step	Action
9	**a.** In the ***Design of Experiments*** dialog, click ***OK***. ***Result:*** The following dialog opens.  **b.** Click ***OK***. *Result:* A ***Scouting scheme*** is generated with the runs to be performed. The method is displayed in the ***Method Editor***. ***Note:*** *If you change the **Scouting scheme**, the **DoE** experimental plan is changed and the results may not be reliable for use in the **DoE** evaluation.*
10	Save the method including ***DoE***.

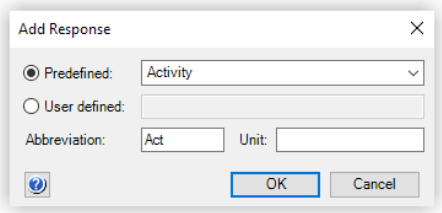
5.2.2 Add responses and factors to an experimental design

Introduction

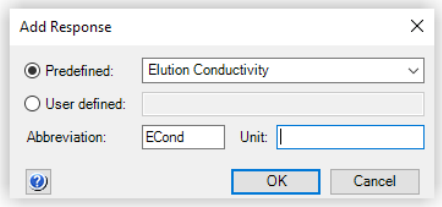
This section describes how to add responses and factors to the ***Design of Experiments*** setup in the ***Method Editor***.

Add responses

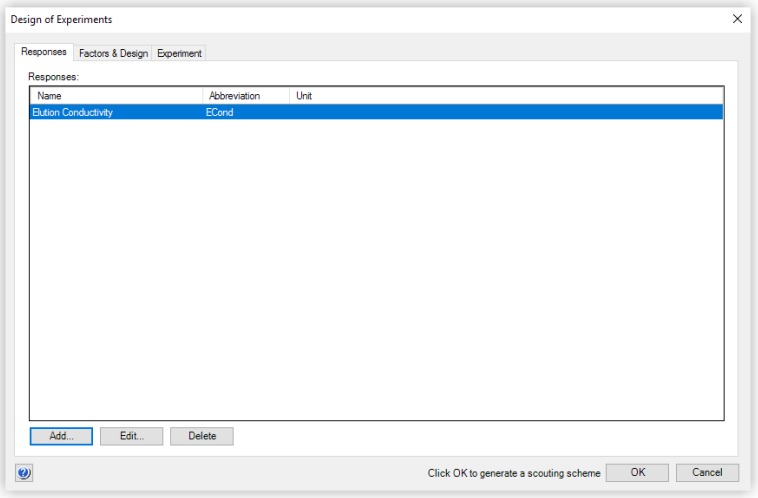
The table below describes how to add responses to an experimental design:

Step	Action
1	Select the ***Responses*** tab.
2	To add a response, click ***Add...*** *Result:* The ***Add Response*** dialog opens.  ***Note:*** *It is possible to add new responses to the experimental design in* ***Evaluation****. These new responses will not be added to the method file as opposed to responses added in the* ***Method Editor****.*
3	**a.** To add a predefined response: Select the response to be added in the ***Predefined*** drop-down list. **b.** To add a user defined response: Select ***User defined*** and type in your own response *Result:* ***Abbreviation***is automatically filled in.

Step	Action
	c. If applicable, enter unit for the response.



4	Click ***OK***. *Result:* The selected response is added to the ***Responses*** list in the ***Design of Experiments*** dialog.



Add factors

The table below describes how to add factors to a ***Design of Experiments***:

Step	Action
1	Select the ***Factors & Design*** tab.
2	To define factors, click ***Add...*** *Result:* The ***Add Factor*** dialog opens.

Step	Action

3 To add a predefined factor:

- Select the factor to be added in the ***Predefined*** drop-down list.

  Result: ***Abbreviation*** and the correct ***Type*** radio button is selected.
- If applicable, type in the ***Unit*** for the factor.

To add a user defined factor:

a. Select ***User defined*** and type in your own factor.

Result:

Abbreviation is automatically filled in.

b. If applicable, type in the ***Unit*** for the factor.

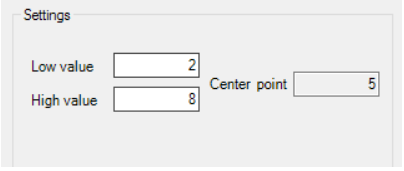
c. Select what kind of factor it is by selecting the appropriate ***Type*** radio button. The table below describes the different types of factors:

Type of factor	Description
Quantitative	Quantitative factors are process parameters that can be measured and have values on a continuous scale (e.g., flow rate and pH values)
Quantitative multilevel	To specify more than two levels for a factor, select the Quantitative multilevel type. For example, if your are performing an experiment at three different temperatures, 4°C, 10°C and 25°C.
Qualitative	Qualitative factors are discrete discontinuous process parameters or categorical data (e.g., Column type and type of salt used).

4 Enter settings for the selected factor:

a. ***Quantitative*** factors:

Enter a ***Low value*** and a ***High value*** for the factor. The center point is automatically calculated.

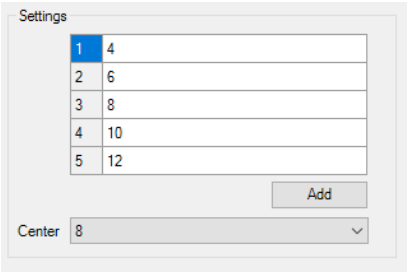


b. ***Quantitative multilevel*** factors:

- Enter the discrete values for the factor in the different rows.

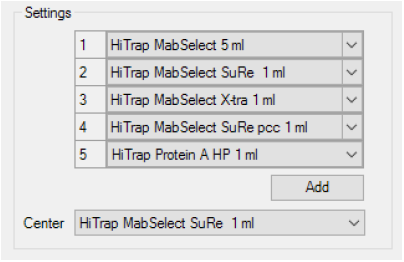
Step **Action**

- To add more rows, click the ***Add*** button.
- A center point is automatically selected. To select another center point, choose the appropriate one in the ***Center*** drop-down list.

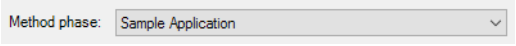


c. ***Qualitative*** factors:

- Select or type in the parameters in the different rows.
- To add more rows, click the ***Add*** button.
- A center point is automatically selected. To select another center point, choose the appropriate one in the ***Center*** drop-down list.

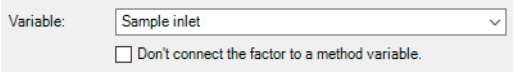


5 Select the phase to which the factor is connected in the ***Method phase*** drop-down list.



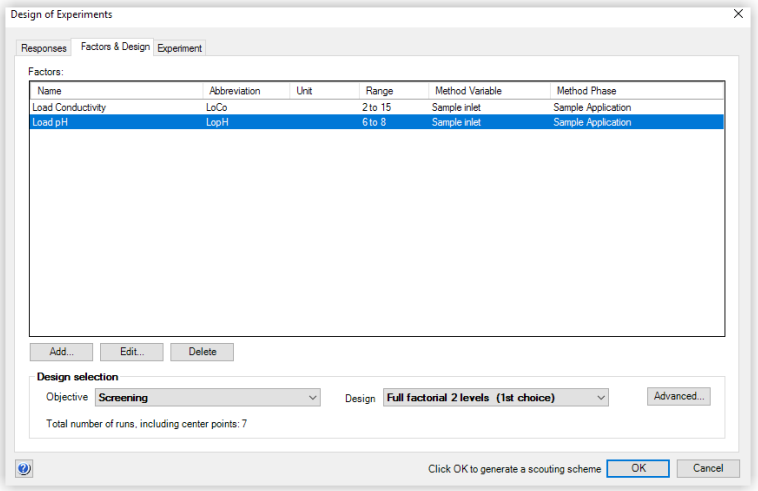
For example, if adding the predefined factor ***Load pH***, the pH at loading is controlled in the method phase ***Sample application***.

6 Select to which ***Variable*** the factor is connected in the ***Variable*** drop-down list.



For example, if adding the predefined factor ***Load pH***, the sample pH at loading is controlled by the ***Sample inlet*** valve position.

Step	Action
	Note: *Variables connected to factors will be included in the **Scouting** scheme that is generated when completing the **DoE** setup.*
7	If the factor is not connected to anything that can be controlled by UNICORN (e.g., if the experiment is performed in a cold room or in room temperature) check the box ***Don't connect the factor to a method variable***.
8	Click ***OK*** to add the factor to the ***Design of Experiments***. *Result:* The factor will be listed on the ***Factors & Design*** tab.



9	To add more factors, repeat this procedure.

5.2.3 Change design and design settings in a Design of Experiments setup

Change design in a Design of Experiments setup

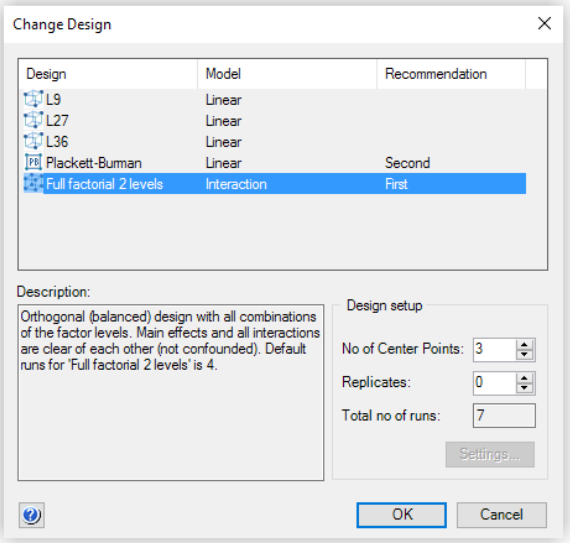
The design suggested by UNICORN can be changed to another design in the setup of the ***Design of Experiments***. The settings for a selected design can also be edited. The table below describes how to change the default design and design settings (i.e., the number of center points and replicates) in a ***Design of Experiments*** setup:

Step	Action
1	In the ***Factors & Design*** tab of the ***Design of Experiments*** dialog, the suggested design is displayed in ***Design*** drop-down list. 
2	Click the ***Advanced...*** button to: **a.** change to another design than the ***1st*** or ***2nd choice*** design available in the ***Design*** drop-down list (continue with step 3) *and/or* **b.** edit the settings for the currently selected design in the ***Design*** table (continue with step 4)

Step	Action

Result:

The ***Change Design*** dialog opens displaying the designs that may be used for the current experimental setup and selected objective.



3 To change to another design, select the appropriate design in the ***Design*** table.

Result:

The ***Description*** field shows a short description of the selected design.

For a description of which designs are supported by UNICORN and when they may be proposed, see *Designs supported by UNICORN, on page 104*.

4 The ***Design setup*** area shows the settings for the selected design.

a. Change the settings for ***No of Center Points*** and ***Replicates*** as appropriate.

The table below describes the different settings:

Setting	Description
No of Center Points	The ***No of Center Points*** means that the center point experiment will be run the selected number of times. It is recommended to use at least three center points to be able to estimate the pure error, that is, the variation in the measurements.

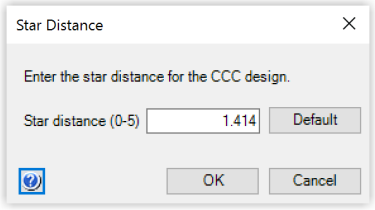
Step **Action**

Setting	Description
Replicates	***Replicates*** means that the whole experiments series (corner and center points) will be replicated the selected number of times.
Total no of runs	This field lists the total number of runs to be performed based on the number of center points and replicates.

b. The ***Settings...*** button is only active if a CCC design using star points is selected. To change the default star point distance in relation to the design box, click ***Settings...***.

Result:

The ***Star Distance*** dialog opens.



Change the ***Star distance*** as appropriate and click ***OK***. To return to the default value, click ***Default***.

5 In the ***Change Design*** dialog, click ***OK***.

Result:

Changes in the ***Change Design*** dialog are saved and the settings in the ***Design setup*** area in the ***Design of Experiments*** dialog are updated.

Note:

If additional variables have been defined in the scouting scheme for a previously saved DoE method, these will be lost and need to be redefined.

5.2.4 Divide the DoE runs into several scouting runs

Introduction

If hardware limitations exist, for example too few sample inlet valve positions are available for the number of runs to be performed, the ***DoE*** runs can be divided into several scouting runs. This section describes how to divide a DoE run into several smaller runs.

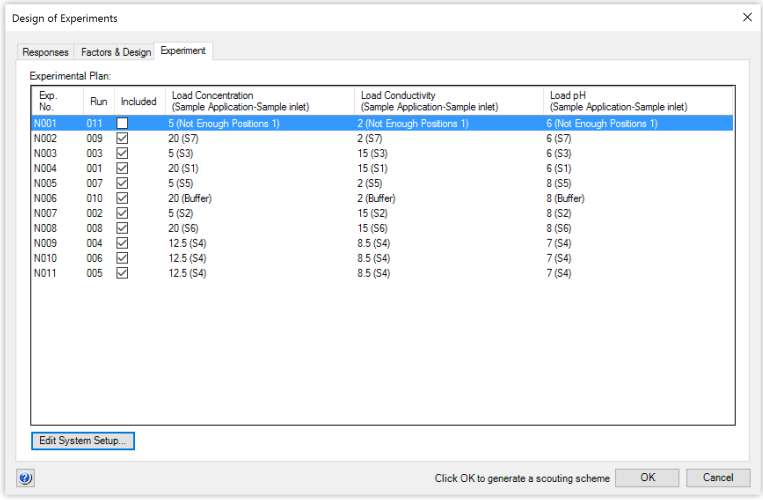
Divide DoE runs into several scouting runs directly in the Scouting scheme

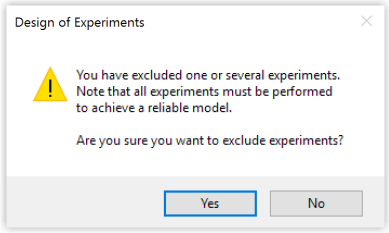
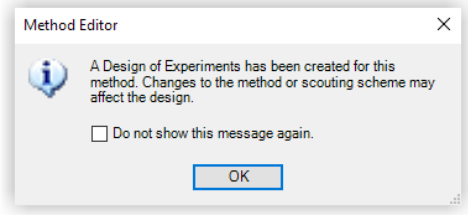
It is possible to include/exclude runs directly in the generated ***Scouting*** scheme and edit, for example, the sample inlet positions. However, for complex experimental plans it is recommended to create multiple ***DoE*** methods, each using the same design but with different sub-sets of scouting runs (see below). As long as the designs are identical, the results can then be merged for analysis.

Divide DoE runs into several scouting runs in the DoE setup

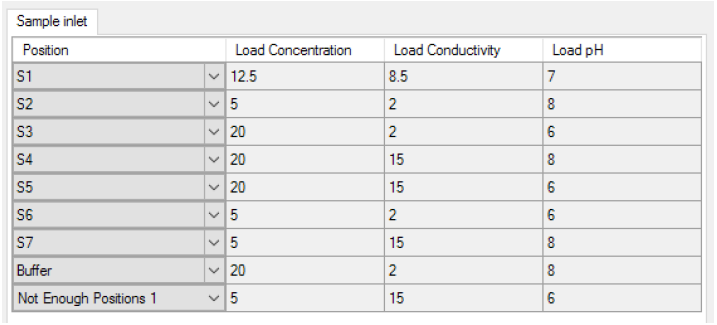
The table below describes how to identify hardware limitations in a DoE run.

Step	Action
1	In the ***Design of experiments*** dialog, select the ***Experiment*** tab. *Result:* The ***Experimental Plan*** is displayed.



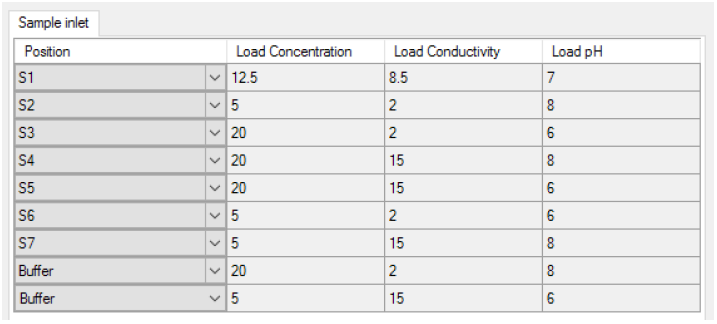
Step	Action
2	If limitations in the hardware exist this will be indicated in the ***Experimental Plan*** by the text ***Not Enough Positions*** for the run(s) in the ***Design of experiments*** dialog. These runs are also excluded from the ***Experimental Plan***.
3	Clear the ***Included*** box in front of the experiments to be excluded in the first set of runs. In the example below ***Run 009*** and ***Run 010*** are excluded from the first set of runs. ***Note:*** *Include at least one center point (runs 5, 9 and 11 in the above example) in each scouting run to have control of experimental variations.*
4	Click ***OK***. *Result:* The following warning dialog opens. 
5	Click ***Yes*** in the warning dialog. *Result:* The following message is displayed. 
6	Click ***OK*** and save the method.
7	To define the second set of runs, open the ***Design of experiments*** dialog again and select the ***Experiment*** tab. *Result:* The ***Experimental Plan*** is displayed.

Step	Action
8	Clear the ***Included*** boxes in front of **all** runs.
9	Click the ***Edit System Setup*** button. *Result:* The ***Edit System Setup*** dialog opens.


Sample inlet

Position	Load Concentration	Load Conductivity	Load pH
S1	12.5	8.5	7
S2	5	2	8
S3	20	2	6
S4	20	15	8
S5	20	15	6
S6	5	2	6
S7	5	15	8
Buffer	20	2	8
Not Enough Positions 1	5	15	6

10	Change the position for the inlet that did not have any position before to a valid position. The inlet that previously had the position must also be changed. **Example:** Change the position for, in this example, the sample inlet position indicated by ***Not Enough Positions 1*** to ***Buffer*** position.


Sample inlet

Position	Load Concentration	Load Conductivity	Load pH
S1	12.5	8.5	7
S2	5	2	8
S3	20	2	6
S4	20	15	8
S5	20	15	6
S6	5	2	6
S7	5	15	8
Buffer	20	2	8
Buffer	5	15	6

Then change the sample inlet position originally set to ***Buffer*** to ***Not Enough Positions 1***.

Step	Action

Sample inlet

Position	Load Concentration	Load Conductivity	Load pH
S1	12.5	8.5	7
S2	5	2	8
S3	20	2	6
S4	20	15	8
S5	20	15	6
S6	5	2	6
S7	5	15	8
Not Enough Positions 1	20	2	8
Buffer	5	15	6

The two inlet positions have been changed.

11 Click ***OK*** in the ***Edit System Setup*** dialog.

12 In the ***Experiment*** tab, check the boxes in front of the runs to be included in the second set of runs.

Note:

In the example shown here, one of the center points (run 9) is also included.

13 Click ***OK*** in the ***Design of experiments*** dialog.

Result:

A new ***Scouting scheme*** is generated. Click ***Yes*** and ***OK*** in any warning and messages dialog that appear.

14 Save the method with a **new** name.

Result:

The two scouting runs are ready to be run in sequence.

Note:

In this example, you must change samples in one of the sample inlets before starting the second scouting run. It is not possible to just create a method queue, start it and leave the system.

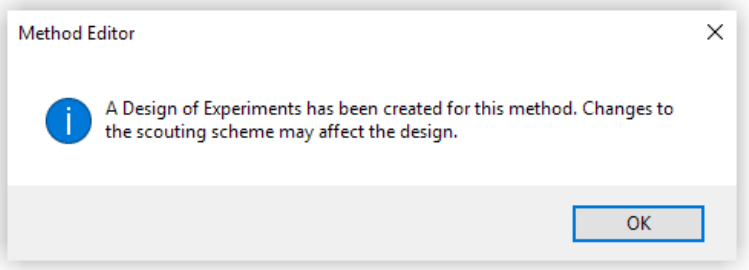
5.3 Run a scouting created with DoE

Introduction

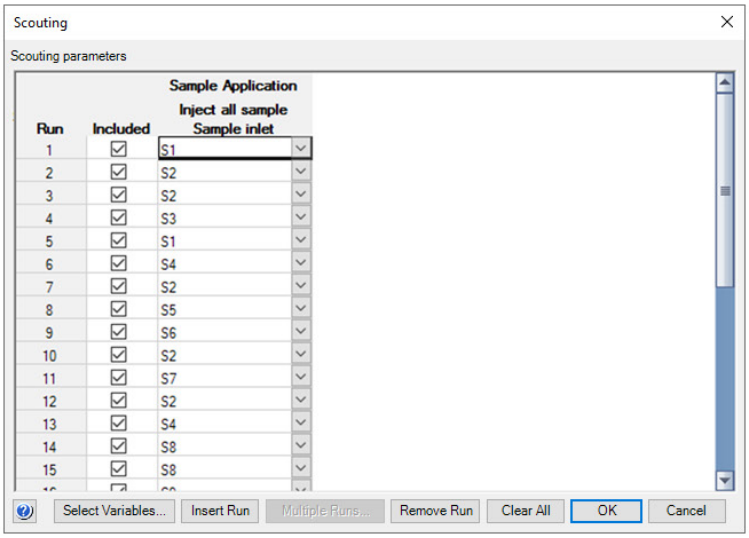
This section describes how to view the optimized ***Scouting*** scheme generated from ***DoE*** and how to print the method including ***DoE***. For information about how to start and monitor ***Scouting*** runs, see UNICORN System Control Manual.

View the Scouting scheme generated from DoE

When creating a ***Design of Experiments*** the final step is the generation of the optimized ***Scouting scheme***. The table below describes how to view the ***Scouting*** scheme generated from ***DoE***:

Step	Action
1	In the ***Method Editor***: • Click the ***Scouting*** icon  or • Select ***Tools*** →***Scouting***. *Result:* The following dialog box is displayed as a reminder of that a ***DoE*** has been created for the method. If you change the ***Scouting scheme***, the ***DoE*** experimental plan is changed and the results may not be reliable for use in the ***DoE*** evaluation. 
2	Click ***OK***. *Result:* The ***Scouting*** dialog box opens displaying the ***Scouting scheme*** where it is possible to view the ***Scouting*** runs to be performed.

Step	Action



3 Click ***Cancel*** or ***OK*** to close the ***Scouting*** dialog box.

Print method including DoE

Before starting the run, it is useful to print the method information to see, for example, which sample positions are used for the different runs. See *Section 3.6.3 Print a method, on page 63* for information about how to print the method.

5.4 Evaluation of Design of Experiments

About this section

This section describes how to perform statistical evaluation of a ***DoE*** scouting.

In this section

5.4.1 Workflow

Introduction

This section describes the workflow when evaluating a ***Design of Experiments*** scouting.

Workflow

The main steps when performing statistical evaluation of an experimental design are:

1. **Generate model**

   This includes evaluating single ***DoE*** runs, opening the ***DoE*** result, and entering response data. The software will then generate a model.

2. **Analyze and edit the model**

   This includes checking that the raw data is OK and performing a basic analysis of the model. The model may need refinement by removing insignificant terms, which should be done with care. Extended analysis can be performed for additional information.

3. **Use the model**

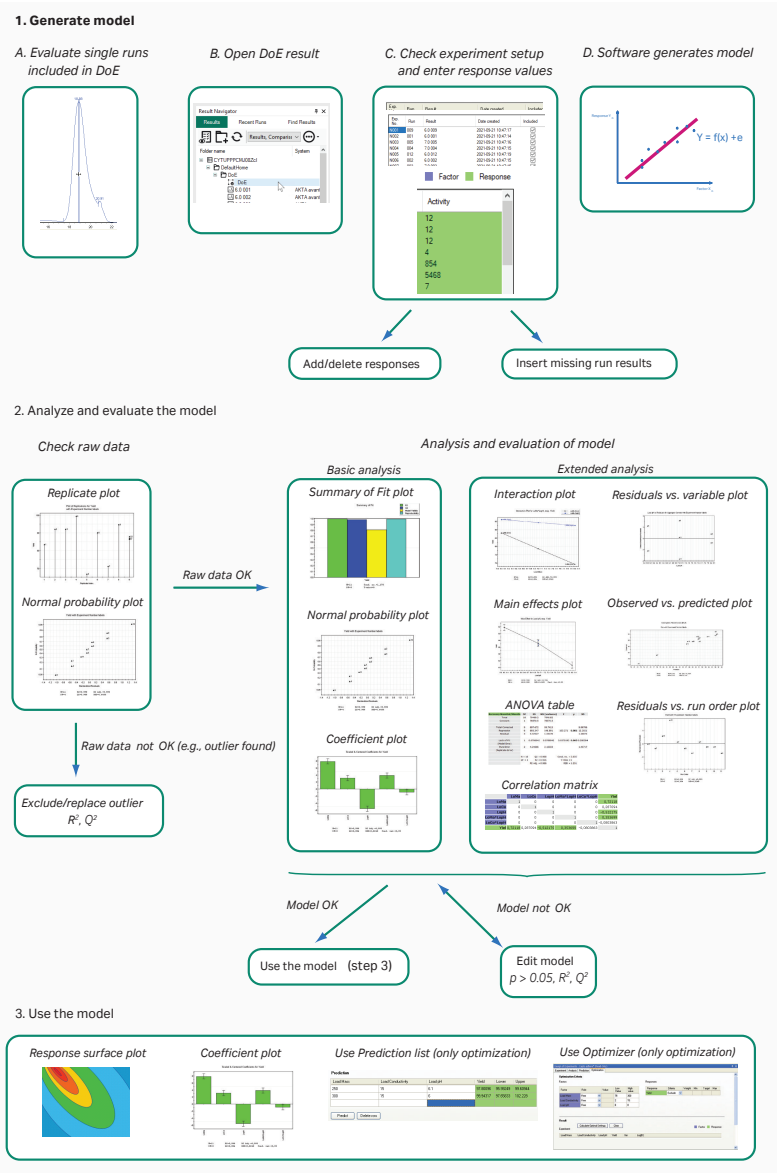
   This includes generating a response surface plot (including a sweep spot plot) as well as using the predictor and the optimizer.

   - Generate response surface plot to get a map of the experimental area and information about how to proceed with new experiments.
   - Use the predictor to predict response values based on entered factor settings (optimization experiments only).
   - Use the optimizer to optimize responses based on entered criteria for factors and responses (optimization experiments only), for example maximizing response 1 and minimizing response 2 while keeping factor 1 constant and allowing the other factors to vary within a defined range.

Basic and extended reports can also be created for the experiment.

Illustration of workflow

The illustration below shows a possible workflow for evaluating a ***Design of Experiments*** scouting:



5.4.2 Generate model

Introduction

This section describes how to open single ***DoE*** runs for evaluation, how to open ***DoE*** results and how to generate a model.

Evaluate the results of the single DoE runs

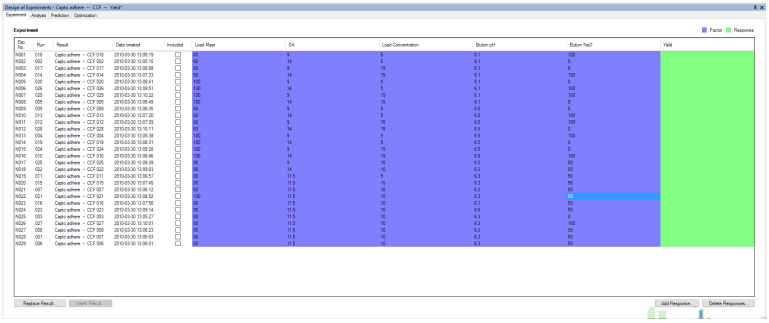
Before opening the ***DoE*** result in the ***Evaluation*** module, it is recommended to evaluate the single runs included in the ***Scouting*** run.

The table below describes how to open and evaluate single runs in the ***Evaluation*** module:

Step	Action
1	In the ***Evaluation*** module, click the ***Open Result Navigator*** icon in the ***Toolbar***.  *Result:* The ***Result Navigator*** is displayed.
2	Browse for the result and double-click the result name (single runs are indicated by the chromatogram icon). *Result:* The result of the run is opened and displayed in the ***Evaluation*** module.
3	Inspect the results visually and check that the runs have been performed as expected.
4	Evaluate the results for the run as appropriate. See *UNICORN Evaluation Manual* for information about how to perform evaluation.
5	Save any changes. ***Tip:*** *It is possible to have a* ***Scouting*** *run result open at the same time even if a* ***DoE*** *result is open.*
6	Repeat this procedures for all the runs included in the ***DoE*** result.

Open the DoE result

The table below describes how to open a ***DoE*** result:

Step	Action
1	In the ***Evaluation*** module, click the ***Open Result Navigator*** icon in the ***Toolbar***.  *Result:* The ***Result Navigator*** is displayed.
2	Browse for the ***DoE*** result and double-click the result name (***DoE*** results are indicated by the design box icon). *Result:* The ***Design of Experiments*** box opens displaying the ***DoE*** scouting run. 

Generate model

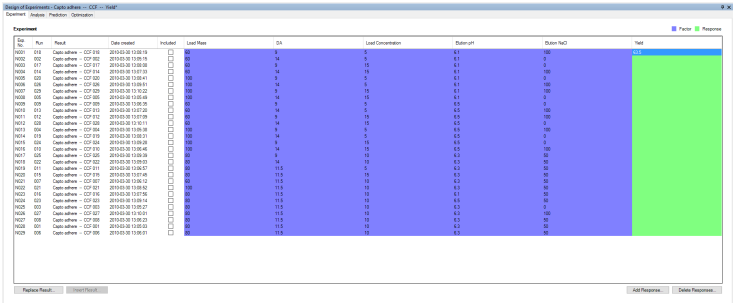
The table below describes how to generate a model for the ***DoE*** result:

Step	Action
1	Responses defined in the method appear in the result in the ***Design of Experiments*** box. To add or delete responses to the experiment, use the ***Add Response...*** and ***Delete Response...*** buttons. See *Add responses, on page 138* and *Delete responses, on page 139* for more information.
2	To enter response data: **a.** Click in a response cell for the appropriate response and experiment, and type in the data (the ***Yield*** column in the example shown here).

Step	Action

Result:

The data is entered in the cell.



Tip:

Response data can be obtained from:

- *external measurements (e.g., biological activity)*
- *peak data from UNICORN (e.g., HETP tests or resolution)*

b. Repeat this procedure for all experiments and responses.

3 Select the runs to be included in the calculations for generating the model by checking the ***Included*** box for the appropriate runs (usually all runs).

To insert a new run or to replace a failed run with a new run use the ***Replace Result...*** and ***Insert Result...*** buttons. See *Replace run results, on page 141* and *Insert new runs, on page 140* for more information about inserting and replacing runs.

Tip:

*Instead of replacing a failed run with a new run, the run can be excluded from the model calculations by clearing the **Included** box in front of the appropriate run. This will however often result in some loss of information.*

4 Click the ***Analysis*** tab.

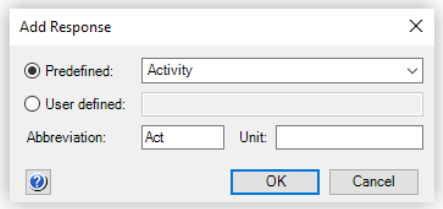
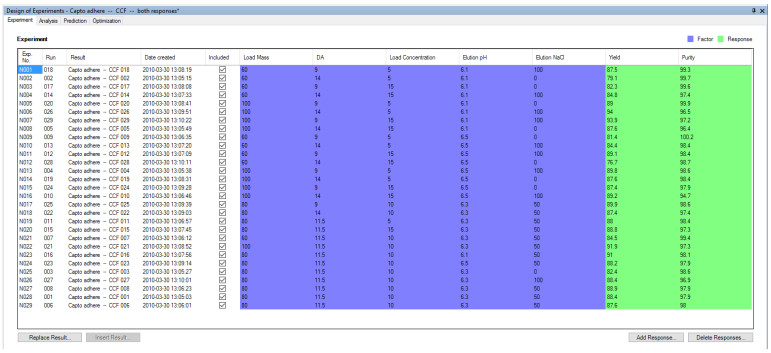
Result:

A model is fitted to the entered data. For information about how to analyze the model, see *Section 5.4.3 Analyze and evaluate the model - basic analysis, on page 143*.

Add responses

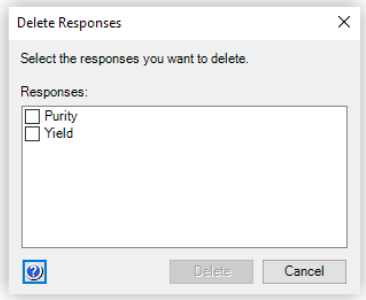
Note: *Responses added in the **Evaluation** module will not be included in the original method.*

The table below describes how to add a new response to the experiment:

Step	Action
1	In the ***Design of Experiments*** tab in the ***Evaluation*** module, click the ***Add Response...*** button. *Result:* The ***Add Response*** dialog box opens. 
2	Select the response to be added in the ***Predefined*** drop-down list or define your own response by selecting ***User defined*** and type in your own response. ***Note:*** ***Abbreviation*** *is automatically filled in.*
3	Enter the ***Unit*** for the response, if appropriate.
4	Click ***OK***. *Result:* The response is added to the ***DoE*** experiment. 

Delete responses

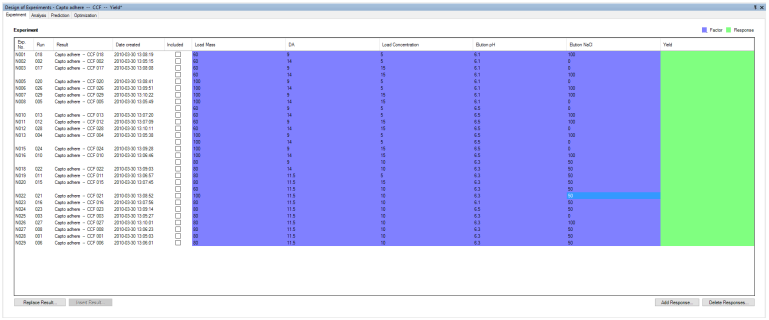
The table below describes how to delete a response from the experiment:

Step	Action
1	In the ***Design of Experiments*** tab in the ***Evaluation*** module, click the ***Delete Response...*** button. *Result:* The ***Delete Responses*** dialog box opens. 
2	Check the box in front of the response to be deleted and click ***Delete***. *Result:* The response is deleted from the experiment.

Insert new runs

Runs can be inserted if there are runs missing in the experiment. This can be the case if the ***DoE*** run has been divided into two scouting runs. In that case there will be two ***DoE*** results.

The table below describes how to add a missing run result to the experiment:

Step	Action
1	If runs are missing in the experiment, the rows for the missing runs are blank. 
2	To insert the missing runs, click the ***Insert Result...*** button.

Step	Action
	Result: The ***Insert Result*** dialog box opens.
3	Browse and select the run(s) that should be inserted. ***Tip:*** *The run order number is found at the end of the result name. This number is the same as in the* ***Run*** *column. This makes it easier to locate the runs to be inserted.*
4	Click ***OK***. *Result:* The runs are inserted in the experiment.
5	If a run does not match any of the missing runs in the experiment an error message will be displayed. Repeat from step 2 to insert the correct run. ***Note:*** *The run to be inserted must have the appropriate factor settings.*

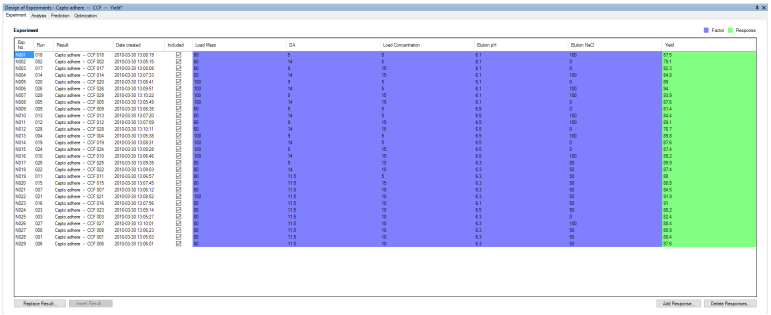
Replace run results

Runs can be replaced with a new run if, for example, the run has failed.

Note: *If a run has failed, there is always a risk that experimental conditions that cannot be controlled may have affected the result (e.g., temperature in the lab, different batches of buffer preparation etc.) Therefore, it is not always a good idea to replace a failed run with a new one. Rerunning a center point experiment will help in keeping track of uncontrolled variations.*

The table below describes how to replace a run result:

Step	Action
1	In the ***Experiment*** tab of the ***Design of Experiments*** box, select the result to be replaced in the ***Result*** column in the ***Experiment*** table.



Step	Action
2	Click the ***Replace Result...*** button. *Result:* The ***Replace Result*** dialog box opens.
3	Browse and select the run that should replace the selected run.
4	Click ***OK***. *Result:* The new run is listed in the ***Experiment*** table.
5	If the run does not match the run to be replaced an error message will be displayed. Repeat from step 2 to insert the correct run. ***Note:*** *The run to be inserted must have the appropriate factor settings.*

5.4.3 Analyze and evaluate the model - basic analysis

Introduction

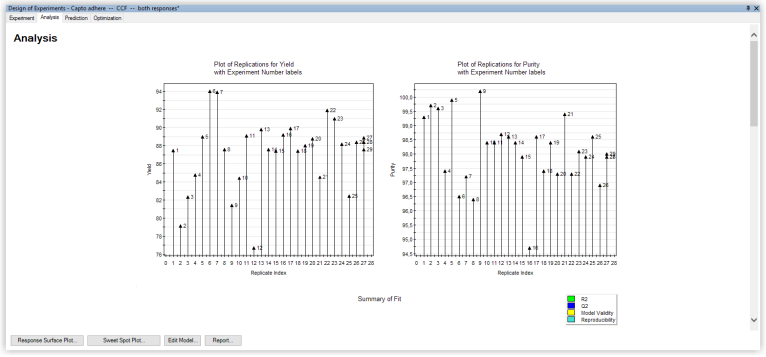
This section describes how to perform basic analysis of the model and how to evaluate the model.

Check the raw data

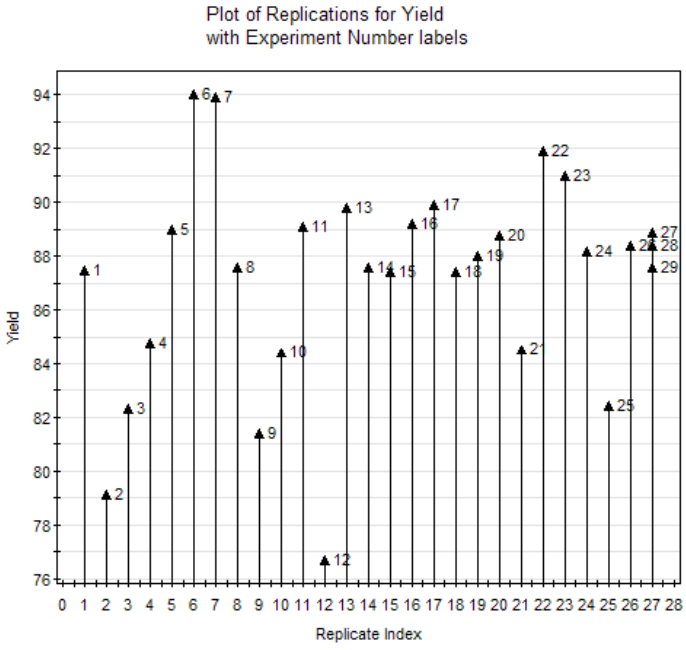
Before starting to analyze the model, the raw data must be checked to ensure that the correct conclusions can be drawn in the analysis and evaluation of the model.

The table below describes how to perform some initial checks that the raw data is OK:

Step	Action
1	Select the ***Analysis*** tab, if not already selected. *Result:* The ***Analysis*** tab opens showing 4 plots for each response: the Replicate plot, the Summary of fit plot, the Coefficient plot and the Normal probability plot of residuals. To be able to see all plots use the vertical scroll bar.



Step	Action
2	For each response, look at the replicate plot. This plot displays the variation in the response for replicated experiments and the variation among the replicates in relation to the variation across the entire design ("reproducibility").



Each arrow in the plot represents an experiment.

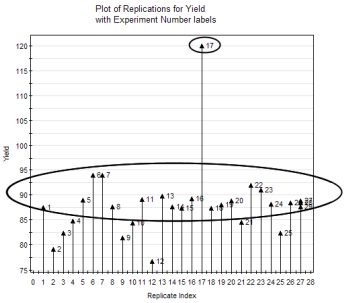
In a good replicate plot (as shown in the example above), the replicate runs should show as small a variation as possible (experiments ***27***, ***28*** and ***29***).

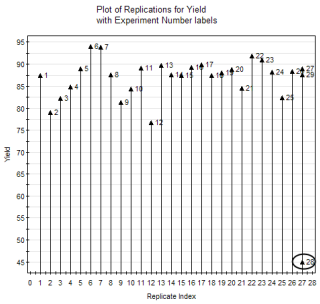
There should normally be some variation across the dataset of non-replicate experiments. However a single experiment should not deviate dramatically from the rest.

Note:

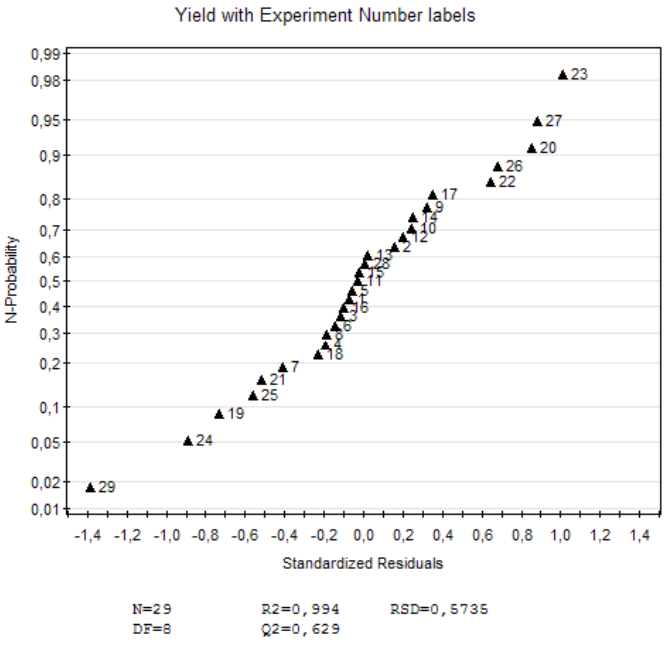
When a robustness test has been performed, variations in the data should instead be as small as possible.

Step	Action
3	The replicate plots can also be used to identify outliers. If, for example, a single experiment deviates a lot from the rest of the experiments (see plot to the left below), or if a replicate deviates a lot from the rest of the replicates (see plot to the right below), this run could be an outlier.





Step	Action
4	Look at the normal probability plot of residuals. The residuals, or minimized errors between the measured data and the theoretical data calculated according to the model, should normally be distributed as shown in the diagram below.



Step	Action
	In a normal distribution of residuals for a good model, the experiments should be distributed close to a straight line and also should lie within a ***Standardized Residuals*** range of -4 to +4 SD (standard deviations). Single experiments that deviate from this may be outliers. A non-linear distribution of experiments may also indicate the presence of insignificant missing terms, for example curvature of the model. See *Analyze and interpret the model - basic analysis, on page 147* and *Section 5.4.4 Analyze and evaluate the model - extended analysis, on page 154*.
5	If the raw data is OK, continue with the basic analysis of the model described in *Analyze and interpret the model - basic analysis, on page 147*.
6	If outliers are detected, try to identify why. The table below gives a few examples of why outliers may be detected. You may also look at the plots in the extended analysis to get more information about the experiment.

Why outlier?	What to do	See...
Bad replicates	Check the individual result, and that correct response values have been entered. If the run has failed, consider performing new experiments and replace the run.	*Replace run results, on page 141* for information about how to replace a run result.
	The run may also be excluded from the experiment setup. Results that are true outliers should be excluded.	*Generate model, on page 137* for information about how to exclude a run from the experiment
Deviating experiments	Check that the correct response values have been entered. Check the individual result. Consider performing new experiments to verify the deviation. If the results are indeed valid, the model may be inappropriate for the area.	See *Section 5.4.2 Generate model, on page 136* for how to check entered response data.

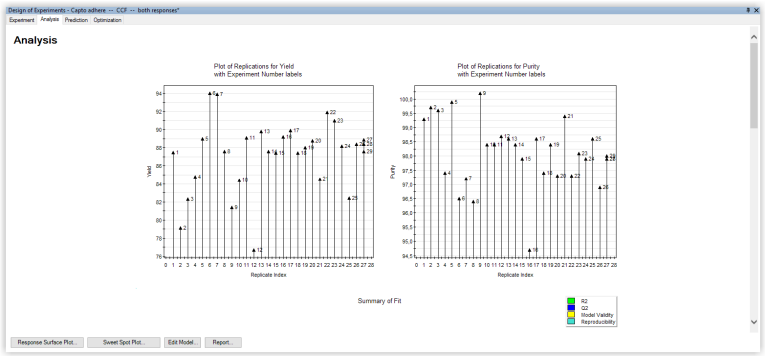
Analyze and interpret the model - basic analysis

Before you can use the model and draw conclusions from it, the model needs to be analyzed to investigate if the model gives a good reflection of the experiment data.

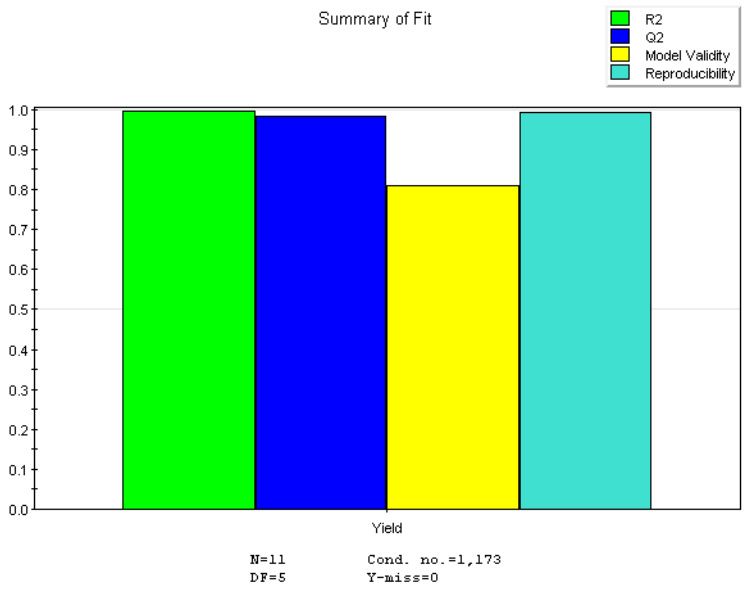
Note: *The plots must be analyzed for each response. A model can be good for one response but not for another. In some cases a good model cannot be obtained when several responses are included in the same model. In this case, try fitting an individual model for each response separately.*

The table below describes how to perform a basic analysis of the model:

Step	Action
1	Select the ***Analysis*** tab, if not already selected. *Result:* The ***Analysis*** tab opens showing four plots for each response: the Replicate plot, the Summary of Fit plot, the Coefficient plot and the Normal probability plot of residuals. To be able to see all plots use the vertical scroll bar.



Step	Action
2	Scroll down to display the ***Summary of Fit*** plot. If the experiment contains several responses, the plot will contain the four bars shown below for each response.



The bars in the plot describe different statistical calculations for each response, measuring how good the model is. It is the contribution of all values that together indicate if the model is good. A good model has high values for all parameters as seen in the plot above. The table below gives a description of the parameters:

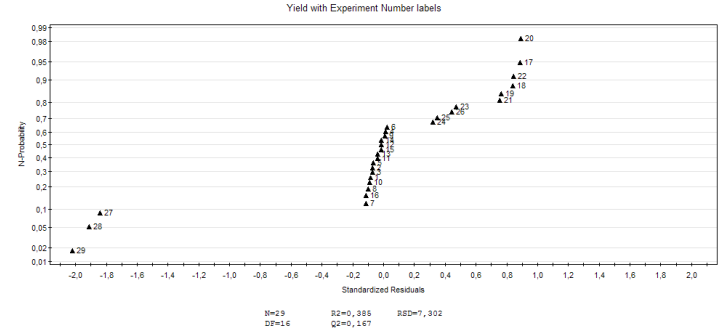
Step **Action**

Coefficient value for...	Description
R2	R^2 describes how well the model fits the current data. It can vary between 0 and 1, where 1 equals a perfect model and 0 corresponds to no model at all. A high R^2-value is necessary for a good model but not sufficient on its own. A value of 0.75 indicates a rough but stable and useful model. ***Note:*** ***R2 Adj*** *is the fraction of variations in the response data that is explained by the model, adjusted for degrees of freedom.* *R^2 does not take into account degrees of freedom.*
Q2	Q^2 describes how well the model will predict new data. It can vary between -∞ and 1. The higher Q^2-value, the better indicator of how well the model will predict new data. Q^2>0.5 is good and Q^2>0.9 is excellent. Q^2 is a better indicator of the usefulness of the model than R^2. ***Note:*** *R^2 should* ***not*** *exceed Q^2 by more than 0.2-0.3 for a good model.*
Model Validity	Model validity is only available if replicated experiments have been performed. A model validity>0.25 indicates a good model. A model validity<0.25 indicates a significant "lack of fit", that is the model error is significantly larger than the pure error (reproducibility).
Reproducibility	A reproducibility<0.5 indicates that there is a large pure error and poor control of the experimental setup (high noise level).

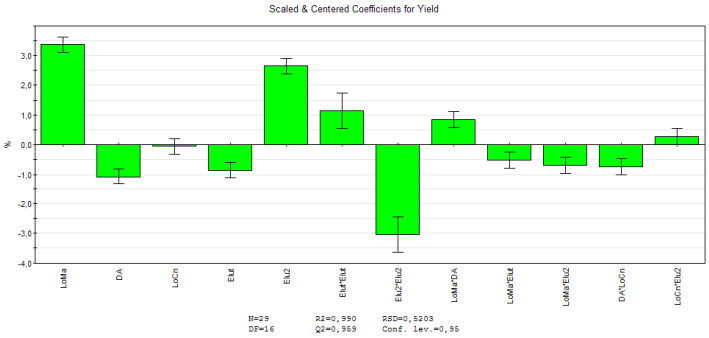
If the ***Summary of Fit*** plot does not look good, there may be several reasons for this. The table below lists a few.

Step **Action**

Summary of Fit plot value	Possible cause	What to do
Low Q^2 and model validity	Non-significant two-way interactions may be present	Look at the coefficient plot (see step 3) and the Interaction plot to see if there are interaction effects.
	Curvature in the model. Is there a need of adding quadratic terms to the model?	Look at the ***Residual vs. variable*** plot (see *Residuals versus variables plot, on page 154*) and the ***ANOVA*** table (see *ANOVA table, on page 158*) to see if these also indicate curvature in the model. If you suspect curvature, try adding a quadratic term to the model.
Model with moderate R^2 (~0.6) and Q^2 (~0.4)	Important factors may be missing. Are there uncontrolled factors that may affect the experiment?	If needed, perform more experiments.
The model is good for one response but not the other	It might be difficult to fit the same model to all responses.	Consider dividing the experiment in two or more to be able to fit one model/response.

Step	Action
3	Look at the normal probability plot of residuals. If the model describes the experimental data well, the experiments should be distributed close to a straight line, and lie within a ***Standardized Residuals*** range of -4 to +4 SD (standard deviations). See *Check the raw data, on page 143*. If the center points (points 27, 28 and 29 in the illustration below) are not linearly distributed, this may indicate curvature in the model rather than true outliers. A low Q^2, model validity and significant lack of fit may also indicate curvature.  If you suspect curvature, try adding a quadratic term to the model. See *Section 5.4.5 Edit the model, on page 161* for more information.
4	Look at the coefficient plot for each response. The coefficient plot can be used to see which factors that affect your response, in which way they affect the response(s) and if there are any non-significant terms in the model. **a.** the main effects, that is, the loading conditions for mass, pH and conductivity (LoMa, LoCo and LopH) **b.** the two-way interaction effects for LoMa/LopH and LoCo/LopH In the example below the following terms have been included in the model: ***Note:*** *If an optimization design (CCC or CCF) has been used, quadratic terms for the model will also be included in the coefficient plot.*

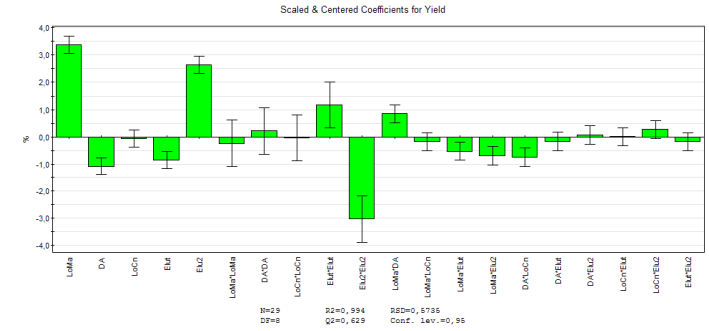
Step	Action



In the example above, the confidence limits (the black error bars shown on each green bar in the plot) do not cross zero. All of the terms are thus significant, with the ***LoCo*LopH*** two-way interaction term being least significant.

Positive bars have a positive influence on the response, in this example the ***Yield***, and negative bars a negative influence. From the above plot it is evident that increasing the ***LoMa*** (Load Mass) and ***LoCo*** (Load Conductivity) values, and decreasing the ***LopH*** (Load pH) value have a positive effect on the response.

5 Non-significant terms can be identified by the confidence limits for a coefficient (the black error bars) crossing zero. The diagram below shows the extreme example where no terms are significant.



Insignificant terms should be removed from the model one at a time before reanalysing the model.

6 If the model does not look good or non-significant terms are present, edit the model or continue with the extended analysis before editing the model.

See *Section 5.4.4 Analyze and evaluate the model - extended analysis, on page 154* and *Section 5.4.5 Edit the model, on page 161* for more information.

Step	Action
7	If the model looks good and all terms are significant, continue with *Section 5.4.6 Use the model, on page 164*.

5.4.4 Analyze and evaluate the model - extended analysis

Introduction

If you want to perform further analysis of the model in order to decide how to proceed, an extended report can be generated. The following plots and tables are displayed in the extended report in addition to the basic analysis:

- Residuals versus variables plot
- Residual versus run order plot
- Interaction plot
- Observed versus Predicted
- Main effects plot
- ANOVA table
- Correlation matrix

This section describes the plots in the extended report and gives information about how to evaluate the plots.

Open and view plots for extended analysis

To be able to view the plots for extended analysis create an extended report.

See *Create a report, on page 173* for information about how to create an extended report.

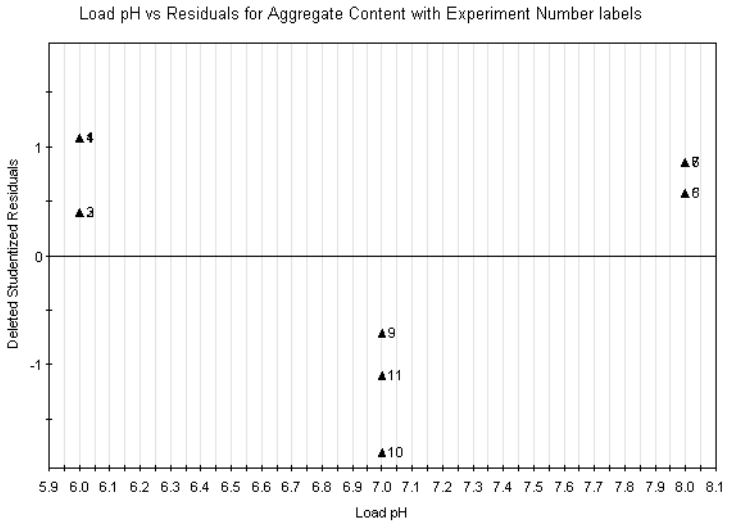
Residuals versus variables plot

The ***Residuals Plot vs. Variable*** shows the residuals (i.e., the minimized error between the measured and theoretical data according to the model) for one factor and one response.

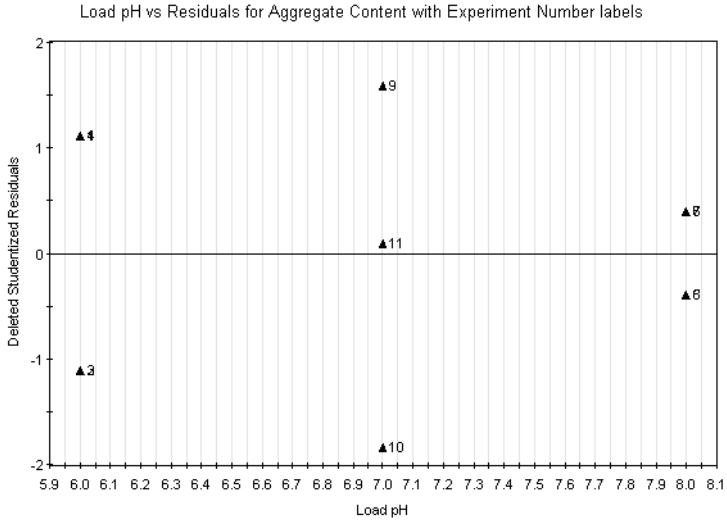
The residuals should be randomly distributed with no pattern. When a curved pattern can be seen in the plot this may indicate that a quadratic term is missing in the model. In this case try adding a quadratic term to the model and see if the model is improved.

See *Section 5.4.5 Edit the model, on page 161* for information about how to add a quadratic term to the model.

The illustration below shows an example of a plot indicating that a quadratic term is missing in the model.



The illustration below shows the plot for the same experiment when a quadratic term has been added to the model. Now the residuals are randomly distributed with no pattern.



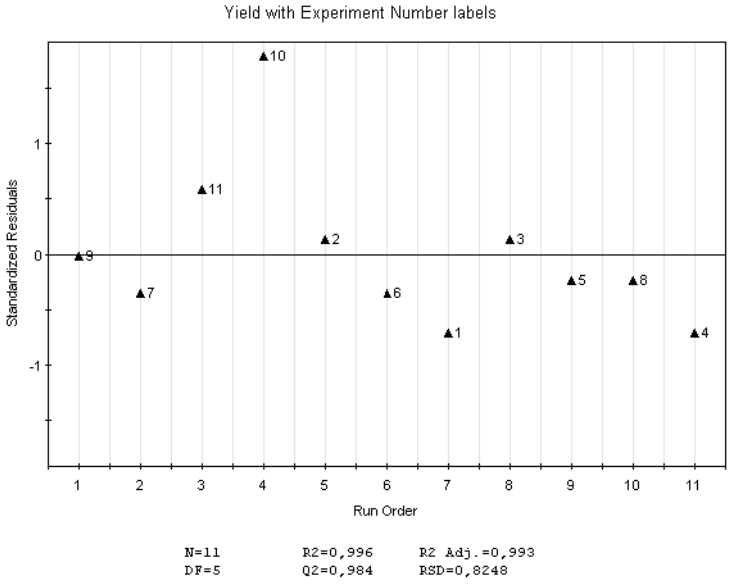
Note: *When you find curvature in the model, the design for the experiment in the* ***DoE*** *setup should be changed to design allowing quadratic terms to be added to the model (Full factorial 3 levels, CCC, CCF, Box Behnken, Rechtschaffner or Doehlert). If the experimental setup is sufficiently stable the star point experiments alone can be run, otherwise it is recommended to rerun all experiments.*

Residual versus run order plot

The ***Residuals Plot vs. Run order*** shows the residuals for the run order and one response.

The residuals should be randomly distributed with no pattern. A pattern in the plot indicates a change in residuals over time. This could, for example, be the result when randomization errors exist in the experiment.

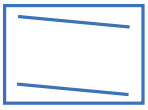
The illustration below shows a plot where the residuals are randomly distributed with no pattern.

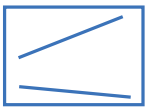
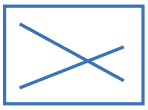


Interaction plot

The ***Interaction*** plot shows if there is any interaction (i.e., when the effect of one factor depends on another factor) between two factors. The illustration below shows an example of an interaction plot. In this example there is an interaction between load mass (***LoMa***) and load conductivity (***LoCo***).

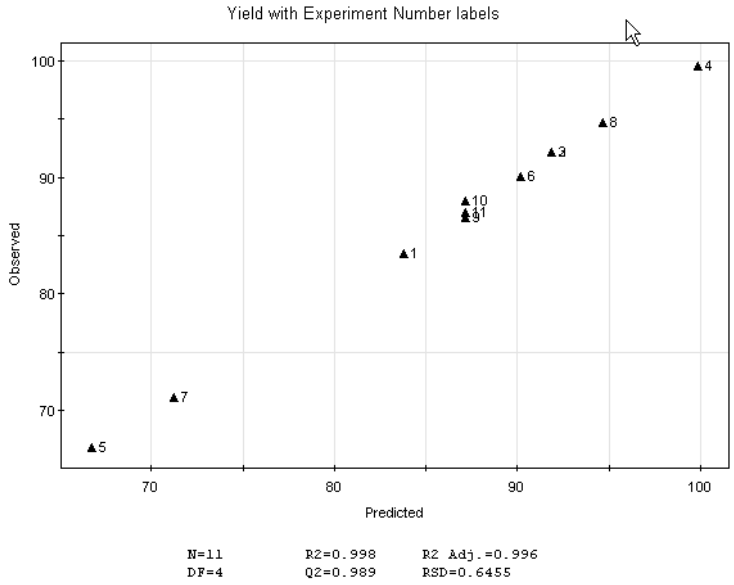
The table below describes how to interpret different interactions plot in a schematic way:

Plot	Description
	The two lines are parallel. This plot shows an example of no interaction between the two factors.

Plot	Description
	The two lines are not parallel. This plot shows an example of interaction between the two factors.
	The two lines are crossing. This plot shows an example of strong interaction between the two factors.

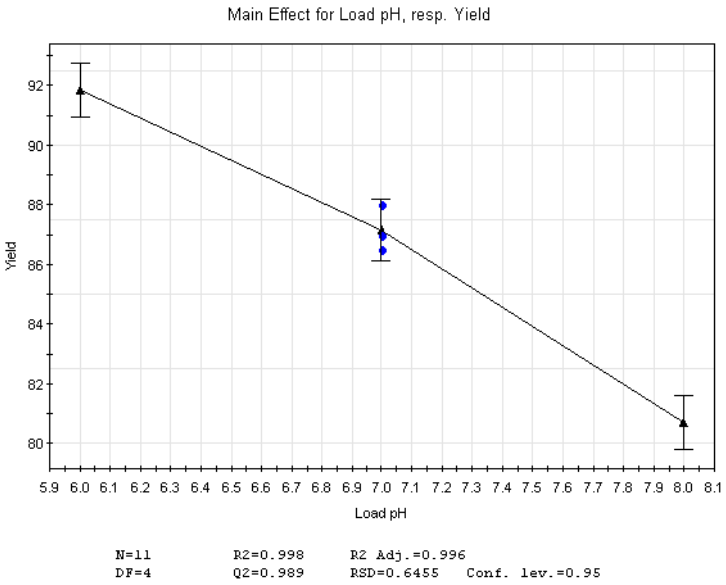
Observed versus Predicted for each Response plot

The ***Observed vs. Predicted for each Response (Y)*** plot can be used to judge the quality of the model. With a good model all the points will fall on the 45° line (illustrated in the plot below).



Main effects plot

The main effects plot displays the predicted response values when a factor varies from its low to its high level, all other factors in the design being set on their averages.



ANOVA table

The ANOVA (analysis of variance) table gives a numerical presentation of the variance analysis.

The illustration below shows an example of the ANOVA table.

Recovery MassOut/MassIn	DF	SS	MS (variance)	F	p	SD
Total	10	79468.2	7946.82			
Constant	1	78570.5	78570.5			
Total Corrected	9	897.672	99.7413			9.98706
Regression	6	893.347	148.891	103.271	**0.001**	12.2021
Residual	3	4.32527	1.44176			1.20073
Lack of Fit	1	0.0786042	0.0786042	0.0370193	**0.865**	0.280364
(Model Error)						
Pure Error	2	4.24666	2.12333			1.45717
(Replicate Error)						
	N = 10	Q2 = 0.988		Cond. no. = 3.635		
	DF = 3	R2 = 0.995		Y-miss = 0		
		R2 Adj. = 0.986		RSD = 1.201		

When looking at the ANOVA table, the p-values for regression, Lack of Fit (model error) and condition number give important information about the model. The table below describes these values in more detail.

Value	Description	Interpretation
Regression p-value	The regression p-value is a measure of the significance of the regression model.	p<0.05 indicates a significant regression model.
Lack of Fit p-value	The lack of fit p-value is a measure comparing the model error with the replicate error. This value is used in the calculation of Model Validity in the Summary of Fit plot.	p>0.05 indicates a good model. If p<0.05, this indicates that the model does not describe the relation between Y and X and that a quadratic term may be missing. See *Section 5.4.5 Edit the model, on page 161* for information about how to add a quadratic term to the model. A low p-value may also be due to other reasons, for example terms missing or that there is no correlation between X and Y that can be modeled.
Cond. no. (condition number)	The condition number can be used to investigate if the design is appropriate to use, especially if any of the default designs suggested in the ***Method Editor*** have been altered. Depending on the design, different condition numbers are expected for the model to be good.	• When the objective is screening and robustness testing - Good design when Cond. no.<3 - Questionable design when Cond .no.=3-6 - Poor design when Cond. no.>6 • When the objective is optimization - Good design when Cond. no.<8 - Questionable design when Cond .no.=8-12 - Poor design when Cond. no.>12

Correlation matrix

The correlation matrix gives a numerical presentation of the correlation between factors and responses and shows if the fit of the model is reasonable. The linear correlation coefficients R between all the terms in the model and all the responses are displayed in the correlation matrix.

Process factors are log-transformed, scaled, and centered and responses are log transformed. The value of the correlation coefficient R represents the extent of the linear association between two terms. The value of R ranges from -1 to 1. When R is near zero there is no linear relationship between the terms. Correlation coefficients above the threshold between a term in the model and the responses are colored green.

The illustration below shows an example of the correlation matrix.

	LoMa	LoCo	LopH	LoMa*LopH	LoCo*LopH	Yiel
LoMa	1	0	0	0	0	0,72118
LoCo	0	1	0	0	0	0,287094
LopH	0	0	1	0	0	-0,512175
LoMa*LopH	0	0	0	1	0	0,353699
LoCo*LopH	0	0	0	0	1	-0,0803863
Yiel	0,72118	0,287094	-0,512175	0,353699	-0,0803863	1

5.4.5 Edit the model

Introduction

Editing of the model may be necessary after analysis of the model, if the current model does not give a good fit. In the analysis you may for example:

- find insignificant terms that need to be removed
- find that the model may have curvature and that a quadratic term needs to be added

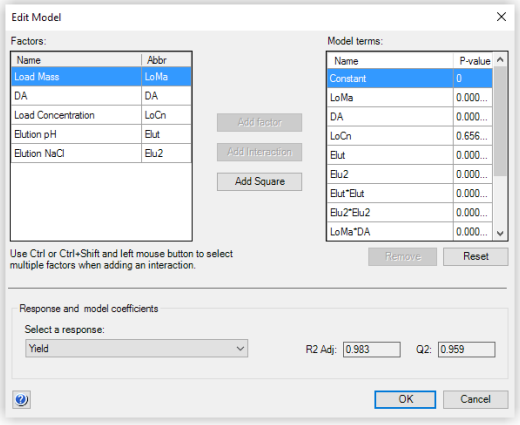
The refined model can be analysed to see if it better fits the data.

This section describes how to edit the model.

Edit the model

The table below describes how to edit a model:

Step	Action
1	In the ***Analysis*** tab, click ***Edit Model...***. *Result:* The ***Edit Model*** dialog box opens.
2	Note the ***R2 Adj*** and ***Q2*** values for the response(s) before starting to edit the model. Select different responses in the ***Select a response*** drop-down list. When editing the model, the ***R2 Adj*** and ***Q2*** values are updated. Higher values indicate a better model. See also *Analyze and interpret the model - basic analysis, on page 147* for a description of the values.



Step	Action
3	Non-significant terms may have been found in the analysis of the model (for example in the coefficient plot). To remove a non-significant term, select the term in the ***Model terms*** table and click ***Remove***. If the ***P-value***>0.05, the term is not significant. ***Note:*** *Always remove non-significant terms from the model one by one, starting with the least significant interaction or quadratic term. When the first term has been removed, the significance of the other terms changes. The* ***P-value*** *can be used to determine which term to be removed next.* ***Note:*** *If you fit a model to two or more responses, a model term that is not significant for one response may be significant for another response. Then the term should not be removed. Before removing a term, always check that the term is not significant for any of the other responses by selecting the response in the* ***Select a response*** *drop-down list and checking the* ***P-value*** *for the term you want to remove.* ***Note:*** *If a main term is not significant but one of its interaction terms is significant the main term should not be removed.* ***Note:*** *If a main term is removed its interaction terms are also removed.* *Result:* The term is removed from the model and the ***R2 Adj*** and ***Q2*** values are updated. If the model refinement gives a higher ***Q2*** value, the model refinement is justified. If one model is fitted to several responses, view the ***R2 Adj*** and ***Q2*** values for all responses.
4	Based on the previous analysis, add the appropriate terms to the model. **a.** Add an interaction term by selecting the appropriate factors in the ***Factors*** table and clicking ***Add Interaction***. ***Tip:*** *Use the* **Ctrl** *or* **Shift** *keyboard key to select multiple factors.* **b.** Add a quadratic term to the model by selecting the appropriate factor in the ***Factors*** table and clicking ***Add Square***. ***Note:*** *Quadratic terms can be added if any of the plots in the analyses indicates that a quadratic term is missing (in the* ***Residuals vs. Variables*** *plot, for example).*

Step	Action
	Note: *When you find curvature (i.e., a quadratic term needs to be added) in the model, the design for the experiment in the **DoE** setup should be changed to an extended Full Fractional (CCC or CCF) design. If the experimental setup is sufficiently stable the star point experiments alone can be added, otherwise it is recommended to rerun all experiments.* *Result:* The terms are added to the model. If the model refinement gives higher ***R2 Adj*** and ***Q2*** values, the model refinement is justified. If one model is fitted to several responses, view the ***R2 Adj*** and ***Q2*** values for all responses.
5	To return to the original model settings, click ***Reset***.
6	When you are satisfied with the editing, click ***OK***. *Result:* The ***Edit Model*** dialog box is closed and the ***Analysis*** tab displayed showing the new plots for the edited model.
7	Perform an analysis of the edited model to see if the new model is ***OK***. See *Analyze and interpret the model - basic analysis, on page 147* for information about how to analyze the model.

5.4.6 Use the model

Introduction

When you have found a good model, use the model to draw conclusions and to decide if more experiments are needed and what experiments to perform. The following plots and tools can be used in the evaluation:

- ***Response surface plot***

  Generate a response surface plot to get a graphical representation of the experimental region. From this, the most interesting area can be used to plan new experiments, verifying experiments and to better understand the impact of large interactions between factors.

- ***Sweet Spot Plot***

  Generate a sweet spot plot to get a graphical representation of the experimental region where two response criteria are satisfied.

- ***Prediction***

  Use the predictor to predict response values for entered factor settings.

- ***Optimization***

  Use the optimizer to enter response and factor settings criteria and obtain suitable factor setting combinations for the set response criteria.

Note: *Information about significant terms and how they influence the response values has already been found in the analysis of the model by looking at the coefficient plot, interaction plot, main effects plot and correlation matrix. See* *Analyze and interpret the model - basic analysis, on page 147* *and* *Section 5.4.4 Analyze and evaluate the model - extended analysis, on page 154* *for information about how to evaluate these plots.*

This section describes how to use the model.

Generate response surface plot and edit settings

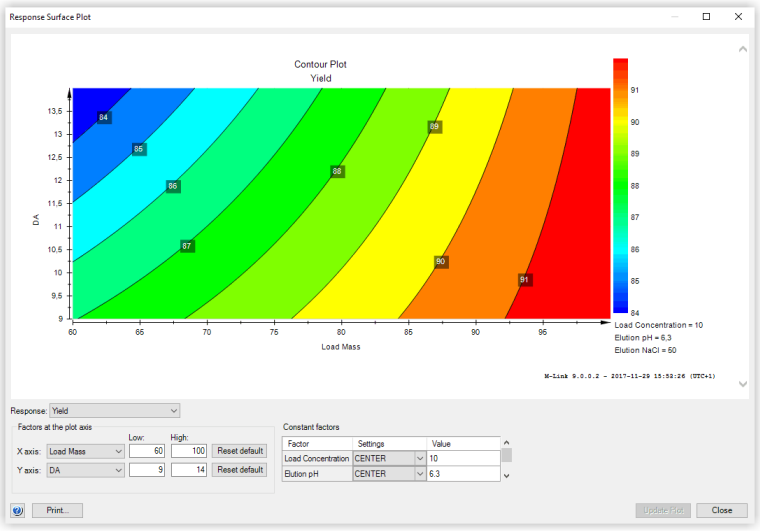
The response surface plot graphically displays the experimental region. It is helpful when you want to:

- get an overview of how different factor settings affect the response
- find the interesting experimental area
- get help in deciding where to start a new investigation
- get help in deciding where to make verifying experiments
- understand the impact of large interactions

Note: *The underlying model must be good and have a high Q^2-value. See* *Section 5.4.3 Analyze and evaluate the model - basic analysis, on page 143**.*

The following table describes how to generate a response surface plot and how to evaluate the plot:

Step	Action
1	In the ***Analysis*** tab, click ***Response Surface Plot***. *Result:* The ***Response Surface Plot*** dialog box opens.



The ***Contour Plot*** shows a "map" of the model. The plot has a color scale from blue to red. For each color, the response value is displayed.

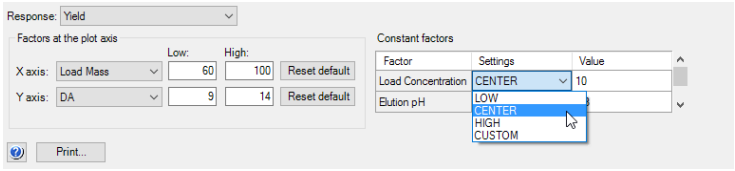
The factors selected are displayed on the ***X axis*** and ***Y axis*** in the ***Contour Plot*** (from ***Low*** to ***High*** as selected in the ***Factors at the plot axes*** area).

If you have more than two factors, the other factors will have constant values. The currently entered constant value(s) is displayed to the right of the contour plot. This means that this value is kept constant while the factors on the X- and Y-axes are varied.

The red area indicates the area where the response is maximized using the factor settings within this area and the current constant value(s).

2 It is possible to change the factors and their corresponding settings for the response surface plot as well as the constant values for the other factor(s). This is done per response if you have several responses.

In this way you can see what happens if constant values are changed and if other factors and/or factor settings are set on the contour plot axes. This will help you to decide if/which complementary experiments need to be performed.

Step	Action
	For example, you may want to investigate which factor settings to use in new ***DoE*** setup to narrow down the area of interest. The coefficient plot can be used to see which terms have the greatest positive or negative effect on the response. This information can be tested by changing the contour plot settings and updating the plot.
3	To change the ***Contour Plot*** settings: **a.** Select the response for the contour plot in the ***Response*** list. **b.** Select factors for the ***X axis*** and ***Y axis*** and their corresponding ranges. To return to the default values, click the ***Reset default*** button. **c.** Select or enter values for the ***Constant*** factors by choosing in the ***Settings*** list as shown below. If selecting ***CUSTOM***, click in the ***Value*** field and enter a value. 
4	Click ***Update Plot***. *Result:* The ***Contour Plot*** is updated.
5	When you have obtained the appropriate information to help you in the decision on how to proceed it is possible to print the ***Contour Plot***. **a.** Click ***Print***. *Result:* The ***Print Preview*** dialog box opens. **b.** Click ***Print***. *Result:* The standard ***Print*** dialog box opens. **c.** Select the appropriate printer and click ***Print***.

Generate sweet spot plot and edit settings

The sweet spot plot graphically displays the range where two or more selected response criteria are satisfied. It is helpful when you want to:

- get an overview of what factor ranges will give a desired response
- find the interesting experimental area
- get help in deciding where to start a new investigation

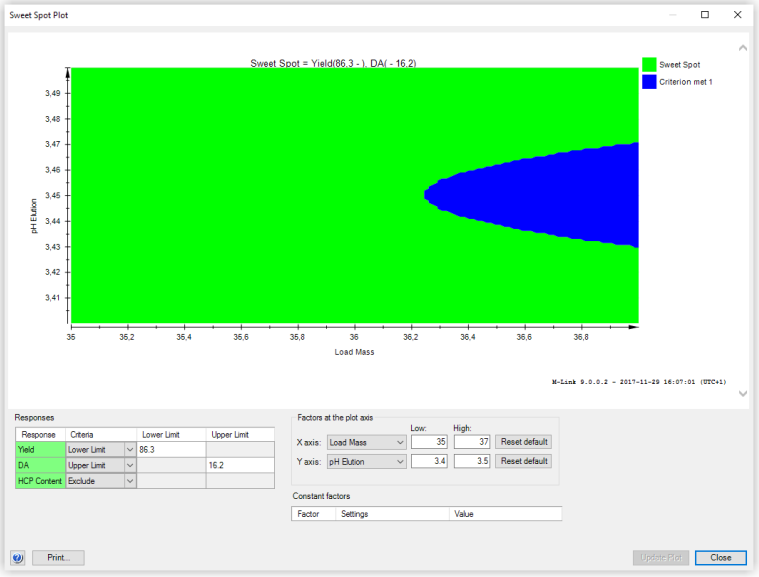
- get help in deciding where to make verifying experiments

Note: *The underlying model must be good and have a high Q^2-value. See Section 5.4.3 Analyze and evaluate the model - basic analysis, on page 143.*

Note: *The sweet spot plot will only provide useful information of two or more factors are set as responses.*

The following instruction describes how to generate a sweet spot plot and how to evaluate the plot:

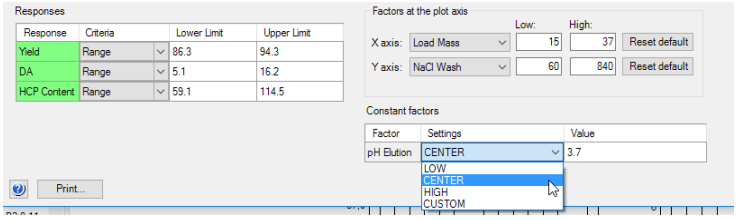
Step	Action
1	In the ***Analysis*** tab, click ***Sweet Spot Plot***. *Result:* The ***Sweet Spot Plot*** dialog box opens.



The plot shows a map of the model, visualizing factor ranges where all response criteria are satisfied. Areas where all criteria are not satisfied are colored different shades of blue. Areas of the plot where all response criteria are satisfied are colored green. The green areas are called "sweet spots". The current sweet spot criteria is listed above the plot.

The sweet spot is defined as a prediction interval. That is, the likelihood that a future observation will fall within the sweet spot criterion is larger further away from the blue border.

The factors selected are displayed on the ***X axis*** and ***Y axis*** in the ***Sweet Plot*** (from ***Low*** to ***High*** as selected in the ***Factors at the plot axis*** area).

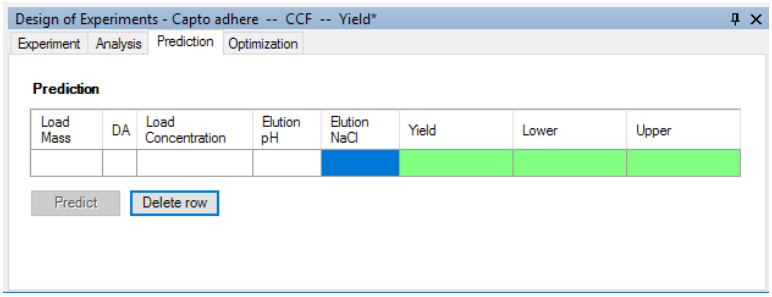
Step	Action
	If you have more than two factors, the other factors will have constant values. The currently entered constant value(s) is displayed to the right of the contour plot. This means that this value is kept constant while the factors on the X- and Y-axes are varied.
2	It is possible to change the response criteria for the selected responses. Response criteria can be ranges, upper and lower limits and excluded ranges. Criteria for the responses are set individually. You can also change the value(s) of the constant factor(s). In this way you can see what happens if you redefine the response criteria and if constant factors are changed. To change the ***Sweet Spot Plot*** settings: **a.** Select the response criteria for the sweet spot plot in the ***Criteria*** lists. There is one criterion for each of the responses. **b.** Select factors for the ***X axis*** and ***Y axis*** and their corresponding ranges. To return to the default values, click the ***Reset default*** button. **c.** Select or enter values for the ***Constant*** factors by choosing in the ***Settings*** list. If selecting ***CUSTOM***, click in the ***Value*** field and enter a value. 
3	Click ***Update Plot***. *Result:* The ***Sweet Spot Plot*** is updated.
4	When you have obtained the appropriate information to help you in the decision on how to proceed it is possible to print the ***Sweet Spot Plot***. **a.** Click ***Print***. *Result:* The ***Print Preview*** dialog box opens. **b.** Click ***Print***. *Result:* The standard ***Print*** dialog box opens. **c.** Select the appropriate printer and click ***Print***.

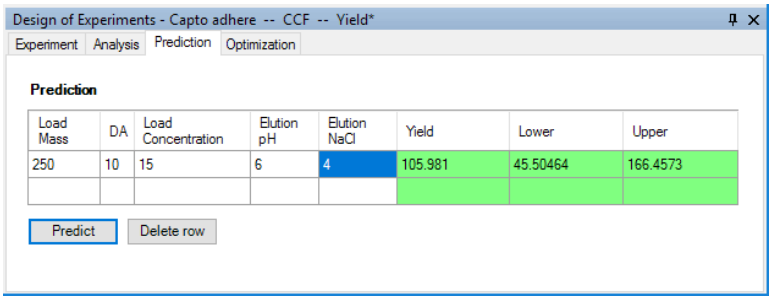
Predict response values

It is possible to predict response values based on entered factor settings using the model. This is useful when you want to find out how detailed factor settings influence the response(s) in an optimization experiment. Factor settings are entered and response values are calculated when using the ***Prediction*** list.

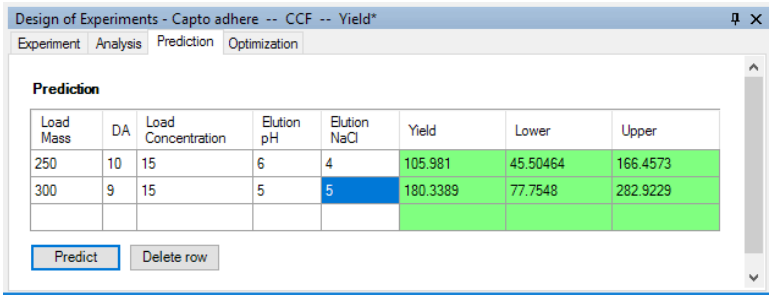
The table below describes how to use the ***Prediction*** list:

Step	Action
1	Select the ***Prediction*** tab in the ***Design of Experiments*** box. *Result:* The ***Prediction*** list opens.
2	Enter the appropriate settings for the different factors in their respective fields.
3	Click the ***Predict*** button. *Result:* The response value is calculated and displayed in the ***Yield*** field, together with the ***Lower*** and ***Upper*** confidence limits. The larger the confidence interval, the more uncertain the calculation is.





Step	Action
4	To enter other factor settings, enter the settings in the empty row below and click ***Predict***. In this way it is possible to compare different response values for different factor settings.



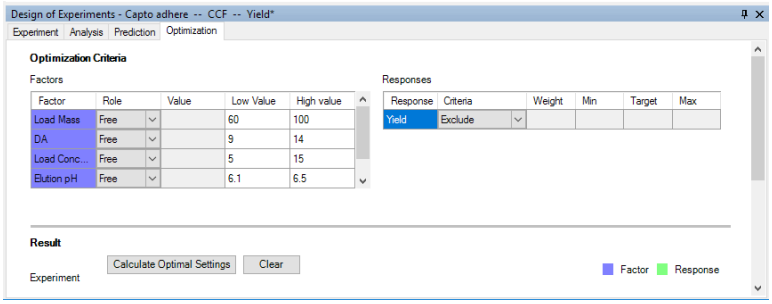
Repeat this procedure until you are satisfied.

Optimize response values and factor settings

It is possible to optimize the response values using the optimizer. When using the optimizer, criteria for the response values and factor settings are entered (e.g., ***Yield***>90%) and factor settings are calculated. In this way, the experimental region can be moved to an optimum.

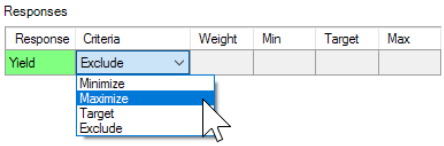
The table below describes how to use the optimizer:

Step	Action
1	Select the ***Optimization*** tab in the ***Design of Experiments*** box. *Result:* The ***Optimization Criteria*** and ***Result*** tables are displayed.



Step	Action

2 In the ***Responses*** area, select the ***Criteria*** for the response.



The following choices are available:

a. ***Minimize***

The response value should be minimized. Enter ***Target*** value and ***Max*** value for the response.

b. ***Maximize***

The response value should be maximized. Enter ***Target*** value and ***Min*** value for the response.

c. ***Target***

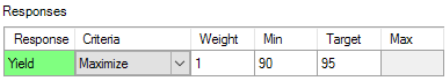
The response value should be optimized to reach the ***Target*** value. Enter ***Min***, ***Target*** and ***Max*** values for the response.

d. ***Exclude***

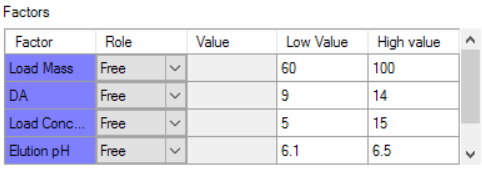
The response should not be included in the optimization (if you have several responses)

Result:

The entered values are displayed.

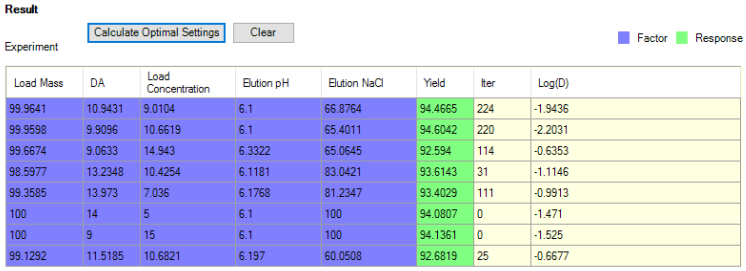


3 In the ***Factors*** area, select ***Role*** and settings for each factor:



a. If the role ***Free*** is selected, the factor settings to be calculated for the response can have values within the entered ***Low Value*** and ***High value*** range. Enter the ***Low Value*** and ***High value*** as appropriate (to get an idea of the new region of interest, use the response surface plot).

Step	Action
	b. If the role ***Constant*** is selected, the factor setting is constant. Enter the factor value in the ***Value*** field.
4	In the ***Result*** area, click the ***Calculate Optimal Settings*** button. *Result:* The results are displayed in the ***Experiment*** table.



Load Mass	DA	Load Concentration	Elution pH	Elution NaCl	Yield	Iter	Log(D)
99.9641	10.9431	9.0104	6.1	66.8764	94.4665	224	-1.9436
99.9598	9.9096	10.6619	6.1	65.4011	94.6042	220	-2.2031
99.6674	9.0633	14.943	6.3322	65.0645	92.594	114	-0.6353
98.5977	13.2348	10.4254	6.1181	83.0421	93.6143	31	-1.1146
99.3585	13.973	7.036	6.1768	81.2347	93.4029	111	-0.9913
100	14	5	6.1	100	94.0807	0	-1.471
100	9	15	6.1	100	94.1361	0	-1.525
99.1292	11.5185	10.6821	6.197	60.0508	92.6819	25	-0.6677

It is possible to see the combination of factor settings that will give a certain response. The number of iterations for optimization is indicated in the ***Iter*** column. Lower (or more negative) ***Log(D)*** values (the logarithm of the distance to the target) indicate better results.

5.4.7 Create and print reports

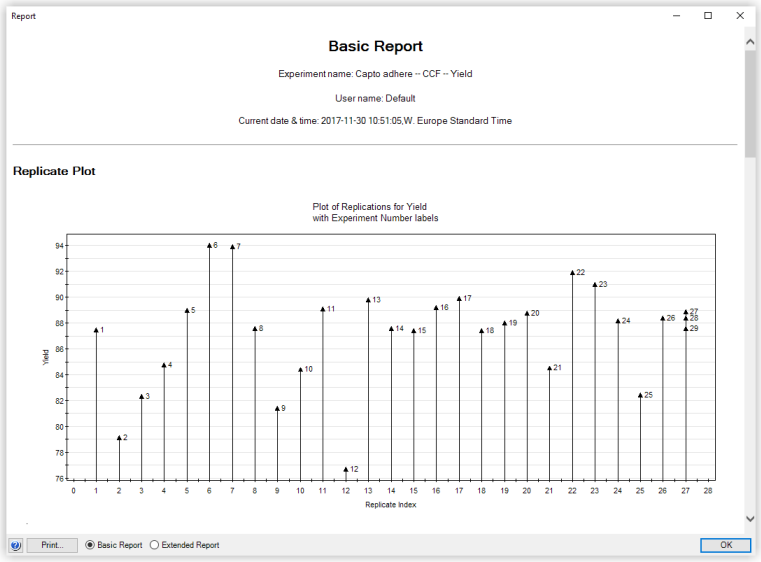
Introduction

This section describes how to create basic and extended reports and how to print the reports.

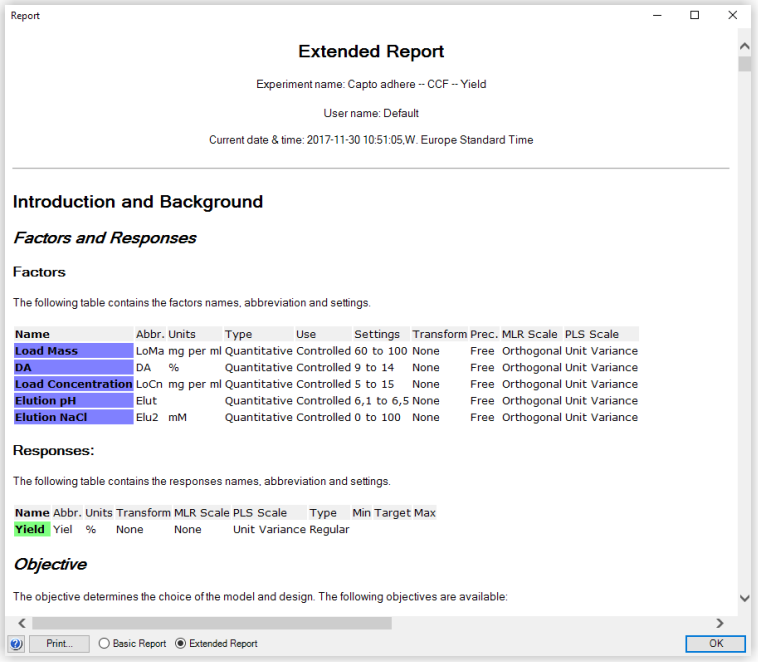
Create a report

The table below describes how to create a report:

Step	Action
1	In the ***Analysis*** tab, click ***Report...***. *Result:* The ***Report*** dialog box opens displaying the ***Basic*** report by default. It displays the Replicate, Summary of Fit, Normal probability and Coefficient plots.
2	To display the extended report select the ***Extended Report*** radio button.



Step	Action
	Result: The ***Extended Report*** opens in the ***Report*** dialog box. This report includes all available plots as well as the experiment setup, objective and design used in the experiment.



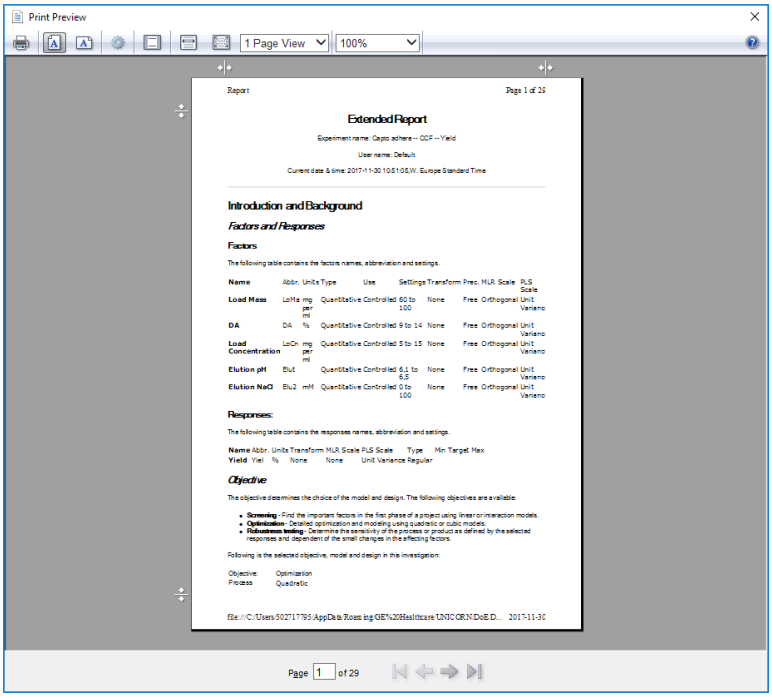
3	To view the information in the report use the vertical scrollbar.

Print a report

The table below describes how to print a report:

Step	Action
1	In the Report dialog box, click the ***Print...*** button.

Step	Action
	Result: The ***Print Preview*** dialog box opens.



2	Click the ***Print...*** button. *Result:* The standard ***Print*** dialog box opens.
3	Select the appropriate printer and click the ***Print*** button. *Result:* The report is printed.

6 BufferPro

About this chapter

This chapter describes how to create, edit and use buffer recipes created using the ***BufferPro*** tool in UNICORN.

BufferPro is only available for some systems.

In this chapter

6.1 BufferPro - Overview

Introduction

This section gives an introduction to the ***BufferPro*** tool in UNICORN, and includes a brief overview of the ***BufferPro*** recipes that are predefined.

What is BufferPro?

The full use of ***BufferPro*** is only available for some systems but all systems have the possibility to use the ***BufferPro*** recipes and calculate buffer composition as described in *Section 6.5 Calculate buffer composition using BufferPro, on page 189*.

The ***BufferPro*** tool allows automatic mixing of buffers during a run. Four stock solutions are generally used in a recipe, the buffering agent, a titrant, a salt stock solution and water. ***BufferPro*** facilitates ***Scouting*** or ***Design of Experiments*** runs using pH as a variable.

BufferPro is optimized for use with anion or cation exchange chromatography, but can also be used with ***SEC*** where the salt concentration may also be used as a variable during ***Scouting*** or ***Design of Experiements***.

Commonly used buffer systems have predefined recipes in UNICORN from which new recipes can easily be created. New or edited recipes may be stored as ***personal*** or ***global*** recipes.

UNICORN uses a robust algorithm to calculate pH ranges for optimal buffering taking into account the buffer type, concentration, temperature and ionic strength. Once an optimal buffer has been found, it is possible using ***BufferPro*** to calculate the buffer composition for the production of bulk-scale buffer solutions if required.

For details on...	See...
Scouting	*Chapter 4 Scouting, on page 86*
Design of Experiments	*Chapter 5 Design of Experiments, on page 98*

Workflow

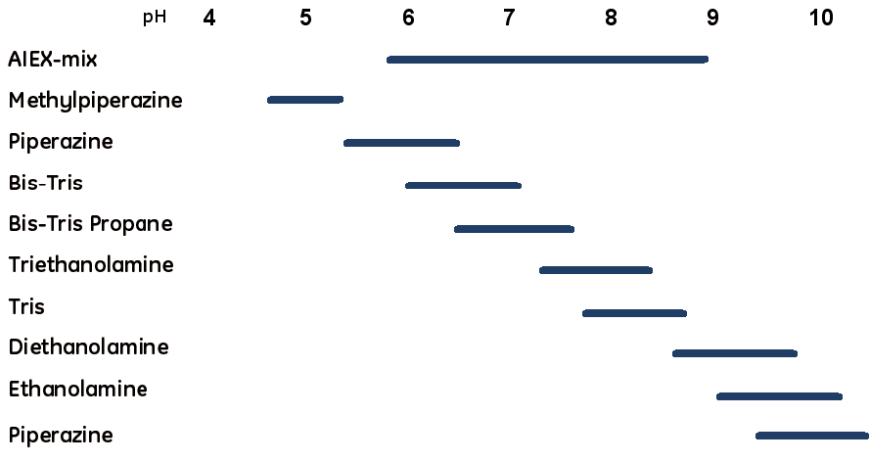
- If required, create a new ***BufferPro*** recipe.

  Tip: *Generally the predefined recipes will be sufficient.*
- Create a method including ***BufferPro***.
- Save the method.

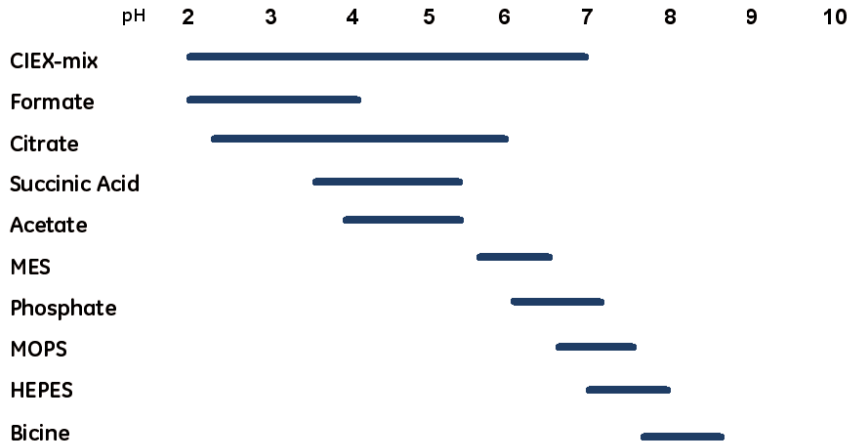
pH ranges for predefined buffers

The following diagrams show the optimal pH ranges for buffers commonly used in anion and cation exchange chromatography. Recipes for these buffers are predefined in UNICORN.

Anion exchange chromatography



Cation exchange chromatography

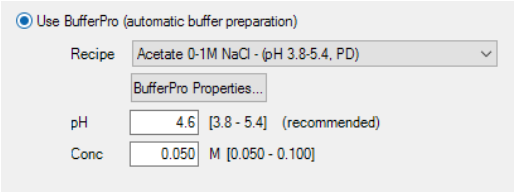


6.2 Create a method using BufferPro

Introduction

This section describes the how to use ***BufferPro*** recipes in a method. For details on how to edit methods see *Chapter 3 Create and edit methods, on page 23*.

Creating a BufferPro method

Step	Action
1	In the ***Method Settings*** phase of a method, select the ***Use BufferPro (automatic buffer preparation)*** option.  ***Note:*** *It is not necessary to have the **Enable pH monitoring** option checked. The output from the pH monitor is not used by the BufferPro algorithm.*
2	Select ***Recipe*** and enter ***pH*** and buffer concentration (***Conc***) within the specified range. ***Note:*** *For broad pH range multi-component buffers the concentration is fixed. For further information see Section 6.7 Predefined BufferPro recipes, on page 194.*
3	***Save*** the method.

6.3 Create and edit BufferPro recipes

About this section

This section describes how to create, edit, rename and delete ***BufferPro*** recipes. Predefined recipes may not be overwritten, renamed or deleted. Edited recipes, including edited predefined recipes, can be saved as ***global*** or ***personal*** recipes. ***Global*** recipes are available for all users, ***personal*** recipes only for the current user.

Note: *The predefined recipes can be used in the majority of cases. There is often no need to create a new recipe before creating a* ***BufferPro*** *method.*

Note: *For systems that do not have the* ***BufferPro*** *tool, it is still possible to edit* ***BufferPro*** *recipes in imported methods.*

In this section

6.3.1 Create and edit a BufferPro recipe

General considerations

The concentration of the buffer stock will affect the pH range and the settable concentration range in the method. The pH range will in general increase with increasing buffer concentration and decrease when lowered.

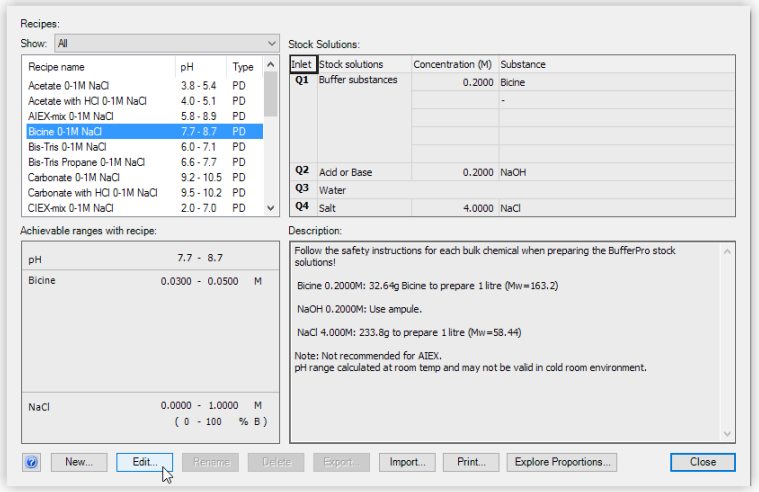
The titrant and buffer agent concentrations should be the same, since there may otherwise not be sufficient titrant to reliably obtain the entire pH range. For recipes titrated with strong acid/base, the concentration range that can be achieved is 15-25% of the buffer stock concentration. For conjugate acid/base titrants the corresponding range is 25-50% of the buffer stock concentration.

The following table describes how to create a new recipe and how to edit existing recipes.

Create/edit a recipe

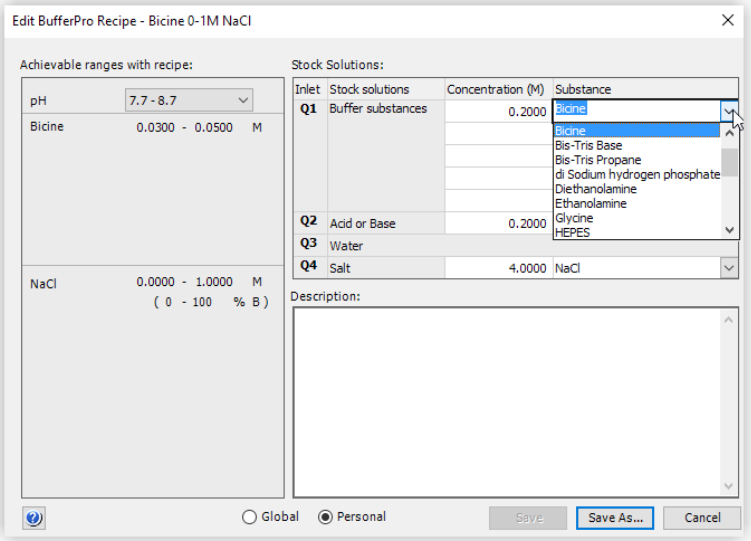
The following table describes how to create or edit a BufferPro recipe:

Step	Action
1	In the ***Method Editor***, select ***Tools*** →***BufferPro Recipes...***.
2	To create a new recipe, click ***New...*** in the ***BufferPro Recipe*** dialog. To edit an existing recipe, select the recipe to be edited from the list and click ***Edit...***. ***Note:*** *The available recipes may be filtered by type (**All**, **Predefined**, **Global** or **Personal**) by using the **Show** drop-down list.*



Step	Action

3 Select a ***Buffer substance*** from the drop-down list.



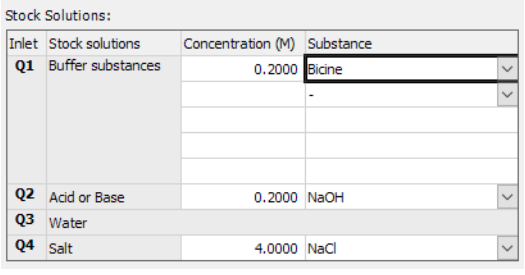
Note:

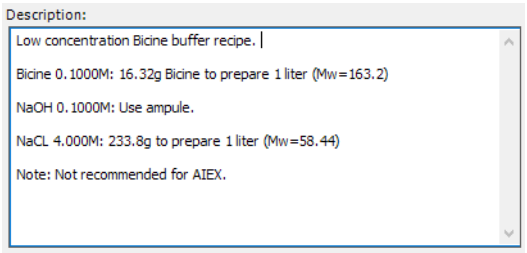
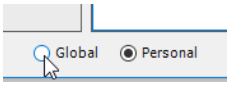
*For ÄKTA avant, up to five buffer substances may be included in the recipe for the **Q1** inlet. If more than one substance is used, the concentration of the final buffer in **BufferPro** will be fixed, and is then dependent on the concentration of the stock solutions.*

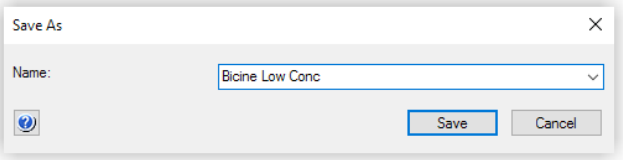
Note:

*To choose a conjugate acid-base pair as the buffer, select the base form as **Buffer substance** apart from phosphate where the acidic or basic form may be chosen. The conjugate acid or base will appear as an option in the **Acid or Base** drop-down list.*

4 Select the concentration and edit the value.



Step	Action
5	Choose a titrant (***Acid or Base***) from the drop-down list and if required edit its ***Concentration***. ***Note:*** *The titrant and the stock solution should generally have the same concentration. This is set as default for* ***Acid or Base*** *concentration.*
6	Choose a ***Salt*** from the drop-down list and edit its ***Concentration*** if required. ***Note:*** *The salt concentration of the stock solution should be four times larger than the desired maximum salt concentration for the gradient.*
7	Enter a description of the buffer.  ***Note:*** *Although the description is optional, it is highly recommended to add the recipe details for future reference.*
8	Select to save the edited recipe as ***Global*** or ***Personal*** and click ***Save as...*** or ***Save***.  ***Note:*** *Recipes can be changed from* ***Personal*** *to* ***Global*** *and vice versa by editing the recipe, changing the type then clicking on* ***Save****.* *Result:* The ***Save As*** dialog opens.

Step	Action
9	Enter a name and click ***Save***. 

6.3.2 Rename a BufferPro recipe

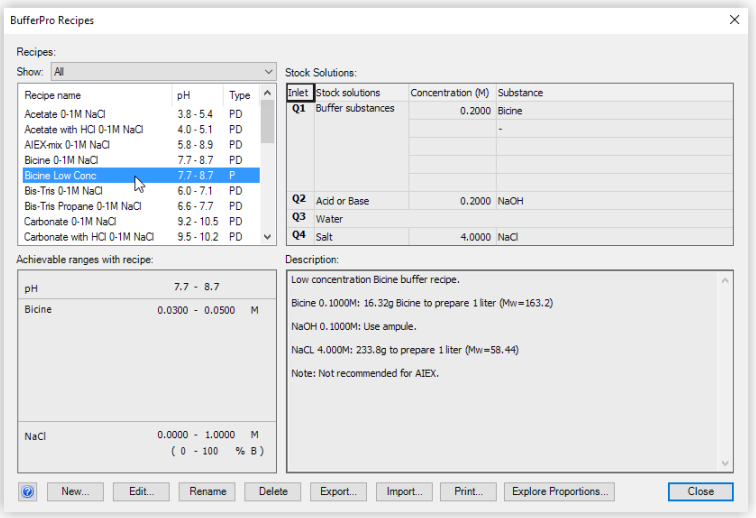
Introduction

The following table describes the steps for renaming a ***BufferPro*** recipe.

Note: *Predefined recipes (shown as **PD** in the **Type** column) cannot be renamed.*

Rename a recipe

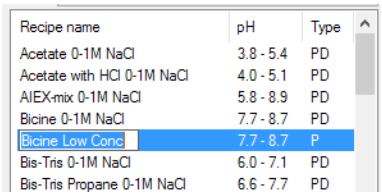
Step	Action
1	In the ***Method Editor***, select ***Tools*** →***BufferPro Recipes...***.
2	In the ***BufferPro Recipes*** dialog, select the recipe to be renamed.



Note:

*The available recipes may be filtered by type (**All**, **Predefined**, **Global** or **Personal**) by using the **Show** drop-down list.*

3	Click ***Rename*** and enter the new name.



6.3.3 Delete a BufferPro recipe

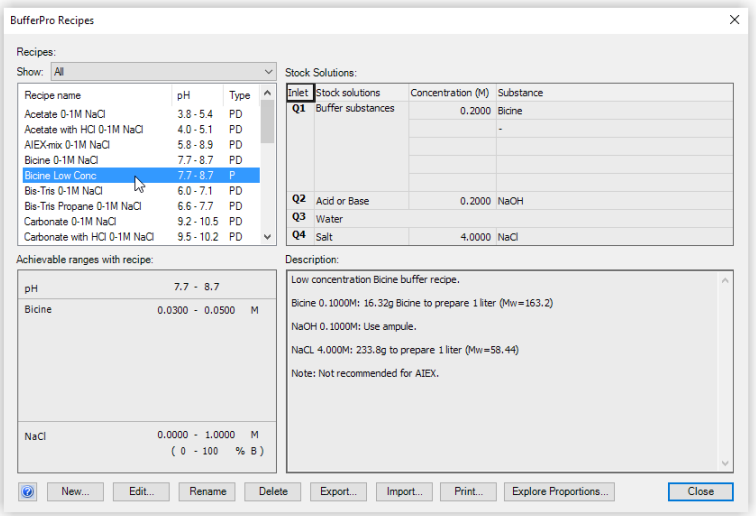
Introduction

The following table describes the steps needed to delete a ***BufferPro*** recipe.

Note: *Predefined recipes (shown as **PD** in the **Type** column) cannot be deleted.*

Delete a recipe

Step	Action
1	In the ***Method Editor***, select ***Tools*** →***BufferPro Recipes...***.
2	In the ***BufferPro Recipes*** dialog, select the recipe to be deleted.



Note:

*The available recipes may be filtered by type (**All**, **Predefined**, **Global** or **Personal**) by using the **Show** drop-down list.*

3	Click ***Delete***. A dialog will appear asking you to confirm the deletion.

6.4 Print a BufferPro recipe

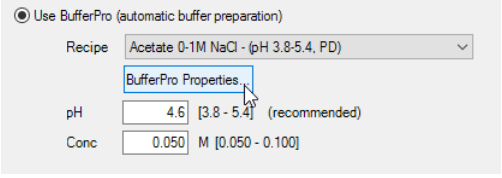
Introduction

This section describes how to print a BufferPro recipe from UNICORN. A recipe can be printed from the ***Phase Properties*** tab in ***Method Editor***, or from the ***BufferPro Recipes*** dialog box.

It is also possible to include the ***BufferPro*** recipes when printing the whole method. See *Section 3.6.3 Print a method, on page 63*.

Printing from Method Editor

The following table describes how to print a recipe from the ***Phase Properties*** tab in ***Method Editor***.

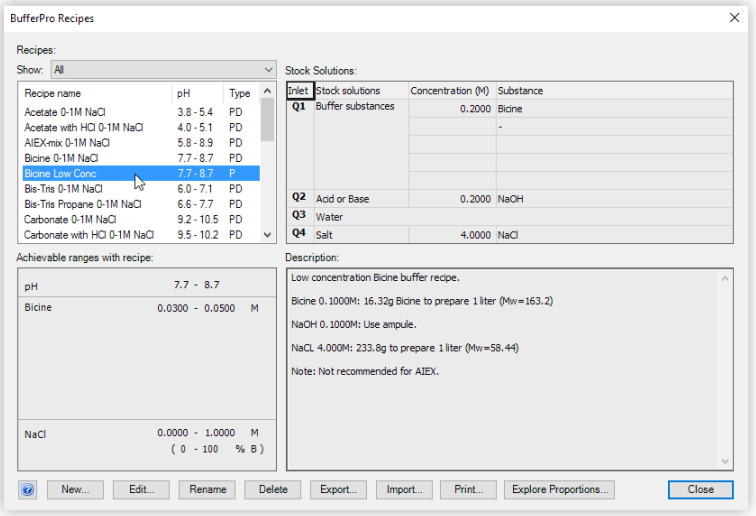
Step	Action
1	In the ***Phase Properties*** tab in the ***Method Editor***, click the ***BufferPro Properties...*** button. 
2	In the ***BufferPro Properties*** dialog box click the ***Print...*** button. *Result:* The ***Print*** dialog box opens.
3	Choose a printer from the drop-down list in the ***Print*** dialog box and click ***OK***.

Printing from *BufferPro Recipes* dialog box

The following table describes how to print a recipe from the ***BufferPro Recipes*** dialog.

Step	Action
1	In the ***Method Editor***, select ***Tools*** →***BufferPro Recipes***.

Step	Action
2	Choose the recipe to be printed from the list in the ***BufferPro Recipes*** dialog box.



3	Click the ***Print...*** button. *Result:* The ***Print*** dialog box opens.
4	Choose a printer from the drop-down list in the ***Print*** dialog box and click ***OK***.

6.5 Calculate buffer composition using BufferPro

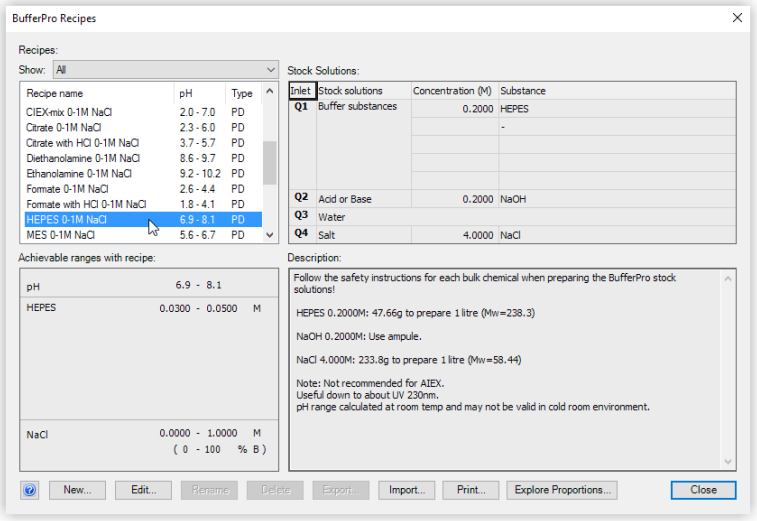
Introduction

This section describes how to calculate an exact buffer composition for a buffer previously optimized using ***BufferPro***. This is desirable when scaling up a purification procedure in order to prepare bulk-scale buffer solutions, for example ion exchange A and B buffers.

Calculating buffer composition

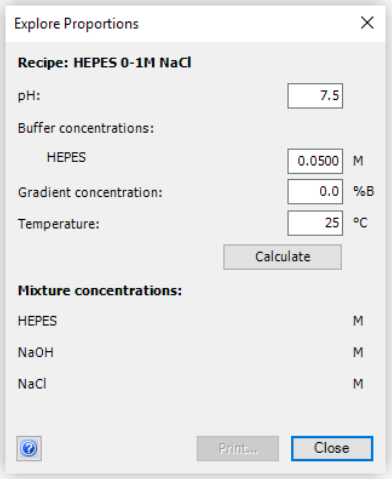
The following table describes the steps needed to calculate the buffer composition of a ***BufferPro*** recipe at a particular pH, buffer and gradient concentration, and temperature. In the examples shown in the table, a pH optimization scouting run has been performed. The buffer at which optimal separation was obtained was 50 mM HEPES, pH 7.8 at 25 °C, and the required peak eluted at 25% of the gradient.

Step	Action
1	In the ***Method Editor***, select ***Tools*** →***BufferPro Recipes...***.
2	Choose the appropriate recipe from the list in the ***BufferPro Recipes*** dialog box.



3 Click the ***Explore Proportions...*** button.

Step	Action
4	In the ***Explore Proportions*** dialog, enter the ***pH***, ***Buffer concentrations***, the desired ***Gradient concentration*** and ***Temperature***.



Note:

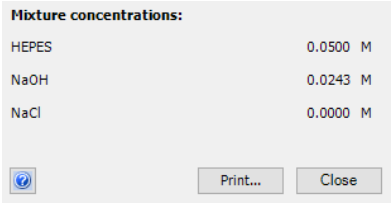
*The **buffer concentrations** may not exceed the limits of the recipe. If this is the case the **Calculate** button will be grayed out.*

5 Click ***Calculate***.

Note:

*If the **pH** given is beyond the optimal buffering range of the buffer recipe, a warning will be displayed.*

6 The actual concentrations of the components in the required buffer will be displayed.



Note:

It is important that the molar amounts are as exact as possible when mixing the buffers. It has been found that four decimal places in molar concentration gives reproducible results.

7 The buffer composition can be printed by pressing the ***Print...*** button.

6.6 Export and import BufferPro recipes

Introduction

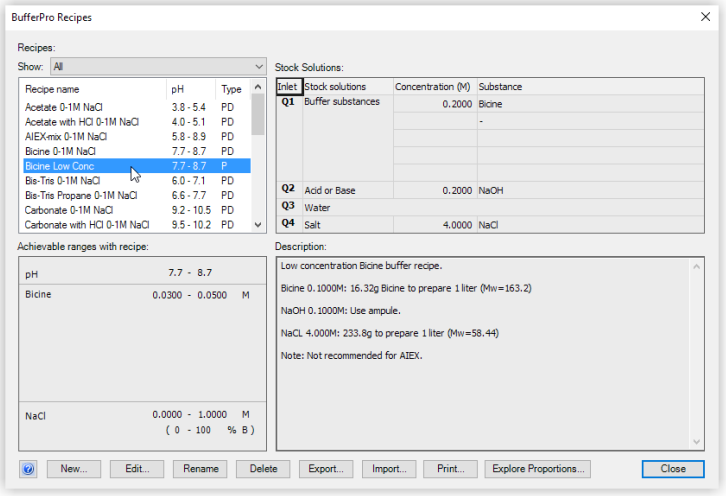
BufferPro recipes are stored internally in the UNICORN database. It is possible to export these recipes to a zip file on the local computer so that the recipe can be imported again later into the same database installation, or imported into another. This section describes how to export and import ***BufferPro*** recipes.

Exporting BufferPro recipes

The following table illustrates the steps required to export one or several recipes.

Note: *Predefined recipes cannot be exported, since these recipes will always be found in a UNICORN installation.*

Stage	Description
1	In the ***Method Editor***, select ***Tools*** →***BufferPro Recipes...***.
2	Choose the recipe to be exported from the list in the ***BufferPro Recipes*** dialog box.



Note:

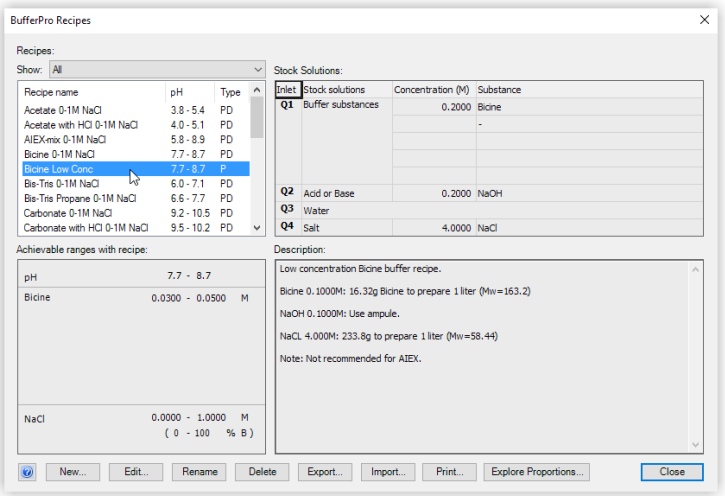
Several recipes may be exported to the same zip file. To select a continuous range, click on the first recipe then Shift-click the last. To add single recipes to a selection, Ctrl-click them.

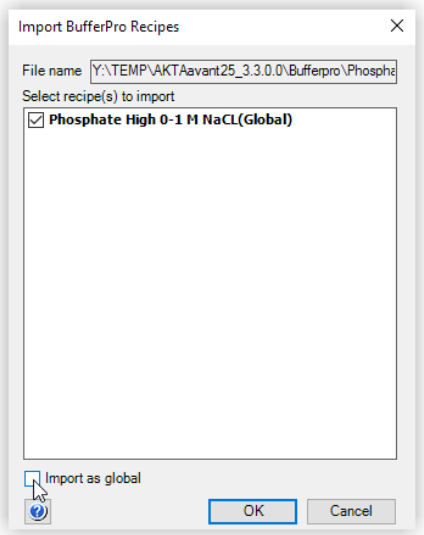
3	Click ***Export...***. Result: The ***Export*** dialog box opens.
4	Choose a location on the computer disk and a filename for the zip file.

Stage	Description
5	***Save*** the file.

Importing BufferPro recipes

The following table illustrates the steps required to import one or several recipes.

Stage	Description
1	In the ***Method Editor***, select ***Tools*** →***BufferPro Recipes...***. Result: The ***BufferPro Recipes*** dialog box opens. 
2	Click the ***Import*** button. *Result*The ***Import*** dialog box opens.

Stage	Description
3	Browse to the zip file containing the ***BufferPro*** recipe(s) on the computer disk and ***Open*** the file. *Result*The ***Import BufferPro Recipes*** dialog box opens. 
4	In the ***Import BufferPro Recipes*** dialog box, uncheck any recipe(s) that you do not wish to import. Select whether the recipe(s) should be imported as ***global***, otherwise they will be imported as ***personal*** recipes.
5	Click ***OK*** to import the recipe(s).

6.7 Predefined BufferPro recipes

Introduction

This section describes in detail the predefined buffer systems in ***BufferPro*** that are delivered with UNICORN.

General points

The following points should be taken into consideration:

- The pK_a of certain buffer substances can vary significantly with temperature. This means that the working pH range for optimal buffering will vary with temperature. It is possible to estimate the appropriate pH ranges using the ***Explore Proportions*** tool, see *Section 6.5 Calculate buffer composition using BufferPro, on page 189*.
- The two broad-range buffer systems, ***AIEX-mix 0-1M NaCl*** and ***CIEX-mix 0-1M NaCl*** may only be used at fixed concentration, since these are multi-component buffers.
- The working concentration for buffers that are mixed using conjugate acid-base pairs is 25-50% of the stock solution concentration. For buffers mixed using strong acid or base solutions the working concentration is 15-25% of the stock solution concentration. Although it may be possible to mix solutions outside this range, UNICORN will show a warning since the pH of the resulting buffer may not be reliable. If in doubt, check the pH of the eluent after running an experiment using a reliable lab pH meter.
- The pH range given is based on the narrowest range for effective buffering for the entire gradient (0-1M NaCl). The ionic strength of the mixed solution affects the apparent pK_a of the buffering agent. For pH outside the recommended range the buffering capacity is unreliable and should be avoided. UNICORN will display a warning in case either the required concentrations or pH will not provide adequate buffering. If in doubt, check the pH of the eluent after running an experiment using a reliable lab pH meter.
- Certain buffer substances are not recommended for anion exchange and others not for cation exchange. For example, phosphate buffers are not suitable for anion exchange. Buffer suitability is noted in the predefined recipes in UNICORN.

pH and concentration ranges for predefined recipes

The following table gives the optimal pH and concentration ranges for buffer recipes that are predefined in ***BufferPro*** at 25 °C.

Buffer system	pH range at 25 °C	Concentration range (M)	Comment
AIEX-mix 0-1M NaCl	pH 5.8-8.9	Fixed at 25% of the concentration of the stock solution.	Broad range buffer system for Anion exchange chromatography.
CIEX-mix 0-1M NaCl	pH 2.0-7.0	Fixed at 25% of the concentration of the stock solution.	Broad range buffer system for Cation exchange chromatography.
Acetate 0-1M NaCl	pH 3.8-5.4	0.05 - 0.1	Titrated with conjugate acid
Acetate with HCl 0-1M NaCl	pH 4.0-5.1	0.03 - 0.05	Titrated with strong acid
Bicine 0-1M NaCl	pH 7.7-8.7	0.03 - 0.05	Titrated with strong base
Bis-Tris 0-1M NaCl	pH 6.0-7.1	0.03 - 0.05	Titrated with strong acid
Bis-Tris Propane 0-1M NaCl	pH 6.6-7.7	0.03 - 0.05	Titrated with strong acid
Carbonate 0-1M NaCl	pH 9.2-10.5	0.05 - 0.1	Titrated with conjugate acid
Carbonate with HCl 0-1M NaCl	pH 9.5-10.2	0.03 - 0.05	Titrated with strong acid
Citrate 0-1M NaCl	pH 2.3-6.0	0.05 - 0.1	Titrated with conjugate acid
Citrate with HCl 0-1M NaCl	pH 3.7-5.7	0.03 - 0.05	Titrated with strong acid
Diethanolamine 0-1M NaCl	pH 8.6-9.7	0.03 - 0.05	Titrated with strong acid
Ethanolamine 0-1M NaCl	pH 9.2-10.2	0.03 - 0.05	Titrated with strong acid
Formate 0-1M NaCl	pH 2.6-4.4	0.05 - 0.1	Titrated with conjugate acid
Formate with HCl 0-1M NaCl	pH 1.8-4.1	0.03 - 0.05	Titrated with strong acid

Buffer system	pH range at 25 °C	Concentration range (M)	Comment
HEPES 0-1M NaCl	pH 6.9-8.1	0.03 - 0.05	Titrated with strong base
MES 0-1M NaCl	pH 5.6-7.0	0.03 - 0.05	Titrated with strong base
Methylpiperazine 0-1M NaCl	pH 4.6-5.3	0.03 - 0.05	Titrated with strong acid
MOPS 0-1M NaCl	pH 6.5-7.6	0.03 - 0.05	Titrated with strong base
Phosphate 0-1M NaCl	pH 5.9-7.2	0.05 - 0.1	Titrated with conjugate acid
Phosphate with HCl 0-1M NaCl	pH 6.2-6.9	0.03 - 0.05	Titrated with strong acid
Piperazine 0-1M NaCl, low pH	pH 5.5-6.4	0.03 - 0.05	Titrated with strong acid
Piperazine 0-1M NaCl, high pH	pH 9.3-10.5	0.03 - 0.05	Titrated with strong base
Succinic Acid 0-1M NaCl	pH 3.4-5.6	0.03 - 0.05	Titrated with strong base
Triethanolamine 0-1M NaCl	pH 7.4-8.4	0.03 - 0.05	Titrated with strong acid
Tris 0-1M NaCl	pH 7.6-8.7	0.03 - 0.05	Titrated with strong acid

7 Method queues

Introduction

This chapter describes how to create and edit method queues in UNICORN. It also describes how to import and export method queues into and from UNICORN, respectively. For information on how to create and edit individual methods, see *Chapter 3 Create and edit methods, on page 23*.

In this chapter

7.1 Method queues - overview

Introduction

A method queue in UNICORN is a linked set of methods to be run. The method queue can contain methods to be run on up to three different systems. Each system may have up to ten methods queued.

For example, a method queue might be useful on a single system when a wash procedure is programmed in a separate method. This method can then be linked to a series of different process methods ensuring the same wash procedure is used in each process. On multiple systems, the product of a separation on the first system might be the starting material for a separation on the next, allowing fully automatic multi-step processing.

Note: *When a method queue is started, an option is available to run the start protocol for the method queue only once. Notification limit warnings related to the number of times a column has been used, for example since the last CIP was performed, are only issued when the start protocol is performed. See* *Set notification limits for a column, on page 233**. In a method queue this may therefore not always be shown exactly when the notification limit is reached. Each run will however be noted in the column history, which should be checked before critical runs in a method queue. See* *View column history, on page 238**.*

Main steps when creating a method queue

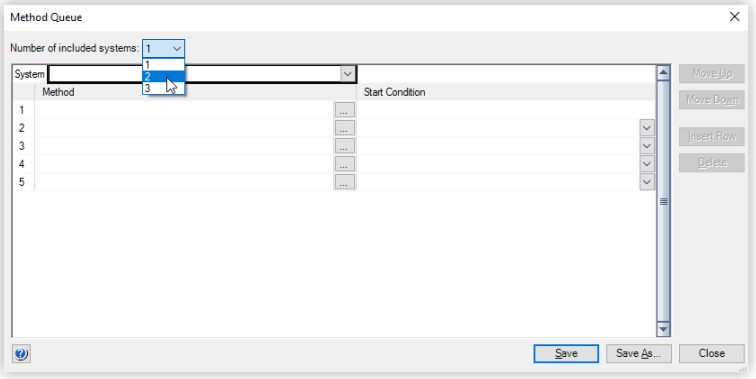
The main steps when creating a method queue are:

Step	Action
1	Create methods for the required system(s). See *Chapter 3 Create and edit methods, on page 23*.
2	Create/open a method queue • Create a new method queue or • Open an existing method queue that can be edited and saved with a new name
3	Save the method queue

7.2 Create a method queue

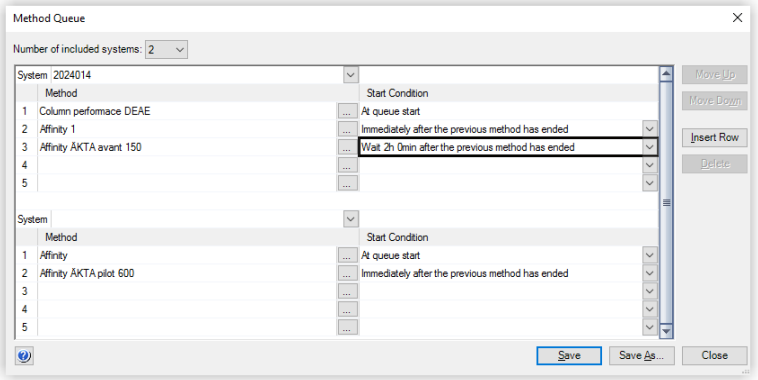
Creating a method queue

The following table describes how to create a method queue.

Step	Action
1	In the ***Method Editor***: • click the ***New Method Queue*** icon in the ***Toolbar***  or • Select ***File*** →***New Method Queue...*** *Result:* The ***Method Queue*** dialog box opens.
2	In the ***Method Queue*** dialog box, choose the ***Number of included systems*** from the drop down list.  *Result:* A separate method queue block will be added to the dialog box for each additional system if required.
3	Choose a system for each method queue block from the ***System*** drop down list.
4	Choose a ***Method*** to add to a method queue by pressing the browse button. *Result:* The ***Select Method*** dialog box opens.

Step	Action
5	In the ***Select Method*** dialog box, browse to the required method and click ***OK***. *Result:* The method is added to the method queue. ***Note:*** *For reasons of system compatibility, the individual methods should be saved for the system on which they are queued.*
6	Select a ***Start Condition*** for the method from the drop-down list. **a.** ***At queue start*** The method will begin at the start of the method queue. Only available for the first method for each system. **b.** ***Immediately after the previous method has ended*** The method will start when the previous has ended on the queue for that system. **c.** ***Wait...*** The method will start after a specified ***Wait*** time has elapsed since the previous method in the queue for the system has ended. A separate dialog box will open where the ***Wait*** time can be specified in ***Hours*** and ***Minutes***. The delay time will be shown in the ***Method Queue*** dialog box once entered. **d.** ***At ready command...*** The method will start when a ***Ready*** instruction in a method on another system has been executed. Using this start condition it is possible to connect methods running on different systems. A separate dialog box will open where the ***System*** and ***Method*** can be chosen. An arrow to the left of the method queues will show the connected methods, as shown in the diagram. ***Note:*** *The first* ***Method*** *for the first* ***System*** *will always have its* ***Start Condition*** *set to* ***At queue start****.* Available ***Start Conditions*** are:

Step	Action



7 Repeat steps 4 to 6 to add further methods to the ***Method*** list for each required system.

8 Click ***Save*** or ***Save As*** to save the completed method queue.

Note:

An error dialog box will be displayed if any of the methods are incompatible with the system on which they are queued.

7.3 Edit a method queue

Introduction

This section describes how to open, delete and edit existing method queues. Methods can be inserted and deleted from a method queue, and their order in the queue can be changed.

Opening a method queue

The table below describes how to open an existing method queue in the database:

Step	Action
1	In the ***Method Editor***: • Click the ***Open Method Navigator*** icon in the ***Toolbar***  or • Click ***File***, and then select ***Open...*** or • Click ***View***, and then ***Method Navigator***. *Result:* The ***Method Navigator*** is displayed.
2	Select the method queue to be opened in the ***Folder name*** column.
3	To open the method queue, • Click the ***Open*** button located in the toolbar of the ***Method Navigator*** pane  or • double-click the selected method queue *Result:* The ***Method Queue*** dialog box is opened with the details for the opened method queue. ***Note:*** *If a method contained in the method queue has been altered since the last time it was saved, an information dialog box will be displayed.*

Delete a method queue

The table below describes how to delete a method queue from the database:

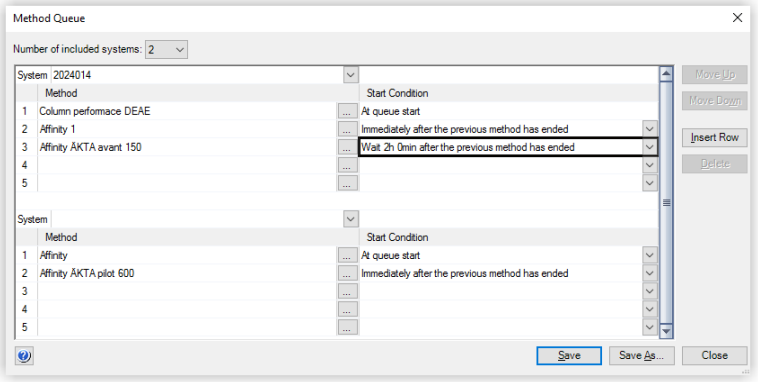
Step	Action
1	In the ***Method Editor***: • Click the ***Open Method Navigator*** icon in the ***Toolbar***  or • Click ***File***, and then select ***Open...*** or • Click ***View***, and then ***Method Navigator***. *Result:* The ***Method Navigator*** is displayed.
2	Select the method queue to be deleted in the ***Folder name*** column.
3	To delete the method queue, • select ***Edit*** →***Delete*** or • press the **Delete** key or • right-click the selected method queue and select ***Delete*** from the context menu. *Result:* A dialog box will appear asking to confirm the delete operation.

Insert a method into a method queue

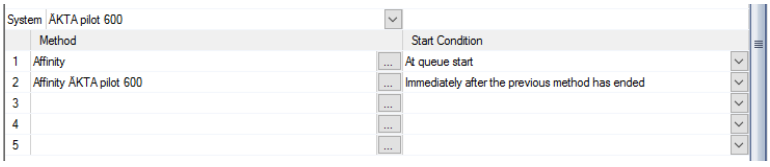
The following table describes how to insert a method into the ***Method*** list for a system.

Step	Action
1	Open the method queue, see *Opening a method queue, on page 202*. *Result:* The ***Method Queue*** dialog box opens with the details for the chosen method queue.

Step	Action
2	In the ***Method Queue*** dialog, select the position in the list at which a method will be inserted by clicking on the ***Method*** column.



3	Insert a new row by clicking on the ***Insert Row*** button. *Result:* An empty row is inserted.



4	Add a ***Method*** and ***Start Condition*** to the ***Method*** list. See *Section 7.2 Create a method queue, on page 199*.
5	***Save*** the method queue.

Delete a method from a method queue

The following table describes how to delete a method from the method queue for a system.

Step	Action
1	Open the Method Queue, see *Opening a method queue, on page 202*. *Result:* The ***Method Queue*** dialog opens with the details for the chosen method queue.

Step	Action
2	In the ***Method Queue*** dialog, select the method to be removed by clicking on its name in the ***Method*** list.
3	Delete the selected row by clicking on the ***Delete*** button. *Result:* The method will be deleted from the method queue.
4	***Save*** the method queue.

Change order of methods in a method queue

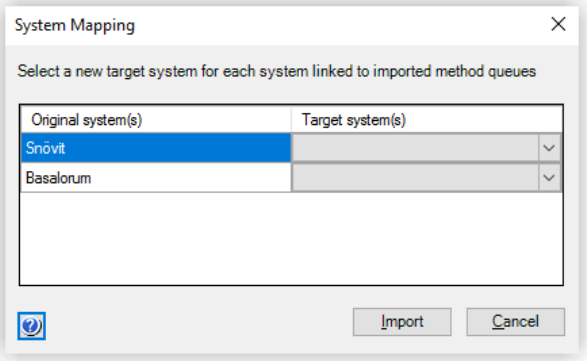
The following table describes how to change the order of methods in an existing method queue.

Step	Action
1	Open the method queue, see *Opening a method queue, on page 202*. *Result:* The ***Method Queue*** dialog opens with the details for the chosen method queue.
2	In the ***Method Queue*** dialog, select a method to be moved by clicking on its name in the ***Method*** list.
3	To move the selected method up in the ***Method*** list, click the ***Move Up*** button. *or* To move the selected method down in the ***Method*** list, click the ***Move Down*** button.
4	To change the order of further methods, repeat steps 2 and 3.
5	***Save*** the method queue.

7.4 Import and export method queues

Import a method queue

Method queues can be imported into UNICORN. The following table describes how to do this.

Step	Action
1	Select ***File*** →***Import*** →***Method Queue(s)...*** *Result:* The ***Import*** dialog opens.
2	Browse to the method queue file (****.UMQ***) in the ***Import*** dialog.
3	**a.** Select the file. **b.** Click ***Open***. **c.** Select the ***Import location*** of choice. *Result:* The ***System Mapping*** dialog box opens.
4	 Select a ***Target system*** for each ***Original system*** listed. ***Note:*** *To proceed with the import, there must be at least the same number of systems active in UNICORN as the number of original systems.* ***Note:*** *The **System Mapping** dialog appears once for every method queue selected.*
5	Click the ***Import*** button to import the method queue. If several method queues are selected, the import of all the method queues starts.

Note: *Method queues may also be imported by importing an entire folder, see* *Import folders, on page 208**.*

Note: *During the method queue import an* ***Import Report*** *is generated at the target location.*

Export a method queue to UNICORN

The following table outlines the steps needed to export a method queue for later import into UNICORN.

Step	Action
1	In the Method Navigator, select the method queue to be exported. ***Note:*** *Several method queues in the same folder can be selected and exported at the same time. You can select several method queues at once by using the* **Shift** *or* **Ctrl** *key when selecting.*
2	Choose ***File*** →***Export*** →***to UNICORN*** →***Method Queue(s)...*** *Result:* An ***Export*** dialog opens
3	Click ***OK*** to continue the export. *Result:* The ***Export to Another UNICORN Database*** dialog box opens.
4	Choose a file name and location and click the ***Save*** button to save the ***.UMQ*** file. ***Note:*** *Export of multiple method queues results in the generation of multiple export files.*

Note: *Method queues may also be exported by exporting an entire folder, see* *Export folders, on page 208**.*

Note: *During the method queue export an* ***Export Report*** *is generated at the target location.*

8 Import and export folders

Import folders

Folders containing for example methods, method queues and/or DoE Results can be imported into UNICORN.

Step	Action
1	Click ***Import*** and then click ***Folder(s)*** on the ***File*** menu. *Result:* The ***Import*** dialog box opens.
2	Browse to the folder of interest (containing .UFol files) in the ***Import*** dialog box.
3	Open the folder by selecting it and clicking ***Open***.
4	If the folder contains method queues a ***System Mapping*** dialog box opens. Select a ***Target system*** for each ***Original system*** listed. ***Note:*** *To proceed with the import, there must be at least the same number of systems active in UNICORN as the number of original systems.*
5	Click ***Import*** to import the folder with all its content. ***Note:*** *For a folder containing several method queues, the **System Mapping** dialog box is only shown once.*

Note: *During the import of a folder an **Import Report** is generated at the target location.*

Export folders

Folder export is recommended for bulk export of all items residing in a folder. During folder export everything in the folder is exported regardless of application or object navigator filter settings. If the folder contains compound objects, such as method queues or DoE results, all individual items located outside the exported folder will be placed together with the exported folder.

The following table outlines the steps needed to export a folder containing for example methods, method queues and/or DoE Results for later import into UNICORN.

Step	Action
1	**a.** Select a Folder in the ***Object Navigator***. To select several Folders, press **Shift** while you click the folders. **b.** Click ***Export***, then ***UNICORN*** and then ***Folder(s)*** on the ***File*** menu. *Result:* An ***Export*** dialog box opens.
2	Click ***OK*** to continue the export. *Result:* The ***Export to Another UNICORN database*** dialog box opens.
3	Choose a location and click ***Save*** to save the folder. *Result:* • The folder is exported as a ***.UFol*** file. • An ***Export Report*** is generated at the target location. ***Note:*** *Export of multiple folders results in the generation of a single export file.*

9 Column Handling

About this chapter

The ***Column Handling*** tool in UNICORN enables handling of Column types and, if enabled, handling of columns using the ***Column Logbook***. The ***Column Handling*** tool can be opened from all available modules in UNICORN.

This chapter gives an overview of the ***Column Handling*** and ***Column Logbook*** tools.

In this chapter

9.1 Overview

Introduction

This section gives an overview of the ***Column Handling*** tool and suggests a workflow when working with Column types and columns.

Definitions

Term	Description
Column type	A group of columns that share several common features, for example, hardware, resins, etc.
Column	A column of a Column type. The bed height of a column can be user-defiend and thefore can differ from the bed height of the Column type. ***Note:*** *Changes in the bed height automatically changes the column volume.*

Example

A laboratory has two Mono Q™ HR 16/10 columns used in different projects but both belongs to the same Column type Mono Q HR 16/10 and share several common features. However each column can be treated and logged separately using ***Column Logbook*** in UNICORN, assuming this option is enabled.

Note: *Generic methods are created with the column volume of the Column type. At the start of a run, the column volume is, if different from the Column type, adjusted according to the column chosen for the run. If it is necessary to use a custom packed column (e.g. an AxiChrom column), create a specific Column type with the defined hardware and resins, and with a default column volume. Specific bed height can be set for each column.*

Open the Column Handling dialog box

To open the ***Column Handling*** dialog box:

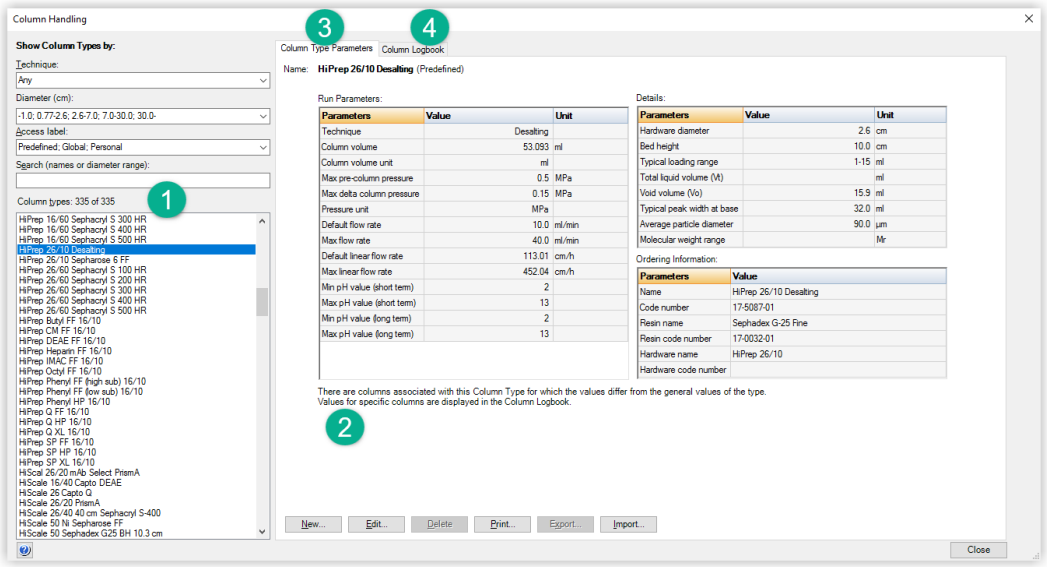
- select ***Tools*** →***Column Handling...*** in any of the UNICORN modules

  or

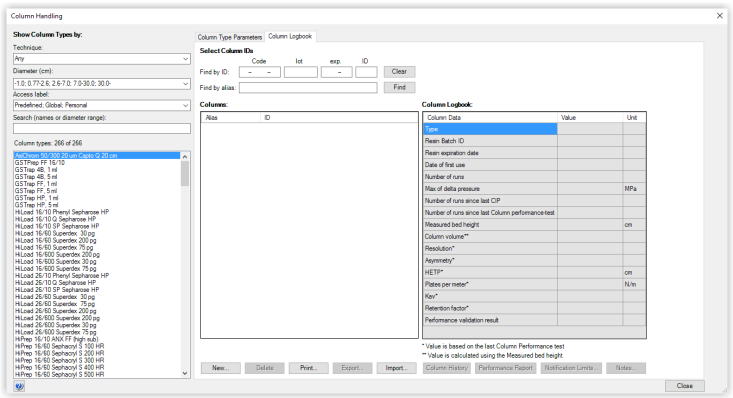
- click the ***Column Handling*** icon in the ***Toolbar*** where available

The Column Handling dialog box

The illustration below shows the ***Column Handling*** dialog box displaying the ***Column Type Parameters*** tab.



Part	Function
1	***Select Column Types by*** area: Shows the available Column types in the ***Column Handling*** dialog box. The list can be filtered to display Column types for a specific technique, diameter, and/or access label.
2	***Column Type Parameters*** tab: Shows the parameters for the selected Column type in the ***Column types*** list. See *Section 9.2 Handling Column types, on page 216* for more information.
3	Additional information is provided for some Column types.

Part	Function
4	***Column Logbook*** tab: Shows available columns for the selected Column type in the ***Column types*** list. The specific parameters for the selected column in the ***Columns*** list are shown in the ***Column Logbook*** area to the right. See *Section 9.3 Handling columns, on page 226* for more information. ***Note:*** *A **Column Logbook** e-license is needed for the **Column Logbook** functionality.* 

Main Column Handling tasks

There are many possible workflows when working with Column types and columns in UNICORN. The table below lists the main tasks that are performed in the ***Column Handling*** tool or the ***Select columns dialog box*** in the ***Start Protocol*** (when starting the run in ***System Control***).

When working with...	the main tasks are...

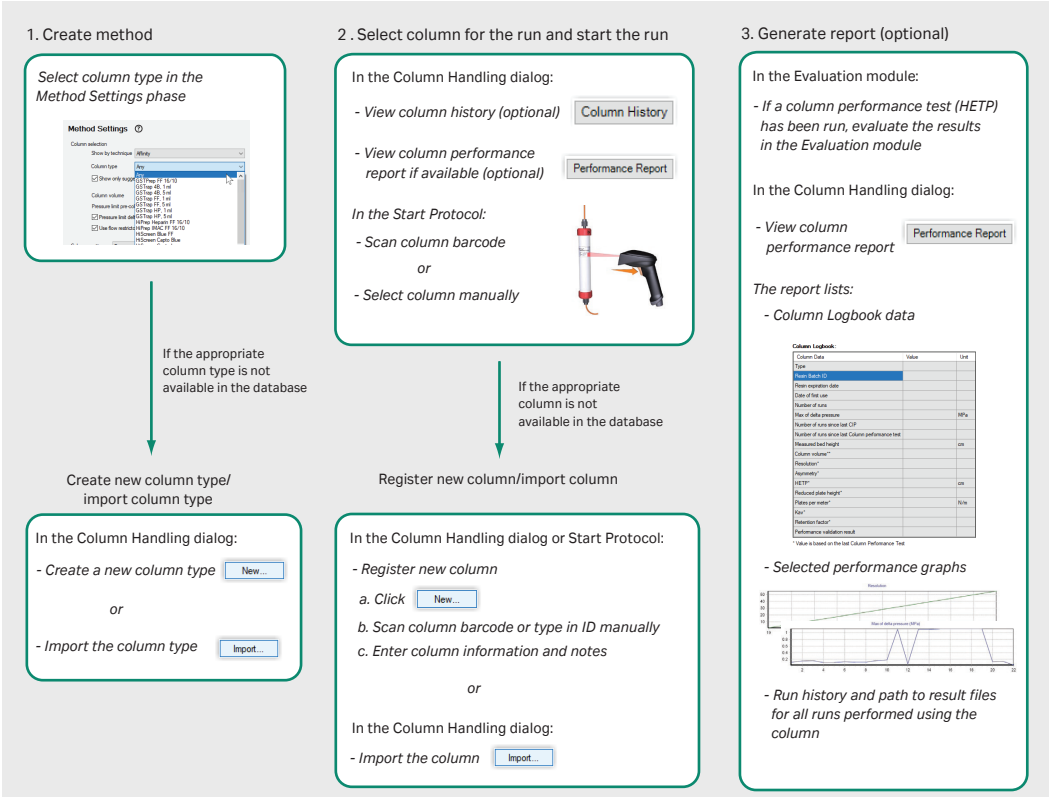
Column types	• *Create new Column types*
	• *Import/export Column types*
	  - Used to transfer Column type data between different databases
	• *Edit Column types*
	  - Edit parameters and delete Column types
	• *Print information about Column types*

When working with...	the main tasks are...
Columns	• *Register new columns* - scan or manually type in the barcode - add notes (optional) ***Tip:*** *New columns can be registered using the* ***Column Handling*** *dialog box or before the run is started.* • *Select columns to be used in the run* ***Tip:*** *Columns to be used for a specific run can only be selected via the* ***Start Protocol****. Before selecting a previously used column, it is possible to view the run history and a performance report (if available) in the* ***Column Handling*** *dialog box.* • *Edit columns* - add/edit notes - set notification limits - delete unused columns • *Print column information* • *Generate a performance report* • *Export and import columns from UNICORN*

Tip: *A new Column type can be created from evaluation after running intelligent packing on ÄKTA pilot 600.*

Illustration of Column Handling workflow

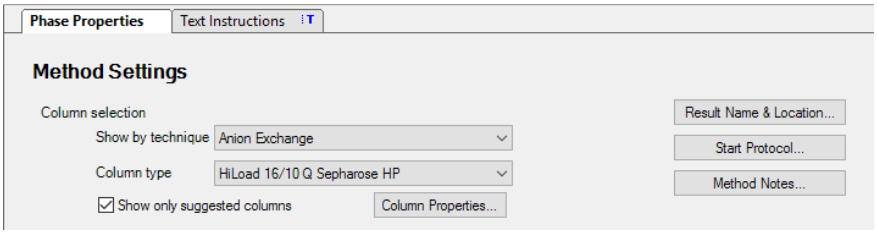
The illustration below shows a possible workflow when working with Column types and columns:



9.2 Handling Column types

Introduction

When you create a new method and select a Column type in the ***Method Settings*** phase or in the wizard window, the CV and column pressure limits are automatically set. For systems using predefined methods, the default flow rate and the pressure for the column are automatically set as well. Most of the work regarding handling of Column types is performed in the ***Method Editor***. The Column type to be used in a method is selected when creating the method as shown in the illustration below.



Column types are either globally available to all users, or only personally available. A number of Column types are predefined in UNICORN (see below for more information about predefined Column types).

Note: *When creating methods , the CV of the Column type is used. When performing method runs and a specific column is chosen, the column-specific CV is used. Otherwise, the Column type parameter values are used.*

This section describes how to add, edit and delete Column types. It also describes how to import and export Column types and how to print information about selected Column types.

Predefined Column types

A number of Column types from Cytiva are predefined in UNICORN. For each Column type, as many columns as needed can be registered. Parameters for the predefined Column types can be edited by saving the Column type with a new name and as a ***Personal*** or ***Global*** column type. The complete list of predefined Column types can be found in the ***Column Handling*** dialog box.

Create a new Column type in Column Handling tool

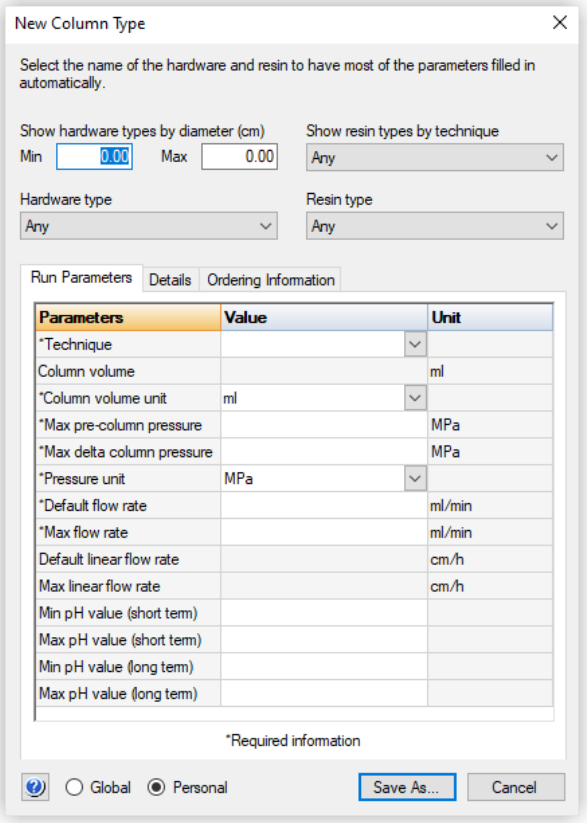
Follow the instructions to add a new column type with the ***Column Handling*** tool:

Step	Action
1	In the ***Column Type Parameters*** tab in the ***Column Handling*** dialog box, click ***New***.

Step	Action

Result:

The ***New Column Type*** dialog box opens.



2 If adding a Column type for which the column hardware and resin are not made by Cytiva, continue to step 4.

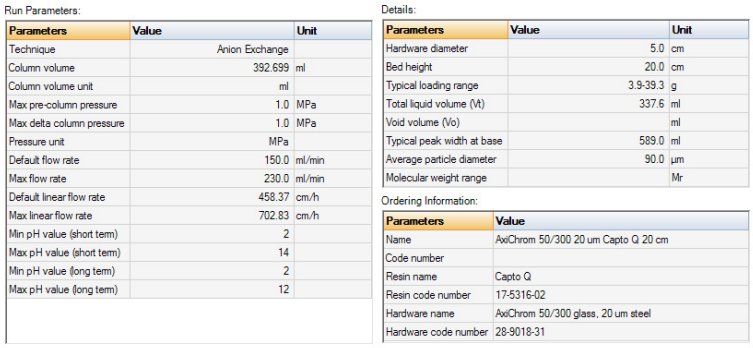
3 **a.** Select the ***Hardware type*** for the new Column type in the drop-down list.

To filter the drop-down list to only show hardware types with certain diameters, enter the diameter range in cm in the ***Min*** and ***Max*** fields for ***Show hardware types by diameter (cm)*** above.

b. Select the ***Resin type*** for the new Column type in the drop-down list.

To filter the drop-down list to only show resin types for a specific separation technique, choose the appropriate technique in the ***Show resin types by technique*** drop-down list above.

Step	Action
	Result: The following parameters are automatically filled in (can be edited as required):



Run Parameters:

Parameters	Value	Unit
Technique	Anion Exchange	
Column volume	392.699	ml
Column volume unit	ml	
Max pre-column pressure	1.0	MPa
Max delta column pressure	1.0	MPa
Pressure unit	MPa	
Default flow rate	150.0	ml/min
Max flow rate	230.0	ml/min
Default linear flow rate	458.37	cm/h
Max linear flow rate	702.83	cm/h
Min pH value (short term)	2	
Max pH value (short term)	14	
Min pH value (long term)	2	
Max pH value (long term)	12	

Details:

Parameters	Value	Unit
Hardware diameter	5.0	cm
Bed height	20.0	cm
Typical loading range	3.9-39.3	g
Total liquid volume (Vt)	337.6	ml
Void volume (Vo)		ml
Typical peak width at base	589.0	ml
Average particle diameter	90.0	µm
Molecular weight range		Mr

Ordering Information:

Parameters	Value
Name	AxiChrom 50/300 20 um Capto Q 20 cm
Code number	
Resin name	Capto Q
Resin code number	17-5316-02
Hardware name	AxiChrom 50/300 glass, 20 um steel
Hardware code number	28-9018-31

4 Enter the remaining parameter values for the new Column type in the ***Run Parameters***, ***Details*** and ***Ordering Information*** tabs. Fields marked with ***** must be filled in.

Values in the gray fields are automatically calculated based on the related parameter values.

For systems that cannot utilize the delta column pressure signal, it is recommended to use the same pressure as the pre-column pressure.

5 Select whether the the new Column type should be ***Global*** (available for all users) or ***Personal*** (only available for the current user).

6 Click ***Save As...*** to save the Column type.

Result:

The ***Save As*** dialog box opens.

7 Type in a ***Column type name*** and click ***Save.***

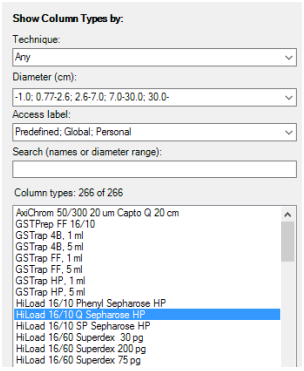
Result:

The Column type is saved in the database and displayed in the ***Column types*** list.

Edit parameters for a Column type

Follow the instructions to edit parameters for a Column type:

Step	Action
1	Choose filtering options from the drop-down lists to show the Column types by ***Technique***, ***Diameter*** or ***Access label***. Select the appropriate Column type for which to edit parameters in the ***Column types*** list.



Result:

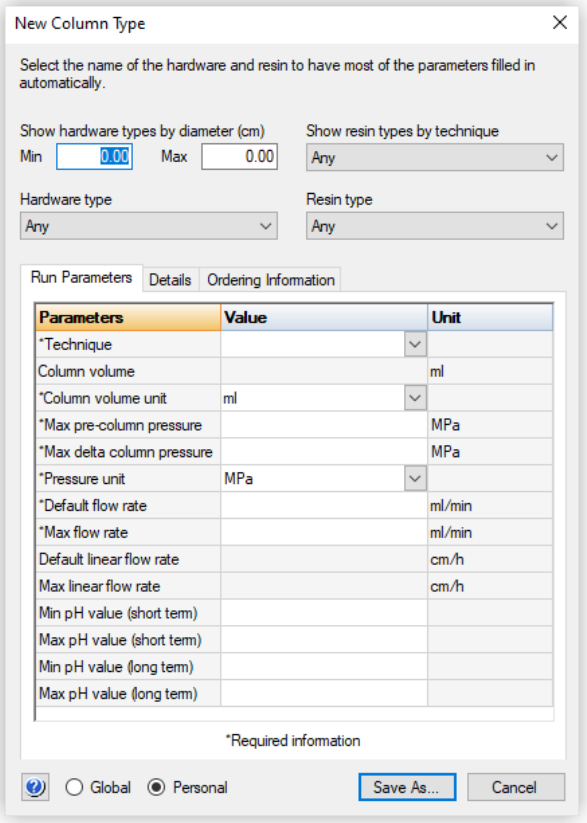
The parameters for the selected Column type are displayed in the ***Column Type Parameters*** tab to the right.

2	In the ***Column Type Parameters*** tab, click ***Edit***.

Step	Action

Result:

The ***Edit Column Type*** dialog box opens.



3 Edit the Column type parameters as appropriate on the ***Run Parameters***, ***Details*** and ***Ordering Information*** tabs.

4 Select whether the edited Column type should be ***Global*** (available for all users) or ***Personal*** (only available for the current user).

5 When parameters for a predefined Column type are edited, the Column type must be saved with a new name.

a. Click ***Save As...*** to save the edited Column type.

Result:

The ***Save As*** dialog box opens.

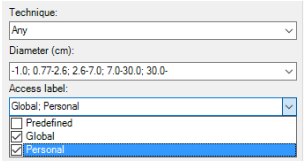
b. Edit the ***Column type name*** and click ***Save.***

Step	Action
	Result: The Column type is saved in the database and displayed in the ***Column types*** list.
6	When parameters for a ***Global*** or ***Personal*** Column type are edited, the Column type can be saved with a new name (see step 5 above) or the changes can be applied to the current Column type name. **a.** Click ***Save***. *Result:* The changes for the Column type are saved. ***Note:*** *When editing parameters for* ***Global*** *Column types, it is recommended to save the edited Column type with a new name. Other users may otherwise not be aware that the parameters have been changed for that Column type.* ***Note:*** *Methods that use the edited Column type must be updated.*

Delete Column types

Note: *It is not possible to delete predefined Column types from the database. If a Column type has any registered columns, it cannot be deleted unless the columns are first deleted. See* *Section 9.3 Handling columns, on page 226* *for information about how to delete columns. If a column of a certain type has been used, it is not possible to delete either the column or the Column type.*

Follow the instructions to delete ***Global*** and ***Personal*** Column types from the database:

Step	Action
1	In the ***Select Column Type*** area, clear the ***Predefined types*** box in the ***Access label*** drop-down list.  *Result:* Only ***Global*** and ***Personal*** Column types are displayed in the ***Column types*** list.

Step	Action
2	Select the Column type(s) to be deleted in the ***Column types*** list. To select several Column types use the **Ctrl** or **Shift** keyboard keys.
3	In the ***Column Type Parameters*** tab, click ***Delete***. *Result:* The ***Confirm Column Type Delete*** dialog box opens.
4	Click ***Yes*** to delete the Column type. *Result:* The Column type is permanently deleted from the database.

Export Column types

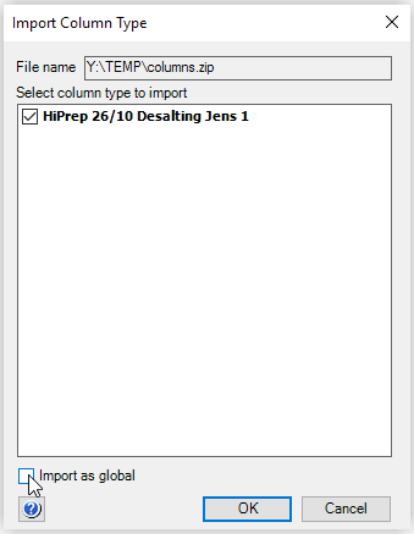
Note: *It is not possible to export predefined Column types from the database.*

Follow the instructions to export ***Global*** and ***Personal*** Column types from the database:

Step	Action
1	In the ***Select Column Type*** area, clear the ***Predefined types*** box in the ***Access label*** drop-down list. *Result:* Only ***Global*** and ***Personal*** Column types are displayed in the ***Column types*** list.
2	Select the Column type(s) to be exported in the ***Column types*** list. To select several Column types use the **Ctrl** or **Shift** keyboard keys.
3	In the ***Column Type Parameters*** tab, click ***Export...***. *Result:* The ***Export Column Type*** dialog box opens.
4	Select in which folder to save the information and type a name for the zip file to be exported. *Result:* The Column type information is exported. This information can be imported into another database.

Import Column types

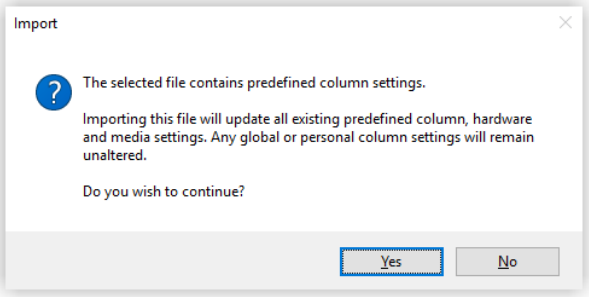
Follow the instructions to import Column types into the database:

Step	Action
1	In the ***Column Type Parameters*** tab in the ***Column Handling*** dialog box, click ***Import***. *Result:* The ***Import*** dialog box opens. ***Note:*** *It is not possible to import Column types from UNICORN 5. They have to be re-created in UNICORN 7.10.*
2	Locate the zip file with the Column type information to be imported and click ***Open***. *Result:* The ***Import Column Type*** dialog box opens displaying the names of the Column types included in the zip file. 
3	Make sure that the check boxes in front of the Column types to be imported are checked. If a Column type is not imported, clear the corresponding check box.
4	Check the ***Import as Global*** box if the Column types to be global (i.e., available for all users) when imported. Otherwise, the Column types are imported as personal Column types for the current user.
5	Click ***OK***. *Result:* The Column types are imported into the database.

Step	Action
	Note: *If a Column type to be imported has the same name as an existing Column type in the database, you will be prompted to type a new name for that Column type. Type in a name and click* ***OK****.*

Import new column list

Follow the instructions to import a new column list into the database:

Step	Action
1	In the ***Column Type Parameters*** tab in the ***Column Handling*** dialog box, click ***Import***. *Result:* The ***Import*** dialog box opens.
2	Locate the zip file with the column list to be imported and click ***Open***. *Result:* The ***Import*** message box opens, explaining what happens when the zip file is imported. 
3	Click ***Yes***. *Result:* The new list of predefined Column types is imported into the database.

Print information about Column types

Follow the instructions to print information about Column types:

Step	Action
1	Select the Column type(s) for which to print information in the ***Column types*** list. To select several Column types use the **Ctrl** or **Shift** keyboard keys.
2	In the ***Column Type Parameters*** tab, click ***Print...*** *Result:* The ***Print*** dialog box opens.
3	Select ***Printer***.
4	Select the Column types for which information is to be printed: **a.** ***All types***: Prints information for all Column types in the database **b.** ***All shown types***: Prints information for all Column types displayed in the ***Column types*** list **c.** ***Selected types***: Prints information for the Column type(s) selected in the ***Column types*** list
5	Select which type of information to include when printing the information: **a.** Check the ***Include the type's parameters*** box to include the information from the ***Run parameters***, ***Details*** and ***Ordering Information*** fields in the ***Column Type Parameters*** tab. **b.** Check the ***Include the associated columns*** box to include the Column ID and alias of the columns registered for the Column type. Check the ***Include column's parameters*** to include the parameters for each column registered for the Column type(s).
6	Click ***OK***. *Result:* The selected information for the Column type(s) is printed.

9.3 Handling columns

About this section

Columns are handled on the ***Column Logbook*** tab in the ***Column Handling*** dialog. The ***Column Logbook*** enables the run history for a column to be traced, for example, how many CIP runs have been performed using that column. Columns are always connected to a Column type.

Note: *The **Column Logbook** tab is only displayed if this option has been selected and a **Column Logbook** e-license exists.*

Working with columns is primarily done in the ***Method Editor*** and ***System Control***, depending on the task to be performed.

In this section

9.3.1 Column identification

Matrix barcode

Most prepacked Cytiva columns are marked with a matrix barcode on the column label. This barcode can be scanned using the 2D barcode scanner to register new columns or to find columns in the database.

Columns can also be labeled with UniTag labels. A UniTag label is a unique identifier for columns that are not prelabelled with a matrix barcode, such as HiTrap™ columns, manually packed columns or columns from other sources. A number of UniTag labels may be supplied with the system, and they can also be purchased separately.

The diagram below shows an example of a column label and a UniTag label with their matrix barcodes.

Column label UniTag label

9.3.2 Register a new column

Introduction

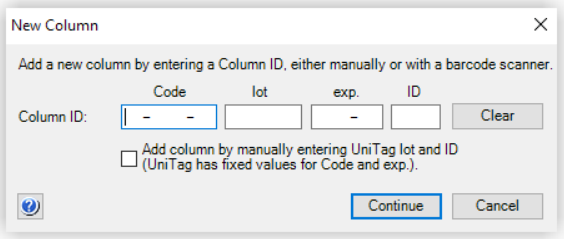
In order to take advantage of the column handling features of UNICORN, each column needs to be registered in the software.

Note: *For some systems, it is possible to create a column from the Evalution module after packing. Then it is possilbe to save the column performance results to this specific column. In all other case, make sure that the column is registered before a column performance test is performed. Otherwise, the results are not entered in the* ***Column Logbook****. It is not possible to enter the performance test results afterwards.*

Register a column in Column Logbook

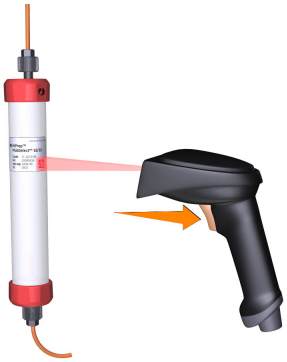
The following table describes how to register an column in the ***Column Logbook***.

Step	Action
1	Select the ***Column Logbook*** tab and then click ***New.*** *Result:* The first ***New Column*** dialog box opens.
2	Register the column using the 2D barcode scanner as follows: **a.** Make sure that the mouse pointer is placed in the first position of the ***Code*** field. **b.** Point the 2D barcode scanner towards the data matrix tag on the column label or the UniTag label.



Step	Action

c. Press and hold the trigger.



d. When the 2D barcode scanner beeps, the column ID is registered and the second ***New Column*** dialog box opens.

3 If no 2D barcode scanner is available, enter the column ID manually:

a. If the column has a column label, enter the column ID shown in the ***Code*** field.

b. If the column has a UniTag label, check the box ***Add column by manually entering UniTag lot and ID*** and manually enter the number for the ***lot***, expiration date (***exp***) and ***ID*** fields.

c. Click ***Continue*** to open the second ***New Column*** dialog box.

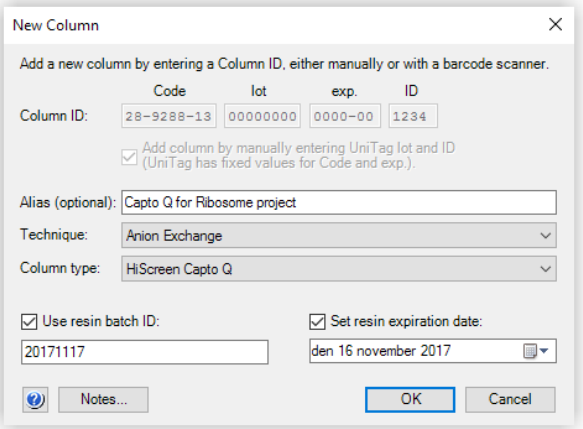
Note:

*The **lot** field should contain eight digits, and the **ID** field should contain four digits. If the lot or ID numbers of the column contains fewer than eight or four digits respectively, insert leading zeros before the number.*

Note:

*If the column has no Cytiva label and you have run out of UniTag labels, check the box **Add column by manually entering UniTag lot and ID**, then enter an arbitrary lot and ID. This procedure is possible, but not recommended.*

Step	Action
4	In the second ***New Column*** dialog box:



a. Enter an ***Alias*** (optional).

Tip:

Alias can be used for easy identification of a column.

b. Select ***Technique*** and ***Column type***.

Note:

For prepacked Cytiva columns with a matrix barcode, these are filled in automatically.

c. Check the ***Use resin batch ID*** and type in the batch number of the resin.

d. Check the ***Set resin expiration date*** and select expiration date for the resin to get a notification in UNICORN when this date is reached. If an expiration date is set in the column ID field, this date is suggested in the resin expiration date. The expiration date of the resin overrules the expiration date of the column.

Note:

The expiration date cannot be set or changed after a column has been registered.

e. Enter notes for the column by clicking the ***Notes...*** button and enter notes in the ***Notes*** dialog box that opens.

f. Click ***OK.***

Result:

The entered information is saved and the registered column is displayed in the ***Column Handling*** dialog box.

9.3.3 Find a column

Introduction

Many features of the ***Column Handling*** tool require an column to be selected. This section describes how to find a column.

Find and select a column

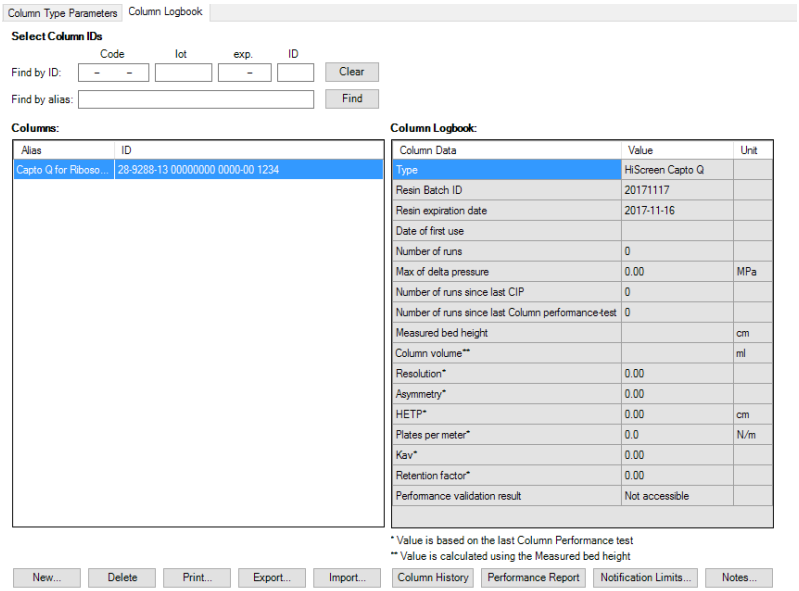
The table below describes how to find/select a registered column in the ***Column Logbook***:

Step	Action
1	Select the ***Column Logbook*** tab.
2	Filter the list of Column types by: **a.** choosing the required technique in the drop-down menu ***Technique*** *and/or* **b.** choosing the diameter in the drop-down menu ***Diameter*** *and/or* **c.** choosing the access labels in the drop-down menu ***Access label*** *and/or* **d.** enter a search criteria in the ***Search*** field. Then select the Column type to which the column belongs.
3	To select several Column types, use the **Ctrl** and **Shift** keyboard keys.

Step	Action

Result:

The columns registered for the selected Column type(s) are displayed in the ***Columns*** list. The ***Column Logbook*** area to the right shows parameters and information for the selected column.



Tip:

To show all registered columns, select all the available Column types by checking the boxes for ***Predefined****,* ***Global*** *and* ***Personal*** *types, then select all the* ***Column Types*** *in the list.*

4

- If you have a short list of columns registered for the Column type, just select the appropriate column in the ***Columns*** list. To select several columns, use the **Ctrl** and **Shift** keyboard keys.
- If you have many columns registered, find and select the appropriate column as described below:
  - position the cursor in the first position of the ***Code*** field, scan the column barcode or UniTag and click ***Find***

    or
  - type in the barcode in the ***Find by ID*** field and click ***Find***

    or
  - type in the alias in the ***Find by alias*** field and click ***Find***

    *Result:*The column is selected in the ***Columns*** list.

9.3.4 Edit columns

Introduction

This section describes the ways in which columns can be edited. This includes adding and editing notes, setting notification limits, and deleting columns.

Add/edit notes for a column

The table below describes how to add/edit notes for an column:

Step	Action
1	Select the column for which to add/edit notes in the ***Columns*** list in the ***Column Logbook*** tab. See *Section 9.3.3 Find a column, on page 231* for information about how to find and select a column.
2	In the ***Column Logbook*** tab, click ***Notes***. *Result:* The ***Notes*** dialog box for the selected column opens.
3	Add/edit notes by typing in the dialog box and click ***OK***. *Result:* The notes for the column are updated.

Set notification limits for a column

Notification limits can be set for columns. Once the limit is reached, the user receives a message stating what action should be taken before the column is used. Examples of such limits are:

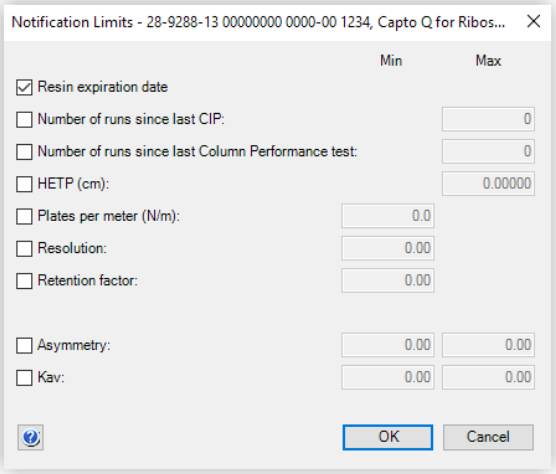
- ***Resin expiration date***
- ***Number of runs since last CIP***
- ***Number of runs since last Column Performance test***

Note: *Warnings related to the number of times a column has been used, for example since the last CIP was performed, are only issued when the start protocol is performed. In a method queue this may not always be shown for every run. Each run will however be noted in the column history.*

The table below describes how to set notification limits for a column:

Step	Action
1	Select the column for which to set ***Notification Limits*** in the ***Columns*** list in the ***Column Logbook*** tab. See *Section 9.3.3 Find a column, on page 231* for information about how to find and select a column.
2	In the ***Column Logbook*** tab, click ***Notification Limits***.

Step	Action
	Result: The ***Notification Limits*** dialog box for the selected column opens.



3	Check the appropriate boxes and enter notification values. When the values are reached or a value is outside the defined range, a warning will be displayed that action should be taken.
4	Click ***OK***. *Result:* The settings are saved and the dialog box is closed.

Delete columns

The table below describes how to delete an column from the database:

Step	Action
1	Select the column to be deleted in the ***Columns*** list in the ***Column Logbook*** tab. See *Section 9.3.3 Find a column, on page 231* for information about how to find and select a column. To select several columns in the ***Columns*** list, use the **Ctrl** and **Shift** keys.
2	In the ***Column Logbook*** tab, click ***Delete***. *Result:* The ***Confirm Column Delete*** dialog box opens.
3	Click ***Yes*** in the ***Confirm Column Delete*** dialog box.

Step	Action
	Result: The column is deleted. ***Note:*** *Used columns cannot be deleted.*

9.3.5 Export and import columns

Introduction

The information for columns is stored in the UNICORN database. This information may be exported to a zip file in order to move the information to another UNICORN installation. This section describes how to export columns from UNICORN and how to import previously exported columns.

Export columns

Columns can be exported from the database to a zip file. The columns can then be imported to another database if appropriate.

The table below describes how to export columns from the database:

Step	Action
1	Select the column(s) to be exported in the ***Columns*** list in the ***Column Logbook*** tab. See *Section 9.3.3 Find a column, on page 231* for information about how to find and select a column. To select several columns in the ***Columns*** list, use the **Ctrl** and **Shift** keys.
2	In the ***Column Logbook*** tab, click ***Export***. *Result:* The ***Export Column Type*** dialog box opens.
3	Select in which folder to save the information and type a name for the zip file to be exported. *Result:* The column information is exported. The column information can be imported into another database.

Import columns

Columns that have been exported and saved locally can be imported into another database.

The table below describes how to import column information to a database:

Step	Action
1	In the ***Column Logbook*** tab, click ***Import***. *Result:* The ***Import*** dialog box opens.
2	Locate the zip file with the Column type information to be imported and click ***Open***.

Step	Action
	Result: The ***Import Column*** dialog box opens displaying the barcodes and aliases of the columns included in the *.zip file.
3	Make sure that the check boxes in front of the columns to be imported are checked. If a column should not be imported clear the corresponding check box.
4	Click ***OK***. *Result:* The columns are imported into the database. ***Note:*** *If a column to be imported has the same barcode or alias name as an existing column in the database, a dialog box will be displayed saying that the column already exists in the database and that it will not be imported.*

Note: *The column-specific parameter values are visible in the columns created in earlier UNICORN versions, but only the values from the Column type are used by UNICORN.*

9.3.6 Print and view column information

Introduction

This section describes how to view the run history for an column, and how to print column information.

View column history

It is possible to view the run history for a column to see how many runs that have been performed using the column. The path to the result files for each run is also displayed. If the run was a column performance test or CIP run, this is shown as a remark.

The table below describes how to view the ***Column History*** for a column:

Step	Action
1	Select the column for which to view ***Column History*** in the ***Columns*** list in the ***Column Logbook*** tab. See *Section 9.3.3 Find a column, on page 231* for information about how to find and select a column.
2	In the ***Column Logbook*** tab, click ***Column History***. *Result:* The ***Column History*** dialog box for the selected column opens. The runs performed using the column are listed. The date, result name and location and any remarks for the run are displayed.
3	Click ***Close*** to close the dialog box.

Print information about columns

The table below describes how to print information for columns:

Step	Action
1	Select the appropriate column(s) for which to print information in the ***Columns*** list in the ***Column Logbook*** tab. See *Section 9.3.3 Find a column, on page 231* for information about how to find and select a column. To select several columns in the ***Columns*** list, use the **Ctrl** and **Shift** keys.
2	In the ***Column Logbook*** tab, click ***Print***. *Result:* The ***Print*** dialog box opens.
3	Select ***Printer***.
4	Select for which column(s) to print information:

Step	Action
	a. ***Selected columns***: Prints information for the column(s) selected in the ***Columns*** list **b.** ***All shown columns***: Prints information for all columns displayed in the ***Columns*** list **c.** ***All columns***: Prints information for all columns in the database
5	Click ***OK***. *Result:* The Column ID, alias and parameters for the columns are printed.

9.4 Column performance

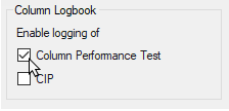
Introduction

Column performance can be tested by measuring the height equivalent to a theoretical plate (***HETP***) and asymmetry factor (***As***) values. Tests should be run directly after packing or obtaining a new column, regularly during the lifetime of the column and when separation performance is seen to deteriorate. By regularly monitoring the performance of a column, UNICORN can generate appropriate warnings when a cleaning procedure needs to be applied, or even when the column lifetime is approaching its end. For a description of how to set such notification limits see *Set notification limits for a column, on page 233*.

This section describes the workflow to run a ***Column Performance Test***, and how to generate a performance report for a specific column.

Column performance test

The following table describes the workflow for generating and analyzing a ***Column Performance Test*** result.

Step	Action
1	Create a ***Column Performance Test*** method, or a method containing a column performance test. For details how to create methods see *Chapter 3 Create and edit methods, on page 23*. ***Note:*** *For methods created using phases, the option **Enable logging of Column Performance Test** should be automatically selected in the Phase Properties for the **Method Settings** phase when using a predefined column performance method. This can be deselected if logging of the performance test is not desired, but it should normally be kept selected.*  *For wizard created methods and text created methods the **Enable logging of Column Performance Test** is not automatically selected, and must be selected in the method settings phase for a column performance test to be logged.*
2	Run the method. For details on running a method, see UNICORN System Control Manual. ***Note:*** *The column must be selected when the method run is started in order to register the results from the column performance test in the column logbook. The result cannot be logged for the column at a later time.*

Step	Action
	In some systems, for example ÄKTA avant, when using a predefined intelligent packing method the Column type is requested to be selected and a column will also be requested at method start. When creating a wizard generated intelligent packing method it is an option to create separate methods for the packing and for the column performance test and therefore the Column type and the column can only be needed for the column performance test method. *Suitable samples that can be used to monitor the column performance are for example 1% acetone (measuring the absorbance at 280 nm), or 2.0 M NaCl and eluting with 0.5 M NaCl.* ***Note:*** *A sample volume between 0.5% and 3% of the column volume and a flow velocity between 15 and 30 cm/h is recommended.* *The calculated number of plates and the asymmetry factor will in part depend on the selected flow rate. To ensure that test results are comparable, always use the same flow rate and system setup for the tests.*
3	Evaluate the ***Column Performance Test***, see UNICORN Evaluation Manual.

Create a performance report

A column performance report can be created before using an column. The report shows if the column is in good condition for use. The performance report contains the following information:

- Run and performance parameters
- Notes
- Performance graphs (optional)
- Run history

The table below describes how to generate a column performance report:

Step	Action
1	Select the column for which to generate a ***Performance Report*** in the ***Columns*** list in the ***Column Logbook*** tab. See *Section 9.3.3 Find a column, on page 231* for information about how to find and select a column.
2	In the ***Column Logbook*** tab, click ***Performance Report***. *Result:* The ***Performance Report*** dialog box for the selected column opens.
3	Select ***Printer***.

Step	Action
4	Check the appropriate boxes in the ***Performance graphs*** area to include the corresponding graphs in the report. ***Note:*** *The parameters and the corresponding values from the **Column Logbook** are always included on the first page in the report together with the latest performance test results. All runs are listed in the **Run History** at the end of the report, including **Column Performance Test** and **CIP** runs which are labeled.* ***Note:*** *Not all systems have feedback on the max delta pressure.*
5	A preview of the report is shown on the right side of the dialog box. Use the buttons above the report to scroll the preview.
6	Click ***Print*** to print the information.
7	Click ***Close*** to close the dialog box.

9.5 Intelligent Packing of AxiChrom columns

Introduction

UNICORN 7.10 features a solution for Intelligent Packing of AxiChrom columns. The AxiChrom column family features hands-free packing using internal hydraulic axial compression. Intelligent packing of AxiChrom columns can be performed in several ways.

For systems that create methods using phases, the Intelligent Packing method can be performed using either a predefined Intelligent Packing method, or by creating a user defined method including an Intelligent Packing phase.

For systems that have the Method Wizard, the Intelligent Packing method is created using the Intelligent Packing Wizard.

The workflows for the systems can differ. For a predefined Intelligent Packing method the Column Peformance tests are included, but for a wizard created Intelligent Packing method, it is possible to separate the packing and the Column performance test in two different methods.

In some systems, for example ÄKTA avant, if Column Handling is used, and the method contains a Column Performance test, it is essential to use the correct Column type, previously created in column handling, in the method settings phase.

In this section

9.5.1 AxiChrom Column types and AxiChrom columns

When the evaluation of the tests is performed, the actual packed bed height should be used. This bed height will be registered in the ***Column Logbook***. The evaluation procedure is described in the UNICORN Evaluation Manual.

Create an AxiChrom Column type in Column Handling tool

Follow the instructions to create the AxiChrom Column type in column handling:

Note: *If a Column type is used when running a column packing method, the specific bed height can be set on the column to ensure that the actual packed bed height is applied when running methods using the column.*

Step	Action
1	Choose the ***Tools*** →***Column Handling*** menu command. *Result:* The ***Column Handling*** dialog box opens.
2	In the ***Column Type Parameters*** tab in the ***Column Handling*** dialog box, click ***New***. *Result:* The ***New Column Type*** dialog box opens.
3	**a.** Select the appropriate AxiChrom column hardware in the ***Hardware type*** drop-down list. **b.** Select the ***Resin type*** for the new AxiChrom Column type in the drop-down list. ***Tip:*** *Only some of the available resins are approved by Cytiva for use in the Intelligent Packing of AxiChrom columns. For systems equipped with predefined methods and phases it is possible to click the* ***Standard verified resins*** *button in the Intelligent Packing phase to view a list of the approved resins and bed heights. This is not applicable for ÄKTA pilot 600.* *Systems that use the intelligent packing wizard will be guided to the approved resins in the wizard.* *Other resins can also be selected, but the packing procedure is then performed with a set of general default settings.* *Result:* Based on the selections, some of the Column type parameters are automatically filled in.
4	Enter the remaining parameter values for the new Column type in the ***Run Parameters***, ***Details*** and ***Ordering Information*** tabs, for example

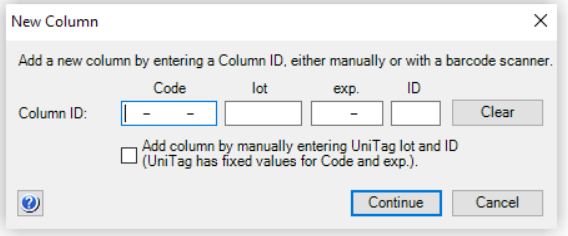
Step	Action
	a. bed height **b.** max flow rat **c.** max delta column pressure (the max delta column pressure value is used as the max pressure value for systems that cannot utilize the delta column pressure) Fields marked with ***** must be filled in. Values in the gray fields are calculated and automatically filled in based on entered values for the corresponding parameters.
5	Select whether the new Column type should be ***Global*** (available for all users) or ***Personal*** (only available for the current user).
6	Click ***Save As...*** to save the Column type. *Result:* The ***Save As*** dialog box opens.
7	Type in a ***Column type name*** and click ***Save.*** ***Tip:*** *To simplify identification, it is recommended to choose a name for the Column type composed of hardware name, resins name, and bed height. However, the Method Editor recognizes the column from the selected hardware irrespective of the name.* *Result:* The AxiChrom Column type is saved in the database and displayed in the ***Column types*** list.

Create an AxiChrom column in Column Logbook

Once a Column type consisting of the AxiChrom hardware and selected resins is created, proceed to register a column.

Step	Action
1	Select the ***Column Logbook*** tab and then click ***New.*** *Result:* The first ***New Column*** dialog box opens.

Step	Action



2 **a.** Register the column either by scanning a UniTag or manually as described in *Section 9.3.2 Register a new column, on page 228*

and

b. click ***Continue***.

Result:

The second ***New Column*** dialog box opens.

3 In the second ***New Column*** dialog box:

a. Enter an ***Alias*** (optional).

Tip:

Alias can be used for easy identification of a column.

b. Select ***Technique*** and the AxiChrom ***Column type*** you previously created.

c. Check the ***Use resin batch ID*** and type in the batch number of the resin in the column.

d. Check the ***Set resin expiration date*** and select expiration date for the resin to get a notification in UNICORN when this date is reached.

Note:

The expiration date cannot be set or changed after a column has been registered.

4 Enter notes for the column by clicking the ***Notes...*** button and typing information in the ***Notes*** dialog box that opens.

5 Click ***OK***.

Result:

The entered information is saved and the registered column is displayed in the ***Column Handling*** dialog box.

9.5.2 Predefined Intelligent Packing method

For systems that support predefined methods, use the ***Intelligent Packing*** method to prepare a method for packing the AxiChrom column.

Step	Action
1	Open a new, predefined Intelligent Packing method.
2	In ***Method Settings***: **a.** For ÄKTA avant, select the Column type to pack. **b.** Select a column position.
3	In the ***Intelligent Packing*** phase: • For ÄKTA pilot 600, select ***Hardware***, ***Bed support***, ***Resin***, and ***Target bed height***. or • Select ***Standard verified packing settings*** (default) or • ***Custom packing settings*** ***Tip:*** *The **Standard verified packing settings** have been validated by Cytiva. If you wish to use other settings, for example other resins or other bed height settings, you must select **Custom packing settings**.*
4	If you selected to enter your own ***Custom packing settings***, you can edit the following settings: • Select to - ***Pack by Packing Factor*** and choose a packing factor value *or* - ***Pack to the target bed height*** or • Change the adapter velocity and or • Optionally, select to use flow conditioning If you selected to use ***Standard verified packing settings***, proceed with the step below.

Step	Action
	Note: *It is not recommended to change the default position selections in the subsequent steps.*
5	If necessary, select the Inlets for hydraulic chamber liquid and for the mobile phase.
6	If necessary, select the column position for the hydraulic chamber (only column position A is used) ***Tip:*** *Click the **Column Connection** button to view information about the connections, including an illustration.*
7	Enter the slurry start concentration to generate a slurry recipe, which is shown in a summary in the ***Start Notes*** at the start of the method run. You can view this recipe by clicking the ***Slurry Recipe*** button. ***Note:*** • *The accuracy of the slurry preparation will affect the packed bed height.* • *This function is not available when **Custom packing settings** is selected.*
8	Verify the settings in the ***Equilibration*** phase.
9	Verify the settings in the upflow ***Column Performance Test*** phase.
10	Verify the settings in the downflow ***Column Performance Test*** phase.
11	Save the method.

Run the method and evaluate the packing

Once the ***Intelligent Packing*** method is ready, you can proceed to perform the actual packing. Refer to the AxiChrom operating instruction for instructions how to prepare the column, connect it to the chromatography system and perform the packing run.

The Intelligent Packing method includes two ***Column Performance Test*** phases, evaluating both the column upflow and downflow performance. Evaluate the results from these test as described in the UNICORN Evaluation Manual and adjust the actual bed height according to the results if necessary.

9.5.3 Wizard generated Intelligent Packing method

For systems that support the method wizard, use the ***Intelligent Packing wizard*** to prepare a method for packing the AxiChrom column.

Note: *If column handling is used, it is essential to select hardware, resins, and bed height in the wizard as well as creating the corresponding Column type and column in column handling.*

The table below describes how to create an ***Intelligent Packing*** method using the Wizard.

Step	Action
1	Create a new method in the Method Editor.
2	In the ***New Method*** window: **a.** Choose the correct system in the drop-down menu and **b.** choose ***Method Wizard*** and click ***OK***.
3	Follow the wizard to create a new method. ***Intelligent packing*** is found in the ***Special method*** droplist. Press ***F1*** to use the Wizard help if needed. ***Note:*** *The **Intelligent packing** method is only active if Intelligent packing has been chosen as a component for the system.*
4	The created method opens in the ***Method Editor***. ***Note:*** *If column handling is used, replace the Column type **Any** in the **Method Settings** phase with the corresponding Column type created in column handling. It is important to use a Column type with the same hardware, resins, and bed height as defined in the method. Information about the Column type for which the method was created can be found in **Method Notes**.* *When the Column type has been created in column handling it is found in the Column type list. See Create a new Column type in Column Handling tool, on page 216, for information on how to create Column types.*

Step	Action

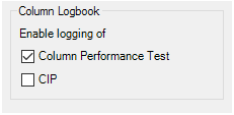
Note:

Columns that do not use the AxiChrom Master for packing will get a warning when saving the method, if the Column type was replaced with the Column type corresponding to the defined type in the intelligent packing method,. This is because the packing pressure is, and shall be, higher than the maximum pressure for the column. Save the method with correct packing pressure by pressing ***Save*** *anyway. This warning will appear every time the method is used.*

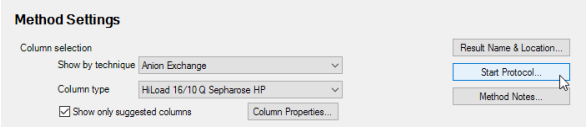
Tip:

To avoid the pressure warning it is advised to generate two separate methods, one for packing and one for column performance test, and only replace the Column type for the column performance test method.

5 Select the ***Phase properties*** tab in the ***Method Settings*** phase and select the ***Enable logging of Column Performance Test*** check box.

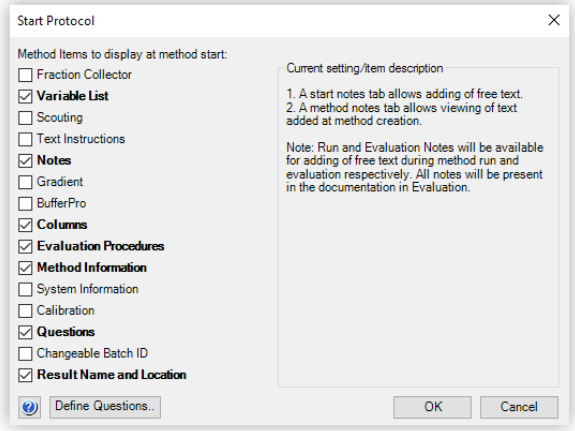


6 **a.** Click the ***Start Protocol...*** button.



Result:

The ***Start Protocol*** dialog box opens.



Step	Action
	b. Select the check boxes of the items you want to display at the start of the method. ***Tip:*** *It is recommended to select the **Notes** checkbox. This will allow the method summary page in **Method Notes** to be available at method start.*

Warning

When saving a created packing method for AxiChrom columns that do not use the AxiChrom master for packing, a warning appears if the Column type has been replaced in the method with the corresponding Column type created in column handling.

This warning is issued because UNICORN compares the Column type max pressure and the method max pressure at saving and method start, and the packing pressure for the AxiChrom column has to be higher than the maximum pressure for the column during run. Press ***Save Anyway*** to save the method and keep the correct packing pressure. This warning will appear every time the packing method is used.

Tip:

To avoid the message it is advised to generate two separate methods, one for packing and one for column performance test (packing test) and replace the Column type for the column performance test method but not for the packing method. This will also allow to generate the Column type after packing and thereby apply the measured bed height rather than the target bed height for the Column type.

7	Save the method.

Note: *It is recommended <u>not</u> to change any variables in a wizard generated intelligent packing method.*

Run the method and evaluate the packing

Once the ***Intelligent Packing*** method is ready, you can proceed to perform the actual packing of the column.

Refer to the AxiChrom operating instruction for instructions on how to prepare the column, connect it to the chromatography system and perform the packing run.

10 Text edit methods

Introduction

For some instruments it is necessary to create and edit methods or phases using ***Text Instructions***. But it can also be an option for fine-tuning or optimizing a method.

This chapter gives an overview of the ***Text Instructions*** tab and describes how to use the ***Text Instructions*** tab to create and edit methods. It also describes some text instruction applications and how to access information about the text instructions.

Note: *Text instruction may differ depending on instrument configuration. The instructions shown in this chapter are examples.*

In this chapter

10.1 Overview

Introduction

This section gives an overview of working with text instructions and a description of the ***Text Instruction*** tab.

In this section

10.1.1 Working with text instructions

Introduction

If a predefined phase in the ***Method Editor*** is selected, the corresponding phase block is selected in the ***Text Instructions*** when changing to the ***Text Instructions*** tab.

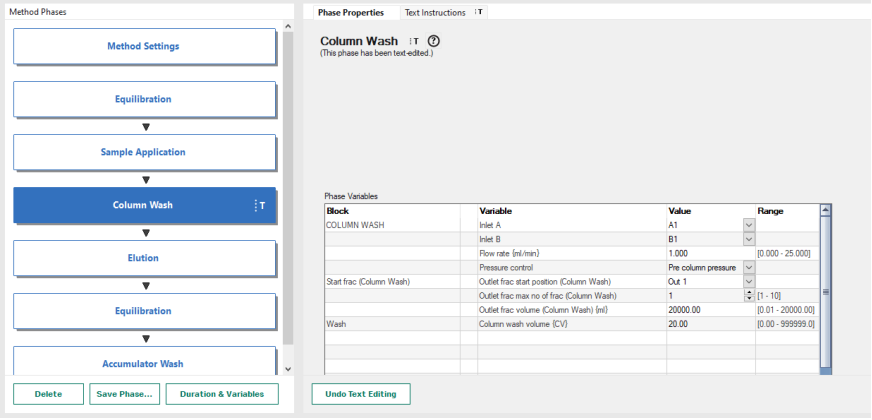
Changes made in the ***Phase Properties*** tab are automatically updated in the ***Text Instructions*** tab. If a predefined method or a method based on predefined phases is edited in the ***Text Instructions*** tab, the settings in the ***Phase Properties*** tab will be replaced by a list of the phase variables. A wizard generated method or an empty method consists of a method settings phase and a user defined phase. The user defined phase will always have a list of the phase variables.

Text editing a method

Adding, editing or deleting any blocks or instructions in a phase in the ***Text Instructions*** area means text editing of the method. When a method has been text edited, one or several of the phases displayed in the ***Method Editor*** window are affected depending on the type of editing performed.

The letter ***T*** next to the phase name in the ***Method Editor*** window indicates that the phase has been text edited.

The illustration below shows the ***Phase Properties*** tab when a phase generated method has been text edited and the indication (***T***) on the phase that has been text edited. The ***Phase Properties*** tab shows a list of phase variables.



Considerations when text editing a method

Before starting to text edit a method, consider the following:

- Editing instructions in the ***Text Instructions*** tab is only recommended for advanced users.

- If the text instructions for a phase generated method are edited manually, the phase properties will no longer show all optional settings but only the ***Phase Variables***. To restore the phase properties in a phase generated method you have to undo the edited text instructions by clicking ***Undo text editing*** which is displayed in the ***Phase Properties*** tab after text instructions have been edited.
- Several phases may be labeled as text edited when editing a single phase in the ***Text Instructions*** tab. This is the case when editing, for example, the phase ***Method Settings*** because several parameters are used in other phases.
- Do not mix text edited and non text edited phases unless you clearly understand the consequences.

10.1.2 The Text Instructions tab

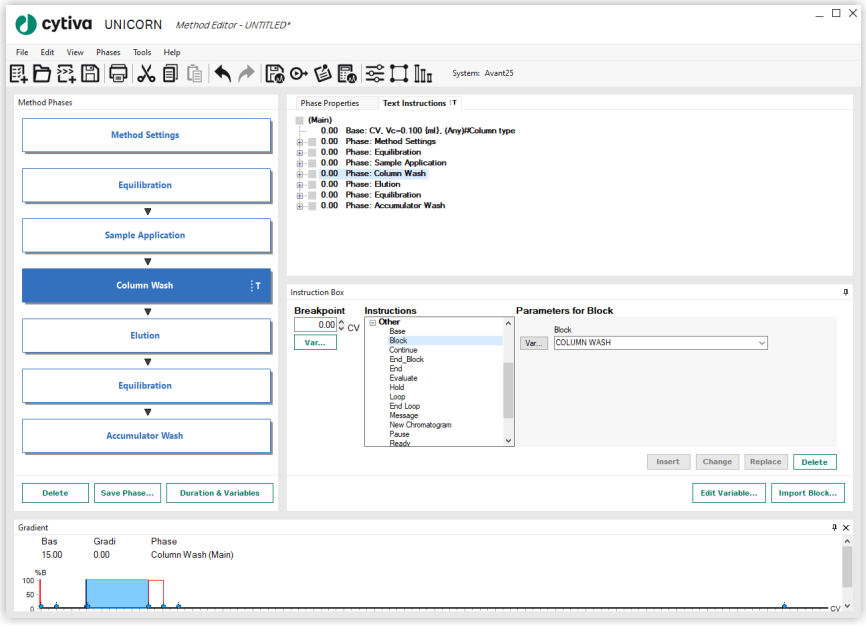
Introduction

This section gives an overview of the ***Text Instructions*** tab in the ***Method Editor*** and the structure of a text method.

Illustration of the Text Instructions tab

The ***Text Instructions*** tab consists of two areas, the ***Text Instructions*** area and the ***Instruction Box***.

The illustration below shows the ***Method Editor*** window when creating a method for a system that allows phase generated methods. The ***Text Instructions*** tab is selected. The phase ***Equilibration*** is selected in the ***Text Instructions*** area and the corresponding phase is highlighted in blue in the ***Method Outline*** and the ***Gradient*** panes.



The table below describes the different areas in the ***Text Instructions*** tab:

Area	Description
1	***Text Instructions*** area: Shows the method as a list of individual text instructions. The instructions are grouped into blocks (denoted by blue square symbols in the figure below) to obtain a logical overview of the method.

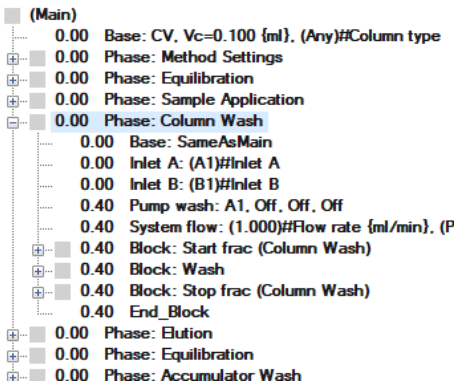
Area	Description
2	***Instruction Box***: Shows the available instructions. It can be displayed using the ***Auto Hide*** function (see *Auto hide optional panes, on page 16* for more information). Use the ***Instruction Box*** to: • insert, change, replace and delete blocks and instructions in the method • delete phases • specify breakpoints, parameters and variables ***Note:*** *It is not possible to add phases using the* ***Instruction Box****. For information about how to add phases, see Section 3.6.1 Edit the method outline, on page 51 .*

Structure of the text method

A method in the ***Text Instructions*** area consists of a ***Main*** block that contains the ***Base*** instruction (mandatory) and the appropriate phases and blocks to be used in the method. Blocks containing valid instructions are denoted by blue square symbols (for a description of other icons that may appear, see *Description of icons and text formats in the text method, on page 258*).

Structuring the method into blocks enables reuse of instructions in the method. It also makes it possible to perform a sequence of instructions using watches (see *Section 10.3.3 Watch instructions, on page 290* for more information about watch instructions).

The illustration below shows an example of a phase generated method in the ***Text Instructions*** area:



The following table describes the different parts in the method:

Part	Description
Main	The main block contains the complete method. It contains the ***Base*** instruction (mandatory) and the appropriate phases with instructions to be executed in a method.
Phase	Blocks at the highest level in the method represents the major steps in the process flow and are called phases. Each phase can contain sub-blocks, that is, blocks at a lower level. ***Note:*** *If a phase generated method has not been text edited, properties for the phase can be set in the* ***Phase Properties*** *tab.* ***Note:*** *A wizard generated method only contains the method settings phase and the user defined phase containing all steps in the process flow.* ***Note:*** *New phases can only be added to the* ***Method Outline*** *using the* ***Phase Library****. It is however possible to copy and paste an existing phase in the* ***Text Instructions*** *tab.*
Block	Each block starts with a ***Base*** instruction, continues with the appropriate instructions and always ends with an ***End_Block*** instruction.
Sub-block	A sub-block is a block at a lower level than a phase that may contain conditional instructions or other instructions for specific events within a phase. Each sub-block starts with a ***Base*** instruction, continues with the appropriate instructions and always ends with an ***End_Block*** instruction.

Description of icons and text formats in the text method

The following table describes the icons and text formats that may appear in the ***Text Instructions*** tab:

Icon/text format	Description
Grey square beside text	A block containing instructions that can be run.
Grey square with a red cross 	A block containing one or more instructions that are not possible to run due to instrument configuration incompatibility (syntax errors).
Bold text	Instructions that can be run.
Red dot 	Instructions that are not possible to run. All such instructions must be deleted or changed before a method can be run. See *Section 10.2.3 Working with text instructions, on page 271*. The errors in the instructions may be of the following types: • Instructions that apply to a different instrument configuration (can occur if a method is written for one system and saved for another). • Instructions for deselected components in the ***System Setup***. • References to blocks that are not defined in the method (e.g., a ***Watch*** instruction but no instructions to be executed when the ***Watch*** is activated).
Normal text	Instructions that will not be run. Instructions with a red dot are formatted as normal text instead of bold text. Unused instructions are also formatted as normal text. Instead of deleting instructions they can be moved to unused instructions below the text method.
Text with a red loop symbol 	When a block is called from within itself this will generate a potentially infinite loop. It is not possible to run such a method.

10.2 Working with methods in the Text Instructions tab

Introduction

This section describes how to create or edit methods using specific text instructions. The general structure of the text method syntax is described, including the major hierarchy of the text method parts (phases and blocks).

In this section

10.2.1 Base instruction

Introduction

Every method block must start with a ***Base*** instruction, defining the base for calculating breakpoints (see also *Structure of the text method, on page 257*). Different blocks can use different bases.

This section describes how to choose and edit settings for the ***Base*** instruction.

What base should I use?

Depending on the experiment, different bases should be used. Use the base that most closely suits the purpose of the block. Column volume (***CV***) is recommended as the base for most steps in a run. In some situations, however, it may be more suitable to use a time or volume base for individual blocks.

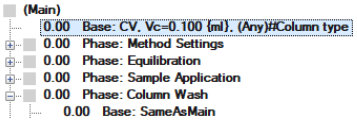
The table follows lists few examples when different bases should be used:

Use...	When...
CV	the method should be adjusted according to the selected column. In this way, you do not need to edit the method when changing column size.
Volume	the same volume should be used regardless of which column is used.
Time	a defined time is required and the volume used is not critical, or if the flow rate is zero.

Edit settings for a base instruction

Follow the instructions to edit settings for a base instruction:

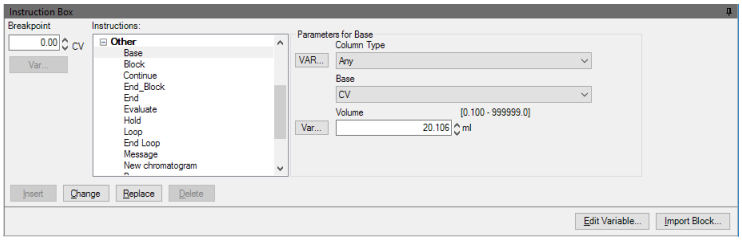
Step	Action
1	Select the base instruction for which to edit the settings in the ***Text Instructions*** area.



Result:

Step	Action

- The settings for the selected ***Base*** instruction are displayed in the ***Parameters for Base*** area in the ***Instruction Box***.



- If a ***Base*** instruction in a phase or block was selected using the same parameter settings as the ***Main*** block, this is displayed in the ***Instruction Box***.

2 **a.** Select the appropriate ***Base*** from the ***Base*** drop-down list:

- ***Volume*** (the unit depends on which ***Instrument Configuration*** used)
- ***Time*** (minutes)
- ***CV***, column volume (the corresponding volume in for example ml can be defined numerically or taken from the ***Column Type*** list)
- ***SameAsMain*** (does not apply for the main block). The block will inherit the base defined in the main block.

Result:

The settings in the ***Parameters for Base*** area are updated.

3 Select the appropriate ***Column Type*** in the drop-down list.

The table below gives a short descriptions of the available options:

Column parameter	Description
Any	Any column can be used in the block. If the ***Column Type*** is set to ***Any*** and the ***Base*** is set to ***CV***, enter the column volume in the ***Volume*** field.
ColumnSameAsMain	The same column as in the main block will be used. When the ***Base*** is set to ***Volume*** but the flow still goes through the column, the ***Column Type*** can be set to ***SameAsMain*** to provide information on, for example, pressure limits for the column.

Step	Action

Column parameter	Description
***Named column type* (e.g., HiTrap Q HP, 1 ml)**	The named Column type will be used in the block. The volume specified in the selected column definition will automatically be used for ***Volume*** parameter in the method block, and thus used to calculate column volumes (***CV***). The ***Volume*** parameter may then not be edited manually. The Column Type parameter can be defined as a variable. This may be useful if it is desirable to change Column type when starting the method run in the ***Variable List*** during the ***Start Protocol*** (see *Set up a Start Protocol, on page 57*). See *Section 10.2.4 Method variables, on page 277* for information about how to define variables.

4 Click ***Change*** or ***Replace*** to save the settings for the selected ***Base*** instruction.

Result:

The parameters for the ***Base*** instruction are updated.

Note:

If the Column type is changed, the ***Column Value Update*** *warning dialog box opens, displaying the changes that will be made in the methods, based on the column default values (see diagram below). If these changes are correct, click* ***OK****, otherwise click* ***Cancel****.*

10.2.2 Working with phases and blocks

Introduction

This section describes how to add, delete and edit phases and blocks in the text method. It also describes how to import blocks from other methods.

Phases vs blocks

Because phases are blocks at the highest level in the text method, the same editing operations can be performed. In this section the name block will be used both for phase blocks and other blocks unless otherwise stated.

Exception

It is not possible to add a phase using the ***Instruction Box***. A new phase must be added from the ***Phase Library***. The ***User Defined*** phase is intended for this purpose, but any phase may be text edited.

See *Section 3.6.1 Edit the method outline, on page 51* for information about how to add phases to the ***Method Outline***.

Method blocks

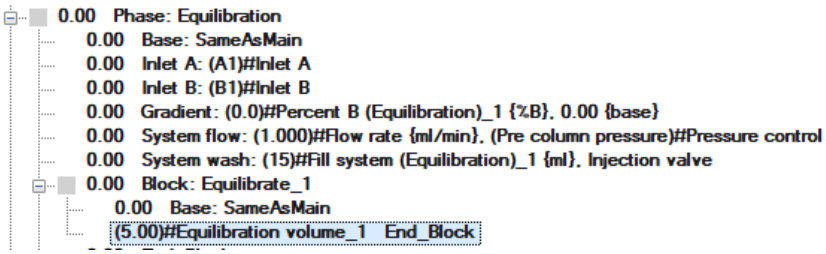
Instructions in each block are executed in the order they are written. The instructions within a block are executed until the block is finished or the ***End_Block*** instruction is executed. Any settings made in a block are valid throughout the method until the settings are changed.

However, if a conditional instruction, e.g., a ***Watch*** instruction controlling the start of a sub-block, is included in a phase the instructions in the sub-block are executed when the condition for that ***Watch*** is met (e.g., when a particular monitor signal meets a given condition).

Block length

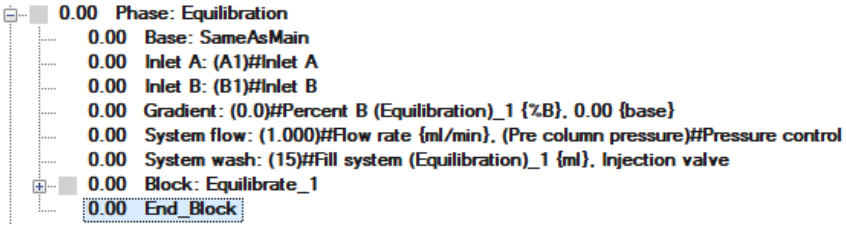
The length of a block is determined by the breakpoint of the last instruction in the block. Even if all breakpoints are set to 0, the instructions might take some time/ volume because they are executed sequentially.

The illustration below shows an example of a method where ***Equilibration*** has a breakpoint set to 5:



In the example above, the value `5.00` will be 5 column volumes (***CV***) if the ***Base*** in the ***Main*** block is set to ***CV***, 5 minutes if ***Base*** is set to ***Time*** or 5 ml if ***Base*** is set to ***Volume***.

To extend the length of a block without performing any other operation, set the break-point of the ***End_block*** instruction appropriately, for example, as in the illustration below:



In this example, the block will end after 0.5 ml, since ***Base*** is set to ***Volume***.

An estimation of the time for running the method can be obtained in the ***Method Duration and Variables*** window. See *View and print the method duration time and variables, on page 61*.

View/hide instructions in blocks

Follow the instructions to view or hide blocks and text instructions in the ***Text Instructions*** tab:

If you want to...	then...
expand all blocks in the method	double-click ***Main***
view the instructions in a block	• click the "+" symbol or • double-click the block name.
hide the instructions in a block	• click the "-" symbol or • double-click the block name.

Add phases

Phases can be added to a text method by:

- adding any phase to the method from the ***Phase library***. The phase ***User Defined*** is designed for use in creating text methods from scratch, and consists only of ***Base*** and ***End_block*** instructions. See *Section 3.6.1 Edit the method outline, on page 51* for information about how to add a phase to the ***Method Outline***.

  or

- by copying/pasting an existing phase in the text method and then edit it. See *Copy and paste blocks, on page 266* for information about how to copy and paste blocks.

Note: *It is not possible to add a new phase using the* ***Instruction Box****.*

Add blocks in a phase

Follow the instructions to add blocks in a phase:

Step	Action
1	Select the instruction or block after which you want to insert the new block.
2	Select ***Other*** →***Block*** in the ***Instruction Box***.
3	**a.** Enter a name for the block in the ***Block*** field. **b.** Click the ***Insert*** button. *Result:* The block is inserted after the block that was selected in step 1.

Copy and paste blocks

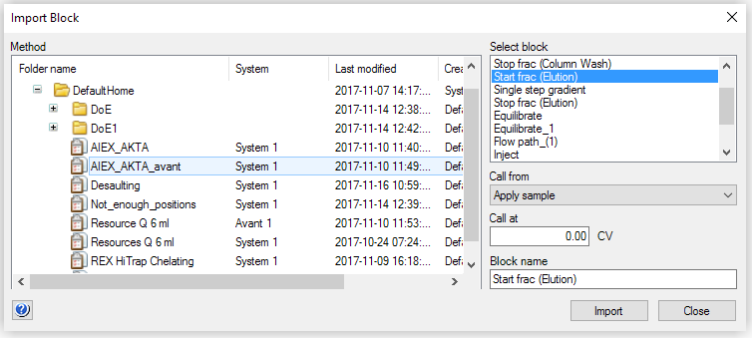
Follow the instructions to copy and paste a block.

Step	Action
1	Select the block you want to copy. • click the ***Copy*** icon in the ***Toolbar***  or • right-click the block and choose ***Copy*** or • select ***Edit*** →***Copy*** (**Ctrl+C**)
2	Select the instruction line just above the point where you want the block to be pasted. • click the ***Paste*** icon in the ***Toolbar*** or • right-click the instruction line and choose ***Paste*** or • select ***Edit*** →***Paste*** (**Ctrl+V**) *Result:* A ***Rename*** dialog box opens.

Step	Action
3	• Click ***Yes*** to rename the block after insertion A new block is created. The variables in the block will get new names so the variable values can be changed without affecting the original block. *or* • Click ***No*** to just insert the copied block with the same name. The block and variables names in the block are copied. If changing variable values in the pasted block, the values will be changed in the original block as well. *Result*The block is inserted in position. ***Note:*** *The pasted block is inserted with the same breakpoint value as the block or instruction selected for point of insertion. When a* ***Phase*** *is copied and pasted the* ***Rename*** *dialog box is not opened.*

Import blocks

Follow the instructions to import blocks from other methods:

Step	Action
1	Click the ***Import Block...*** button. *Result:* The ***Import Block*** dialog box opens.
2	Locate and select the method you wish to import the block from in the ***Method*** folder structure.  *Result:* All available blocks are listed in the ***Select block*** field.

Step	Action
3	Select a block to import from the method in the ***Select block*** list.
4	**a.** Select the block where the imported block will be inserted in the ***Call from*** drop-down list. **b.** Type the breakpoint that the imported block will be called at in the ***Call at*** text box. **c.** If necessary, type a new name for the block in the ***Block name*** text box (optional).
5	**a.** Click the ***Import*** button. **b.** Click ***OK*** to confirm if you also want to import sub-blocks (if any) *Result:* The block is imported into the method you are editing. Unless you have specified a breakpoint that is earlier, the block will be inserted at the end of the block that it is called from. ***Note:*** • *If one or more of the imported blocks has a name that already exist in the method, a dialog box is displayed. Select* ***Use the block(s) already in the method*** *to add a call to the existing block instead of importing the blocks with a name conflict. Select* ***Import the block(s) with a new name*** *to import the blocks but with a new name.* • *If there are variable names in the imported block(s) that already exist in the method, the variable value in the imported block will be changed to the value set in the method before the import. If the variables use different units, the variable will be added with a new name instead.*

Move blocks

Blocks can be moved by drag and drop within the method. You can also use ***Cut*** and ***Paste*** as described:

Step	Action
1	Select the block you want to move. • click the ***Cut*** icon in the ***Toolbar***  or • right-click the block and choose ***Cut*** or

Step	Action
	• select ***Edit*** →***Cut***
2	Select the instruction line just above the point where you want the block to be moved. • click the ***Paste*** icon in the ***Toolbar***  or • right-click the instruction line and choose ***Paste*** or • select ***Edit*** →***Paste*** *Result:* The block is now removed from its original breakpoint and pasted at the new breakpoint. ***Note:*** *The pasted block is inserted with the same breakpoint value as the block or instruction selected for point of insertion.*

Rename blocks

Follow the instructions to rename a block:

Step	Action
1	Right-click the block in the text instruction area and click ***Rename***. *Result:* The block name is highlighted in a box.
2	Type in a new name. ***Note:*** *If the block you renamed is called in a* ***Block*** *or* ***Watch*** *instruction, the block name in these instructions will be changed automatically.*

Delete blocks directly in the text instructions

Follow the instructions to delete a block:

Step	Action
1	• Right-click a block and click ***Delete***. *or* • Select a block and click ***Delete*** in the ***Instruction Box***. *or* • Select a block and press the **Delete** key on the keyboard. *Result*A dialog box will appear asking if the block should be deleted permanently or moved to unused blocks. ***Note:*** *If deleting a phase, the phase will be deleted right away.*
2	**a.** Click ***Delete*** to delete the block permanently. **b.** Click ***Move to <Unused>*** to delete the block from the method and place it after the method.

Delete blocks using the Delete block(s) dialog

Follow the instructions to delete a block:

Step	Action
1	In the ***Instruction box***, click ***Delete block(s)***. *Result:* The ***Delete block(s)*** dialog opens, displaying all blocks in the method in alphabetical order.
2	Select the block(s) you want to delete from the method and click ***OK***. ***Note:*** *If any of the selected blocks contain sub-blocks, a dialog box is displayed asking what you want to do with sub-blocks that become unused. Select* ***Delete*** *to remove the sub-blocks from the method permanently or* ***Move to unused*** *to place them in the unused section of the method.* *Result:* The selected blocks are permanently deleted from the method.

10.2.3 Working with text instructions

Introduction

Instead of editing the method in the ***Phase Properties*** tab, instructions may be edited one at a time in the ***Text Instructions*** tab. The instructions in a block are always executed sequentially. This section describes the general principles for how to edit instructions.

Help texts for the instructions

It is possible to display help texts for the instructions that can be inserted in the ***Instruction Box***.

Follow the instructions to display the help text for an instruction:

Step	Action
1	In the ***Instruction Box***, select the appropriate instruction for which to display help text.
2	Press **F1** on the keyboard. *Result:* A dialog box with help text for the selected instruction will be displayed.

Insert a new instruction

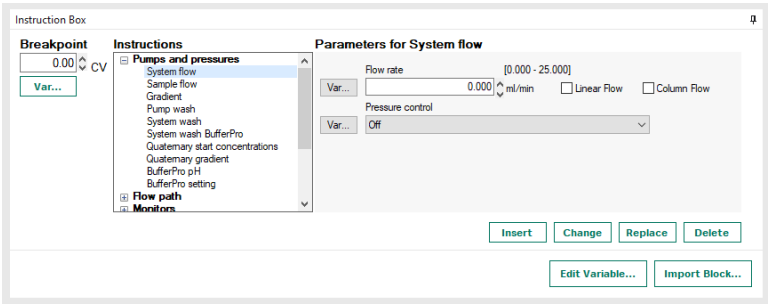
Follow the instructions to insert a new text instruction in the ***Text Instructions*** area:

Step	Action
1	Select a block and display the instructions within the block.
2	Select the instruction in the block after which you want to add the new instruction.
3	Open the ***Instruction Box*** if it is hidden. Do the following: **a.** Set the appropriate breakpoint in the ***Breakpoint*** box. **b.** Choose the instruction type and the instruction in the ***Instructions*** field. For basic help on each instruction, select the instruction and press **F1**.

Step	Action

c. Type values for instruction parameters in the ***Parameters*** text boxes.

The allowed range is shown in brackets beside the text box. If a scroll bar appears at the right side of the ***Parameters*** field, additional parameters are available.



4 Click the ***Insert*** button.

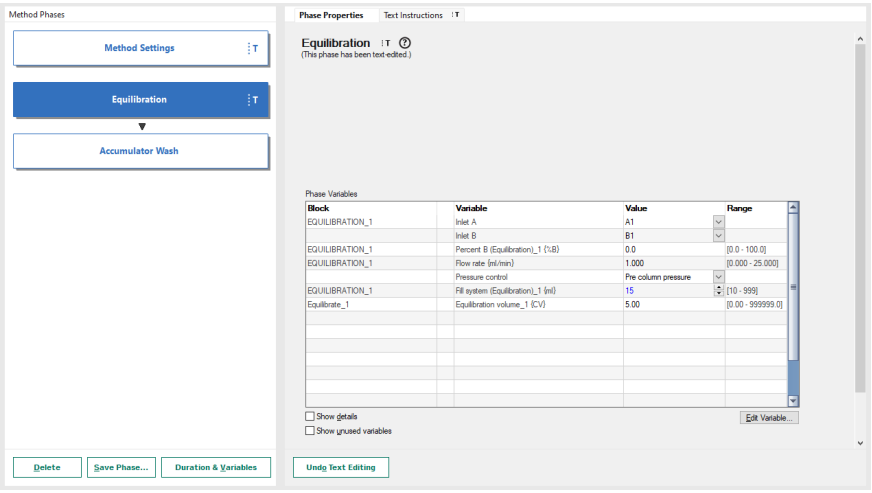
Result:

The instruction will be inserted in the block:

- at the position of the breakpoint of the new instruction, if there are no other instructions at that breakpoint
- immediately after the currently highlighted instruction, if the highlight is at the same breakpoint as the new instruction
- as the last instruction at the breakpoint, if there are several instructions at the same breakpoint and none of these is highlighted.

Note: *Once a phase generated method has been edited in text editing mode, the phases affected by the edited instruction are indicated with the letter **T**, and the **Phase Properties** tab changes to show a variable list, as shown below. For a phase generated method you can click **Restore Phase Properties** to return the method to the state before the text edit. Any changes that were made in the **Text Instructions** tab will be removed.*

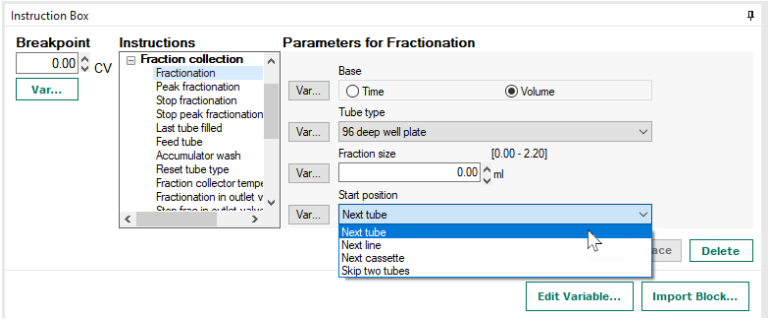
*New phases from the **Phase Library** may be inserted in the method after text editing and the settings for these new phases can be edited in the **Phase Properties** tab or **Text Instructions** tab.*



Change or replace an instruction

Follow the instructions to edit instructions in the ***Text Instructions*** area:

Step	Action
1	Select an instruction in the text method. *Result:* The current ***Breakpoint*** and parameters for the selected instruction is displayed in the ***Instruction Box***.
2	Edit or select parameter values in the ***Instruction Box***:



3	To add the edited or a new instruction to the method, click one of the following buttons: **a.** ***Insert*** **b.** ***Change***

Step	Action
	c. *Replace*
	Note:
	*The **Insert** button adds the edited instruction immediately below the instruction that was selected in the method.*
	*The **Change** and **Replace** buttons are equivalent unless changes are made to the breakpoint or gradient length. Both buttons replace the highlighted instruction with the newly edited instruction. The differences are explained below.*

Effects of the Change button and the Replace button on breakpoints

The following table describes the difference in function between the ***Change*** and ***Replace*** buttons when changing breakpoints:

Button	Function
Change	This button shifts all subsequent instructions in the block according to the change in the breakpoint. ***Change*** does not affect the relative order of instructions in the method. You cannot change the breakpoint of an instruction to earlier than the nearest previous breakpoint in a block.
Replace	This button moves the selected instruction but does not change the breakpoint of any other instruction. ***Replace*** can change the relative order of instructions in the method.

Effects of the Change button and the Replace button on gradient length

The ***Length*** parameter in the ***Gradient*** instruction affects the length of a gradient. The change will have different results depending on which button is used. The following table describes this:

Command	Function
Change	If this button is used to change the length of a gradient, the breakpoints for any instructions issued during the progress of the gradient will be adjusted proportionately so that they are always placed at the same relative position within the gradient. Instructions issued after the end of the gradient will be shifted by the amount of the change. Since the gradient works over time, any instruction that you want to insert after a gradient should be placed after the combined breakpoint and gradient length. ***Note:*** *Moving the **End_block** instruction in a gradient block with the **Change** button does not affect the length of the gradient.*
Replace	If this button is used to change the length of a gradient, other instructions are not affected.

Move instructions

A selected instruction may be dragged-and-dropped in a new location to change the order of instructions. The symbol shown in the illustration below will be displayed if the instruction cannot be dropped in a specific location.

Delete instructions

Follow the instructions to delete method instructions in the ***Text Instructions*** tab:

Step	Action
1	Select the instruction in the ***Text Instructions*** tab.
2	• Right-click the instruction and click ***Delete***. *or* • Click ***Delete*** in the ***Instruction box***. *or* • Press the **Delete** key on your keyboard.

End_Block instruction

If you delete the ***End_Block*** instruction, the block will end at the last instruction in the block. If a gradient is currently being formed, the gradient will continue into the next block.

10.2.4 Method variables

Introduction

Variables are used when you want to vary parameter values in a method. Variables must be defined when you want to:

- perform scouting and ***Design of Experiments*** (***DoE***) where different parameters are varied to find, for example, optimal settings for a process.

  See *Chapter 4 Scouting, on page 86* and *Chapter 5 Design of Experiments, on page 98* for more information.
- change parameter values in the ***Start Protocol*** immediately before the start of a method run without using the ***Method Editor***, allowing one method to be used for runs under a variety of conditions. Each parameter defined as a variable is assigned a default value, which is used if no changes are made to variable values at the start of a run.

Viewing method variables

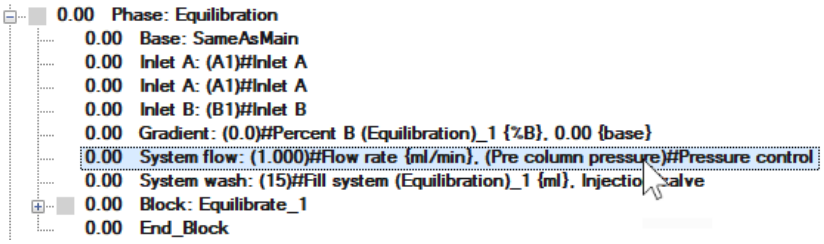
All variables in a method are listed on the ***Variable List*** tab in the ***Method and Duration*** dialog, grouped according to the phase and block in which they appear. For information about how to view the variables in a method, see *View and print the method duration time and variables, on page 61* for more information.

If a method has been text edited or created using the wizard the phase variables for the selected phase will be displayed in the ***Phase Properties*** tab. If the ***Method Settings*** phase has been edited, some additional parameters will also be displayed. It is possible to edit variable names, values and the other settings displayed in the ***Phase Properties*** tab.

Identifying variables in the Text Instructions area

Parameters that are defined as variables in the text method are indicated in the ***Text Instructions*** area.

The parameter is given as the default value in parentheses followed by the variable name. The following illustration shows an example of this:



For example, in ***(1.000)#Flow rate {ml/min}***:

- `(1.000)` is the default value for the variable

- `{ml/min}` is the variable unit
- `Flow rate` is the variable name

Identifying variables in the Instruction Box

Parameters that are defined as variables in the text method are also indicated in the ***Instruction Box*** for the selected instruction in the ***Text Instructions*** area.

When the instruction is shown in the ***Instructions*** field of the ***Instruction box***, the ***VAR*** button beside the parameter field is displayed in capital letters for variables (that is: ***VAR*** not ***Var***).

Variable name conventions

Variables are defined with names that can be explicit descriptions of the variable function, for example ***Sample volume*** and ***Gradient length***. Suitable choices of variable names can make the method easier to read and understand, and also help the operator in setting variable values at the start of a method run.

When defining and/or renaming variables, consider the following:

- The names can be up to 50 characters long and the following characters can be used:
  - Letters (`A–Z`)
  - Digits (`0–9`)
  - The underscore character (_)
  - The `Space` character
- The case of letters is retained, but not significant. The names ***Flow Rate*** and ***FLOW RATE*** are treated as identical.

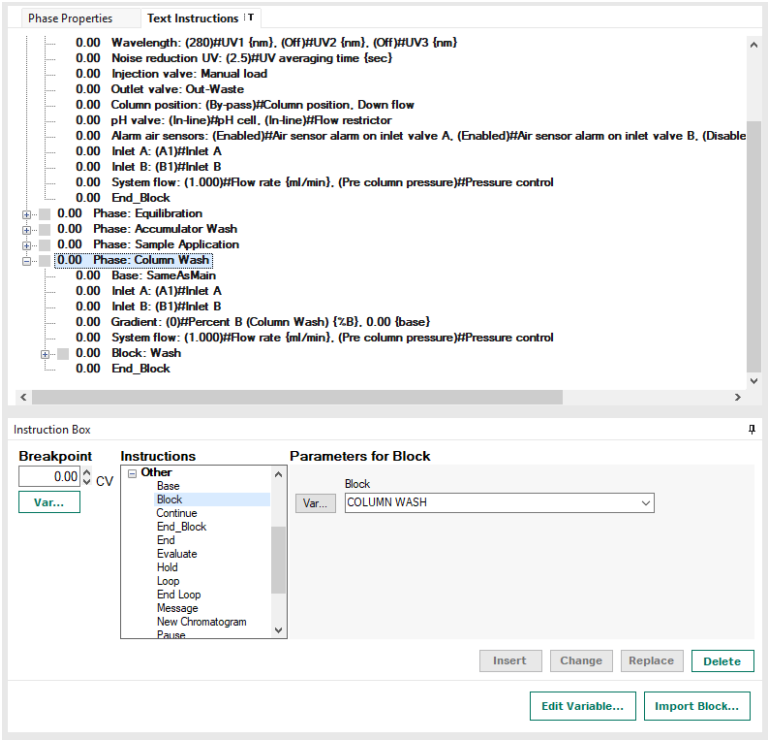
For information about defining and renaming variables, see *Define new variables, on page 278* and *Edit variables, on page 280*.

Define new variables

Only one variable that affects block length (breakpoint or gradient length) may be defined within each block. However, any number of parameters may be defined as variables within a block. Follow the instructions to define a new variable.

Step	Action
1	Select the instruction where you want to define the variable in the ***Text Instructions*** area. *Result:* The parameters for the instruction are shown in the ***Instruction Box***.

Step	Action



2 **a.** Locate the breakpoint or the required parameter in the ***Instruction Box***.

b. Click ***Var***.

Result:

The ***New Variable*** dialog box opens.

3 **a.** Type a name for the variable (see *Variable name conventions, on page 278* for information about how to name variables).

b. Select the ***Visible in details only*** check box if you want to set the variable as a "details" variable. Detail variables become visible in the ***Variable List*** if the ***Show details*** check box is selected. This option can be used to simplify the workflow later.

c. Click ***OK***.

Result:

The ***Var*** button changes to ***VAR*** to confirm the new variable.

Step	Action
	Note: *If a breakpoint or gradient length is defined as a variable, changing the variable value in the **Variable List** tab when the method run is started will shift other instruction breakpoints accordingly. This functionality is equivalent to using the **Change** button to alter a breakpoint or gradient length (see Section 10.2.3 Working with text instructions, on page 271 for how the **Change** button affects instructions within gradients).*
4	Click ***Change***. *Result:* The variable is saved and displayed in the ***Text Instructions*** area.

Edit variables

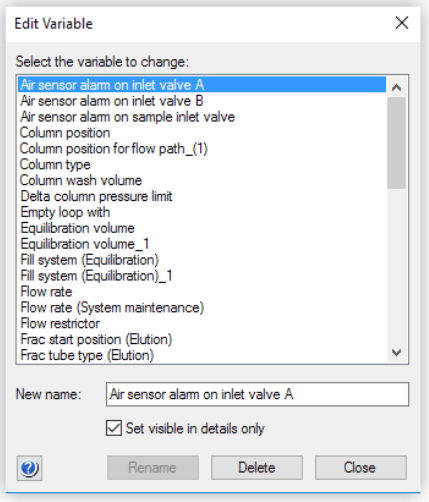
Editing a variable includes renaming and deleting the variable and choosing whether the variable should be a detailed variable or not. For information about how to edit the variable values, see *Edit variable values, on page 282*.

Edit a variable using the Edit variable button

Follow the instructions to edit a variable using the ***Edit Variable*** button:

Step	Action
1	**a.** In the ***Instruction Box***, click ***Edit Variable***. **b.** Alternatively, if the phase containing the variable has been text edited, click the ***Phase Properties*** tab to display the phase variables, select the variable and click ***Edit Variable***. *Result:* The ***Edit Variable*** dialog box opens displaying all variables (if opened from the ***Text Instructions*** tab) or the phase variables (if opened from the ***Phase Properties*** tab).

Step	Action
2	Select the variable to be edited (if not already selected). Do one or several of the following as appropriate:



a. Type in a new name in the ***New name*** field and click ***Rename***.

b. Select the ***Set visible in details only*** check box if the variable should be a detailed variable. Clear the check box to set it to a normal variable.

c. Click ***Delete*** to delete the variable.

Confirm that you want to delete the variable in the message box that appears.

3	Click ***Close*** to close the dialog box.

Edit a variable using the VAR button in the Instruction Box

Follow the instructions to edit a variable using the ***Instruction Box***:

Step	Action
1	Select the instruction containing the variable to be edited in the ***Text Instructions*** area. *Result:* The parameters for the instruction are shown in the ***Instruction box***.
2	Click ***VAR***.

Step	Action
	Result: The ***Edit Variable*** dialog box opens.
3	Do one or several of the following as appropriate: **a.** Type in a new name in the ***Variable name*** box. **b.** Select the ***Visible in details only*** check box if the variable should be a detailed variable. Clear the check box to set it to a normal variable. **c.** Click ***Clear*** to delete the variable.
4	Click ***OK***.
5	To save the changes, click ***Change*** in the ***Instruction Box***. *Result:* The text instruction is updated.

Edit variable values

To edit default variable values, you can either

- edit the value in the ***Phase Properties*** tab if the text method has been edited.

  or

- edit the instruction in the ***Instruction box*** of the ***Text Instructions*** tab.

Changes made in the ***Phase Properties*** tab are automatically updated on the ***Text Instructions*** tab and vice versa.

Edit variable values in the phase variables list

If the phase containing the variable value to be edited has been text edited, it is possible to edit the variable value on the ***Phase Properties*** tab. Follow the instructions edit variable values in the ***Phase Properties*** tab for a text edited phase:

Step	Action
1	Click the ***Phase Properties*** tab to display the ***Phase Variables*** list.
2	Change the variable value for the appropriate variable in the ***Value*** box by selecting a new value in the drop-down list or typing in the box. ***Tip:*** *To show detailed variables, select the* ***Show details*** *check box.* *Result:* The variable value is updated.

Step	Action
3	Repeat this procedure for the appropriate variables.

Edit variable values in the Instruction Box

Follow the instructions to edit variable values in the ***Instruction Box***:

Step	Action
1	Select the instruction containing the variable value to be edited in the ***Text Instructions*** area. *Result:* The parameters for the instruction are shown in the ***Instruction box***.
2	Change the value for the appropriate variable(s) (indicated by ***VAR***).
3	Click ***Change***. *Result:* The settings are saved and the text instruction updated in the ***Text Instructions*** area.

10.3 Specific instructions

Introduction

This section describes some text instruction applications, for example:

- Gradient instructions
- Alarms
- Conditional instructions
- Messages, set marks, pause and hold instructions

In this section

10.3.1 Gradients and eluent concentrations

Introduction

Gradient instructions allow definition of an A- and B-buffer mix. The starting point for the ***Gradient*** is always the current eluent composition. The instruction can be read as follows: "form a ***Gradient*** to reach ***Target*** after ***Length***". Linear gradients and step gradients can be created using ***Gradient*** instructions.

Gradient instructions are given in the ***Text Instructions*** editor of the ***Method Editor***. This type of instruction defines gradients and immediate changes in eluent concentration.

Linear gradients

A gradient can be defined as a linear gradient. The eluent composition changes linearly over time.

Example of instruction

```
10.00 Gradient 50{%B}, 20{base}
```

The example instruction above forms a gradient to 50%B (***Target***) starting at breakpoint 10 with duration 20 method base units (***Length***). The example instruction will finish at breakpoint 30. If the current eluent concentration is greater than 50%, the gradient will be negative.

Step gradients

A gradient can be defined in several steps. A step gradient is an immediate change in eluent composition. To form a step gradient, set the ***Length*** parameter to 0 in the ***Gradient*** instruction.

Example of instruction

```
10.00 Gradient 50{%B}, 0{base}
```

The example instruction above forms a step from the current eluent composition to 50%B at breakpoint 10. The method continues with 50%B.

Insert a Gradient text instruction

The table below describes how to insert a ***Gradient*** instruction:

Step	Action
1	At a suitable ***Breakpoint*** in the method, select the instruction line immediately before where you want to insert the gradient (this decides when the gradient begins).
2	**a.** Expand the ***Pumps and pressures*** item in the ***Instructions*** field of the ***Instruction Box***.

Step	Action
	b. Select ***Gradient***.
	c. In the ***Parameters for Gradient*** field, select appropriate values for:
	• ***Target*** (final eluent composition expressed in % eluent B)
	• ***Length*** (duration of the gradient)
	Tip:
	*To form a step gradient, set the **Length** parameter to zero.*
	Tip:
	*For many purposes, it can be useful to define the length of the gradient as a variable. When this is done, breakpoints for instructions issued during or after the gradient in the same block are automatically shifted in proportion to the length of the gradient when the variable value is changed. This is the same functionality as the **Change** button command in the **Instruction Box**.*
3	Edit the ***Breakpoint*** for the gradient, if appropriate.
	Note:
	*The breakpoint for a **Gradient** instruction defines the time or volume (according to method base) for the start of the gradient. A gradient with a non-zero duration occupies time and volume in the method, and breakpoints for other instructions may be set to occur before the gradient is completed. The instruction is simply carried out at the requested breakpoint, while the gradient is forming.*
4	Click the ***Insert*** button.
	Result:
	The new ***Gradient*** instruction is inserted in the method in the ***Text Instructions*** area.

Instruction after a gradient

Any instruction that you want to insert after a gradient should be placed after the combined breakpoint and gradient length, since gradients function over time.

Instructions that affect gradients

The table below describes the instructions that affect the gradient:

Instruction	Effect
Gradient	A new gradient will start at the requested breakpoint. The remaining duration of any previous gradient is ignored.

Instruction	Effect
Flow	The eluent flow rate will change at the requested breakpoint. If the current base is volume or column volume, the duration of the gradient will be changed. If the method base is time, the volume of the gradient will be changed. ***Note:*** *If the flow is changed, the slope of the gradient will also change.*
End_Block	The gradient formation will continue uninterrupted unless a new ***Gradient*** instruction is issued. For example, this means that a block can be called conditionally during gradient formation without interrupting the gradient.

10.3.2 Alarm instructions

Introduction

This section is a description of how alarms work in UNICORN and of the ***Alarms*** text instructions. It also describes the differences between ***Alarms*** and ***Warnings***.

Alarms and Warnings

The ***Alarms*** parameter settings define the high and low ***Alarm*** limits for process monitor signals. You can define these limits either in the system settings or as part of a method. Settings in the method will override the system settings.

The limits that will generate a ***Warning*** from the system are defined in the instrument configuration files and you cannot edit these settings.

Conditions can also be applied to process monitor signals such that a block of instructions will execute when a particular condition is satisfied (for example, when the absorbance of the eluent exceeds a certain limit). This is done using ***Watch*** instructions which are described in *Section 10.3.3 Watch instructions, on page 290*.

The table below describes the general difference between ***Alarms*** and ***Warnings***.

If the signal exceeds...	then...
Alarm limits	• an alarm sounds ***Note:*** *The alarm can be disconnected in* ***System Settings****.* • an alarm message is displayed • the process is paused (i.e., the method execution is suspended and all pumps are stopped) • the alarm is noted in the ***Run log***. The situation must be acknowledged and corrected before the process can be continued.
Warning limits	• a warning message is displayed • the process continues • the warning is noted in the ***Run log***.

Note: *The* ***Alarms*** *are not active unless the mode is set to* ***Enabled****.*

Alarms in a network

Alarms and warning messages are displayed on all stations with a connection to the concerned system. This is regardless of the activity that is currently performed in UNICORN and regardless of the identity and access rights of the current user.

Alarms and warnings can however only be acknowledged from the station that is connected in control mode.

Insert an Alarm text instruction

The table below describes how to insert an alarm instruction into the method.

Step	Action
1	Select the instruction line immediately before where you want to insert the ***Alarm***, at a suitable ***Breakpoint*** in the method. (This will decide when the alarm conditions begin.)
2	**a.** Select ***Alarms*** in the ***Instructions*** field of the ***Instruction Box***. **b.** Select the desired alarm from the list.
3	Select appropriate values for ***High alarm*** and for ***Low alarm*** in the ***Parameters*** field. ***Note:*** *There are no high and low settings for **Air sensors**, only enabled or disabled.*
4	Click the ***Enabled*** radio button.
5	Click the ***Insert*** button. *Result:* The new ***Alarm*** instruction is inserted in the method.

Available alarms

The alarms available depend on the instrument configuration. Alarms for the following monitor readings may be set:

- System pressure
- Sample pressure
- Delta column pressure
- Pre-column pressure
- UV1
- Conductivity
- pH
- Air sensors

10.3.3 Watch instructions

Introduction

Watch instructions allow the progress of a method run to be determined by events during the method run. For example, start collecting fractions when the first peak elutes.

The ***Instrument Configuration*** files include ***Watch*** instructions for each monitor defined in the system. These instructions are used to monitor method runs, and instruct the system to call a specified block or an instruction when a particular signal meets a given condition. As long as the condition is not met, the block is not activated.

Note: ***Watch*** *instructions available for the instrument configuration are listed in the* ***Instruction box****.*

When is a Watch active?

The breakpoint when the ***Watch*** instruction is issued determines when the watch begins, not when the block is activated.

A watch is active from the point at which it is issued until:

- the ***Watch*** condition is met.
- depending on systems, a new watch is set for the same monitor.
- a ***Watch off*** instruction is issued for the monitor.

  or
- the method ends.

Insert a Watch text instruction

The table below describes how to insert a watch instruction in the text method. Setting up additional Watch parameters is described afterwards, see *Insert a Watch parameters instruction, on page 293*.

Step	Action
1	At a suitable ***Breakpoint*** in the method, select the instruction line immediately before where you want to insert the watch (this decides when the watch begins).
2	**a.** expand ***Watch*** in the ***Instructions*** field. **b.** select the desired ***Watch*** type: • ***Hold until*** Subsequent instructions in the block will execute when the conditions have been met • ***Watch***

Step	Action
	A specified action will be performed when the conditions have been met • ***Watch off*** Cancels the active watches on the specified signal
3	Select a signal for the watch from the Signal drop-down menu. See *Monitor signals to watch, on page 291* for available signals that can be selected.
4	For watch types ***Hold until*** or ***Watch***, select the appropriate ***Parameters for Watch***: **a.** ***Test*** See *Test options in the Parameters field, on page 292* for a description of the different ***Test*** options. **b.** ***Value/Slope/Minutes/Factor*** depending on the selected test **c.** select an appropriate ***Action***. See *Actions when a Watch condition is met, on page 292* for a description of the different ***Watch Action*** options.
5	Click the ***Insert*** button. *Result:* The new ***Watch*** instruction is inserted in the ***Text Instructions*** area. ***Note:*** *A **Watch off** instruction can be added to the method at a breakpoint where the watch no longer is needed.*

Note: *Watch parameters may be set as variables so that the method easily can be adjusted for different run conditions.*

Monitor signals to watch

The monitor signals that can be watched differ depending on the ***Instrument Configuration*** but may include the following:

- pH
- Cond
- UV (1,2 and 3)
- Pressure (System, Sample, Pre-column and Delta-column)
- Flow (System and sample)
- Air sensor (System pump A and B, sample pump)

The buffer concentration may also be set as a watch parameter.

Test options in the Parameters field

The table below describes the ***Test*** options that are available for the ***Watch*** instruction in the ***Parameters for Watch*** field:

Option	Explanation
Greater than	The signal exceeds a certain value.
Less than	The signal falls below a specified value.
Slope greater than	The rate of change of the signal exceeds a specified value, expressed in monitor units/minute (for example, mAU/min).
Slope less than	The rate of change of the signal falls below a specified value, expressed in, for example, mAU/min.
Less than or valley	The signal falls below a specified value or a valley is detected. A valley is detected only after a ***Peak max*** has been detected, and the valley is defined by a local minimum followed by an increase to 102% of the local minimum value plus the ***Delta peak*** value (see *The Delta peak setting, on page 293*).
Peak max	The signal falls to a specified fraction of the most recent peak maximum minus the ***Delta peak*** value.
Stable signal	The signal is stable, within the accepted fluctuation given by the relevant ***Watch parameters*** instruction (see *Insert a Watch parameters instruction, on page 293*), for the period specified by the minutes parameter.
Equals	The signal equals a specified value.

Note: *In order to set a valid slope value, use the* ***Differentiate*** *function in the* ***Evaluation*** *module to measure the slope of the test chromatogram.*

This information only applies to Evaluation Classic.

Actions when a Watch condition is met

The selection in the ***Action*** drop-down list will determine what happens when the condition of a Watch instruction is met. The table below describes the possible actions:

Instruction	Effect
Block name	Calls the named block. ***Note:*** *All available method blocks are listed in alphabetical order in the drop-down list, before the general actions which are listed below.*

Instruction	Effect
Continue	Continues the method if paused or held.
End_block	Ends the current block and return to the point from which the block was called.
Hold	Holds the method, the flow continues. See *Hold instruction, on page 295*.
End_method	Ends the method.
Next_breakpoint	Indicates that the run may execute the next breakpoint.
Pause	Pauses the method, the flow is stopped. See *Pause instruction, on page 295*.

Insert a Watch parameters instruction

Watch parameters instructions are used to define accepted limits and fluctuations for a signal in a ***Watch*** instruction. ***Watch parameters*** instructions should therefore be inserted just before the ***Watch*** instruction on which the limits are required.

Step	Action
1	Select the instruction line immediately before the ***Watch*** instruction to which the parameters will apply.
2	**a.** Expand ***Watch parameters*** in the ***Instructions*** field of the ***Instruction Box***. **b.** Select the desired watch parameters from the list. **c.** Select appropriate values for the ***Accepted fluctuation*** and ***Delta peak*** (for the ***Watch UV parameters*** and ***Watch cond parameters*** instructions) in the ***Parameters*** field. For information about the ***Delta peak*** setting and how to use it, see *The Delta peak setting, on page 293*.
3	Click the ***Insert*** button. *Result:* The new ***Watch parameters*** instruction is inserted in the method in the text area.

The Delta peak setting

The ***Delta peak*** setting in the ***Watch parameters*** helps the software to detect valleys, peaks and peak maxima, and to avoid trigger Watch actions based on noise.

The ***Delta peak*** value should be set

- large enough so that signal noise does not activate the conditions

  and

- small enough so that the condition is activated close to the valley or peak.

As a general guideline, set the value to 2-3 times the noise level and 5-10% of the smallest expected peak height. If you set a too high value you can prevent a new peak from being detected after a local minimum.

Use of the Delta peak setting

The ***Delta peak*** setting in the ***Watch parameters***

- sets the threshold for signal increase after a local minimum that will be interpreted as a valley for the ***Less than or valley*** condition. A valley and a new peak are detected when the signal increases to 102% of the local minimum plus the ***Delta peak*** value.

  Note: *A valley is detected only after a* ***Peak max*** *has been detected.*

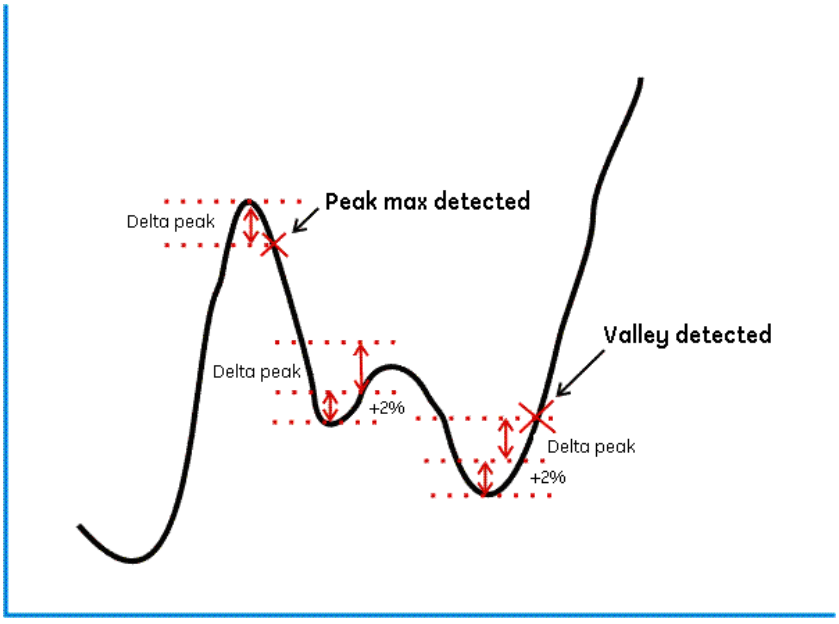
  Example:

  If there is a local minimum at 50 mAU and a ***Delta peak*** of 10 mAU, a valley will be detected at:

  `(1.02 x 50) + 10 = 61 mAU`

- sets the threshold for signal decrease after a local maximum that will activate the ***Peak max*** condition. ***Peak max*** is detected when the signal falls to the specified fraction of the most recent peak maximum minus the ***Delta peak*** value.

  The schematic figure below illustrates the ***Delta peak*** setting where ***Peak max*** is detected when the signal falls by ***Delta peak*** from a local maximum if the ***Peak max Factor*** is set to ***1*** in ***Watch*** →***Watch*** →***Parameters for Watch***:



10.3.4 Pause or hold a method

Introduction

A method can be programmed to be delayed at critical points. There are three instructions for this purpose: ***Pause***, ***Hold*** and ***Hold until***. These instructions are described below.

Pause instruction

The ***Pause*** instruction suspends execution of the method and stops the pumps so that the system comes to a standstill. The valves remain in the position they were in before the pause.

The pause may be defined as ***Infinite*** or for a specified number of minutes.

Resume the method

It is possible to define the pause time for the method in the ***Pause*** instruction. The method will continue when the set time has elapsed.

The method may also be resumed if you click the ***Continue*** icon on the ***System Control*** toolbar:

Note: *If the pause is set to Infinite, the method must be resumed manually by clicking the* ***Continue*** *icon.*

Hold instruction

The ***Hold*** instruction suspends the execution of the method, but continues to pump eluent at the current flow rate and concentration settings. For example, this instruction is useful for giving the operator time to load a sample loop.

Resume the method

The method may be resumed if you click the ***Continue*** icon on the ***System Control*** toolbar:

Note: *With the* ***Hold*** *instruction, the method must always be resumed manually by clicking the* ***Continue*** *icon.*

Hold until instruction

The ***Hold until*** instruction is a special kind of ***Watch*** instruction. The method is put on hold until a specific condition is met (***Signal***, ***Test*** or ***Value***) or the ***Timeout*** is reached. Thereafter the remaining instructions in the method are executed. See *Section 10.3.3 Watch instructions, on page 290* for a description of ***Watch*** instructions.

Insert a Pause, Hold or Hold until instruction

The table below describes how to insert a ***Pause***, ***Hold*** or ***Hold until*** instruction:

Step	Action
1	At a suitable ***Breakpoint*** in the method, select the instruction line immediately before where you want to insert the ***Pause***, ***Hold*** or ***Hold until*** instruction (this decides when the instruction begins).
2	To insert a ***Hold*** instruction, select ***Other*** →***Hold*** in the ***Instructions*** field of the ***Instruction Box***.
3	To insert a ***Pause*** instruction, select ***Other*** →***Pause*** and enter the ***Time*** for the method to be paused in the ***Time*** field. To pause the method for infinite time, check the ***Infinite*** box.
4	To insert a ***Hold until*** instruction: **a.** select ***Watch*** →***Hold until*** in the ***Instructions*** field of the ***Instruction Box*** **b.** select the appropriate parameters for the ***Hold until*** instruction in the ***Parameters for Hold until*** area. See *Section 10.3.3 Watch instructions, on page 290* for descriptions of the available settings.
5	Click the ***Insert*** button. *Result:* The new instruction is inserted in the ***Text Instructions*** area. ***Note:*** *Instructions that share the same breakpoint as the* ***Hold until*** *instruction, but are placed after it in the method, will be executed after the* ***Hold until*** *conditions have been met.*

10.3.5 Messages and Set marks

When to use a message

Messages are used to inform the operator of the progress of the run or to prompt the user for an action. It is a good idea to issue messages at critical points in the method, for example, in combination with a ***Pause*** instruction to inform the operator that the inlet tube needs to be moved to another inlet.

Insert a Message instruction

The ***Message*** instruction can be used to set up a message that will be displayed for the user during the execution of the method run. The message can be for information in a screen only, or it can require a signature before the user can control the system. The messages are all added to the logbook text.

The table below describes how to add a ***Message*** instruction to the method.

Step	Action
1	**a.** Select ***Other*** in the ***Instructions*** field of the ***Instructions box***. **b.** Select ***Message*** in the instructions list.
2	Type a message in the ***Message*** text box in the ***Parameters*** field.
3	Select one of the display options on the ***Mode*** menu: **a.** ***Screen***, that is, only a text message is displayed. **b.** ***Noscreen***, that is, the message will not be displayed but only inserted into the logbook. **c.** ***Authorize***, that is, the message will require a signature from the user before the user can interact with the system again.
4	**a.** Select a sound on the ***Sound*** menu if desired. **b.** Click the ***Insert*** button.

Note: *If the **Message** instruction is inserted in a conditional block it will only be displayed if the conditions of the block (for example a **Watch**) is fulfilled.*

When to use a Set mark

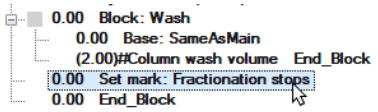
Set mark instructions are useful text messages. They can be used

- to highlight certain stages in a method
- to insert manual notes, for example, when a specific event occurs in a run (only in ***System Control***)

Set marks differ from ***Messages*** in that they are inserted into the chromatogram at set points as well as into the logbook during a method run.

Example of a Set mark

The illustration below shows an example where ***Set marks*** are used to highlight the start and end of fractionation in a method:



Insert a Set mark

Set marks are inserted from the ***Instructions box***. The table below describes how to do this:

Step	Action
1	Select ***Other*** →***Set mark*** in the ***Instructions*** field.
2	Type the message in the ***Mark text*** field.
3	Click the ***Insert*** button. *Result:* A new line with the ***Set mark*** is added to the text method.

11 Troubleshooting

Introduction

This chapter describes different problems which may arise when creating methods in UNICORN, and how to solve the problems.

The Phase Properties tab only shows a Phase Variables table

The table below describes how to restore the options and settings to the ***Phase Properties*** tab:

Problem description	Solution
The ***Phase Properties*** tab shows only a list of variables and not the regular options and settings for the selected phase. The phase is marked with the letter "T" in the method outline. ***Note:*** *User defined phases only have a list of variables.*	The phase has been edited in the ***Text Instructions*** tab. Click the ***Undo text editing*** button to return to the default settings and restore the ***Phase Properties*** options and settings. Note that if the text edited settings also involve subsequent phases and the general ***Method Settings***, all these phases are changed as well and you must restore them all individually.

Options are not available in the phase properties

The table below describes what to do if the standard settings available in the ***Phase Properties*** for a predefined phase are not suitable for your specific application needs:

Problem description	Solution
Options that you need are not available for selection or editing in the ***Phase Properties***.	• Add a ***User Defined*** phase to the method and edit the properties in the ***Text Instructions*** tab or • Text edit the phase where the option is required.

There are red instructions in a method

Red instructions (instructions with a red dot) in a method are syntax errors and may have several causes. A phase containing syntax errors is marked in the method outline with a red cross. The table below describes some solutions to syntax error problems:

Problem description	Solution
The method instructions do not correspond to the components you have chosen for your system.	Check your system components under ***System Properties*** in the ***Administration*** module and that the correct instrument configuration is selected.
Syntax errors are not corrected by changing the component configuration.	Close and reopen the method.
Syntax errors appear because the method was connected to the wrong system. That is, the instrument configuration of the system is incompatible with the method.	• Edit the method so it can be run on the currently chosen system. • Save the method for a system that has all components installed. ***Note:*** *The red instructions must be replaced or removed.* • Reselect the required component under ***System Properties*** in the ***Administration*** module (if the component is actually present on the system). Reopen the method and replace the red instructions with the corresponding instruction for the added component.
Syntax errors appear because the system's instrument configuration has been updated with a new instrument configuration that differs in the instruction set.	Select the red instruction and either delete it or replace it with a corresponding instruction (if available) from the ***Instruction box***. Repeat this for all red instructions before saving the method.
Syntax errors appear because the method was converted for use with a system with a component set up differ from the component set up of the system for which the method was originlly created.	Select the red instruction and either delete it or replace it with a corresponding instruction (if available) from the ***Instruction box***. Repeat this for all red instructions before saving the method.

Problem description	Solution
A phase is marked as incorrect (with a red cross). This may appear if • the instrument configuration has been changed • components have been removed or • the method was converted from a system with a different component set up	Replace the phase with a compatible predefined phase from the ***Phase Library*** . This phase will automatically be adapted to the current instrument configuration and component settings. ***Note:*** *If predefined phases are not available for the system, the red instructions in the phase have to be deleted or replaced.*

Breakpoints are not correctly calculated

The table below describes how to solve problems with calculation of breakpoints in the method, for example in the ***Method Duration and Variables*** dialog box.

Problem description	Solution
Method breakpoints are not calculated. All values are shown as zero.	If the method block uses volume or column volume base, the breakpoints are calculated from the pump flow rate. Check that the flow rate is not zero.

A converted method generates unexpected results

The table below describes how to solve problems when a converted method generates unexpected results.

Problem description	Solution
When running the method, volumes are generally smaller or larger than expected	• Ensure that the method uses Column Volume (CV) as base unit • Verify that all parameter settings that need manual adjustments after the conversion are updated and • Review all text edited phases to locate system parameters that must be edited. For more information, see *Section 3.6.6 Scale or convert methods, on page 70*.

A column cannot be selected when converting and scaling a method

The table below describes how to solve problems when a column cannot be selected for conversion and scaling of a method.

Problem description	Solution
When converting the method including scaling of the column, the field for column scaling is inactive	The reason for this may be that either the option ***Scale*** was not selected, or that the ***Any*** column was selected in the original method. If ***Scale*** was selected, either • select a column in the original method and repeat the conversion or • convert the method to the new system first and select a column in the converted method afterwards.

Print screen does not send a copy of the screen to the printer

The table below describes how to solve a printing problem:

Problem description	Solution
The **Print Screen** command only makes a copy of the screen to the clipboard and not to the default printer.	If you want to print the view on the screen, press the **Print Screen** key and paste the image from the clipboard into an appropriate program, such as Microsoft Paint, and then print out the image.

Inappropriate inlet settings for CIP or preparation

The table below describes how to ensure that the inlet settings are correct for a predefined CIP or preparation phase:

Problem description	Solution
When a CIP or preparation phase (CIP column, CIP system, Prepare column or Prepare system) is started, inlets that are not defined in the phase are briefly used. ***Note:*** *This happens for a very short time and it will normally not cause any problems.*	Check which inlets are chosen in ***Method Settings***. Choose the same inlets as required for the CIP or preparation phase.

Export of a method to a network drive fails

The table below describes how to solve a method export problem:

Problem description	Solution
Export of a method to a network drive fails.	Ensure that the destination network drive is mapped and that you have the appropriate access rights.

Predefined method cannot be created after new Instrument Configuration installation

The table below describes how to solve a predefined method creation problem:

Problem description	Solution
A new instrument configuration is installed. After this, it is still impossible to create a new, predefined method.	The Method Editor must be restarted after importing the new instrument configuration.

Incorrect Bar code reader usage closes Column Handling dialog box

The table below describes how to solve a bar code reader usage problem:

Problem description	Solution
Using the bar code reader to read column bar codes without first selecting a proper input field will cause the Column Handling dialog box to close.	Use the bar code reader only as described in the user documentation.

Page intentionally left blank

Give feedback on this document

Visit cytiva.com/techdocfeedback or scan the QR code.

cytiva.com/unicorn

For local office contact information, visit cytiva.com/contact

29732303 AA V:1 02/2023