



# BatchXpert

English V1.0

*Excel Reporting*

# Content

1	General .....	2
2	Prequisites.....	2
3	General Working Principle .....	3
4	Creating an new Report .....	3
4.1	“BatchXpert Report With Menu.xlsm” .....	3
4.2	“BatchXpert Report Without Menu.xlsm” .....	4
5	Saving/Exporting Batch Data.....	5
6	Function Reference .....	6
6.1	Batch Functions .....	6
6.2	Unit Functions.....	7
6.3	Parameter Functions.....	7
6.4	Total Time functions .....	8
6.5	Event Functions.....	8

# 1 General

The BatchXpert System provides extensive Reporting Functionalities that also includes reporting functionality directly incorporated into Microsoft Excel sheets. These Excel Templates incorporate special Macro functions that allow data retrieval from the BatchXpert historic databases.

BatchXpert automatically installs two default templates on each BatchXpert Station that can be used to create customized Reports from. The default Templates are stored in the BatchXpert installation directory under "Reports" and are called "BatchXpert Report Without Menu.xlsm" and "BatchXpert Report With Menu.xlsm".

## 2 Prerequisites

For the Excel Reporting functions to work properly, it requires a proper installation of one of the following Excel Version:

- Microsoft Excel 2010 (Version 14)
- Microsoft Excel 2013 (Version 15)

Macro Execution must be enabled for the Excel Worksheets to work properly. This may involve adjusting the "Trust Level" in the Trust Center of Microsoft Excel, which can be accessed via the Excel Main Menu "File" -> "Options" -> "Trust Center".

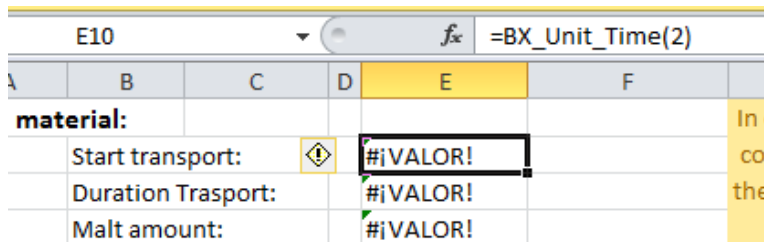
The Reports also require a BatchXpert installation on the PC that is intending to execute the Excel Reports. It is NOT necessary that the PC is also a full BatchXpert operating Station, but since the report is using some BatchXpert internal Libraries that must be installed on the PC it also requires a default BatchXpert installation (without GraphPic and without SQL).

Since the Reports retrieve their data from the Historical Databases of a BatchXpert station it also needs physical and logical access to the SQL-Server Database where the data is stored. The "Connection String" used for accessing the Database can be adjusted in the "Settings" worksheet of the reports.

### 3 General Working Principle

These Report are working on the principle that the user selects an reference Batch and can then use Excel User defined function in cells and formulas to retrieve data from the historical data stored related to the selected reference Batch. The Method of selecting an reference Batch depends on the selected Report tamplate, that is used for creating the report. For more information please refer to the Chapter [“Creating an new Report”](#).

After an Reference Batch is selected, on can use User defined functions to retrieve historical data from the selected reference batch. This can be done by inputting formulas into Excel cells that will be resolved with the Batches data.



The available functions and their usage can be read in the chapter [“Script function Reference”](#) in this manual. The result of these functions an also be used as input to other formulas.

### 4 Creating an new Report

To create an new Report copy one of the Report templates from the BatchXpert Installation directory on any BatchXpert Station and rename it to an name of your choosing.

The default installation directory of the Report templates is:

C:\Program Files (x86)\BatchXpert\Reports

By default there exist two types of templates of Excel Reports. The diferece of these two Templates is the method that must be used for selecting an reference Batch.

#### 4.1 “BatchXpert Report With Menu.xlsm”

This template is intended to be used as primary template for Report creation. It includes a customizable toolbar that can be adjusted to the needs of the report.



This Menu bar integrates into the Excel interface and allows for easy selection of “Batch Area”, “Program or Recipe” and available Batches. It also allows for setting of the Time window for selectable batches in the drop down selections.

The report also includes various Settings for adjusting this “BatchXpert Report” menu to the requirements of the specific Report that is being created. These Settings can be adjusted in the “Settings” worksheet of each Report, and also a description in the form of comments is provided for each Setting.

## 4.2 “BatchXpert Report Without Menu.xlsm”

This template is intended to be used in situation where the main Ribbon Menu cannot be used for selecting a Reference Batch. This Template does not include the “BatchXpert” tab, but provides a selection interface via prepared cells in the “Settings” worksheet of the Template.

User Selection	
Prid	
Batch Number	
Batch Area No	1
From Date	07-04-2015
To Date	07-08-2015

These Fields represent two Methods for selecting an Batch, where all data will be loaded from  
You can select an Prid, to directly select an batch  
Or select an Batch Number by using an Batch Area Number and an Timeframe

In the provided Fields it is possible to select a Reference Batch by inputting a “Process Identifier” (Prid), or by adjusting a Batch Area No, and Time window, and inputting a Batch Number.

These fields are intended to be linked to designated input fields or other means of selection in the main “Report” worksheet. Usually these fields are referenced by “Excel Cell References” to input fields, where the operator can input the appropriate Batch Number, or Prid if available.

## 5 Saving/Exporting Batch Data

Sine the Reports content is dynamically generated from live historical data, the Template does not store the content, but only the reports Layout. This means that if you save the excel Workbook, only the layout and Settings are saved, BUT NOT the reports content (the result of the used functions in the Cells). These values will be regenerated every time the Report is refreshed.

If you want to transfer the actual reports content, or want to archiv the reports content, please use the included “Exporting/Save” Functionality. This Function can be accessed from the BatchXpert Ribbon Menu



Or by directly executing the provided “SaveReport” Macro. This Macro lets you also choose an filename before saving the actual content (NOT ONLY THE LAYOUT) to an independet, Macro free Excel file.

**The resulting Excel File does not contain any Macro or script logic and is completely independent, and can thus transported on any PC (with or without BatchXpert) and viewed with any version of Excel.**

## 6 Function Reference

All provided Reporting Functions are following the BatchXpert Reporting patterns, and to have similar parameters and functions as the equivalent functions that can be used in the “Batch Summary Reports” for the “Batch Report Viewer”. A Complete reference for these functions can be found in the Manual “BatchXpert Batch Summary Reference”.

Since Excel does not allow Function overloading, all BatchXpert related functions are starting with “BX\_”. Since Excel does not support Object Properties in its User defined functions, the Properties of the Units, Events or Parameters are appended with their name at the end of the functions name.

The Functions names are following form:

`BX_{function name}_{Property Name}({parameters})`

The available functions are:

### 6.1 Batch Functions

These functions return properties related to the selected reference Batch. Please refer to the “Batch Class” in the “BatchXpert Batch Summary Reference” Manual.

Parameters:

These functions do not have parameters.

BX\_Batch() As String  
BX\_Batch\_Prid()  
BX\_Batch\_AreaNo()  
BX\_Batch\_AreaName() As String  
BX\_Batch\_BatchNo()  
BX\_Batch\_ProgNo()  
BX\_Batch\_ProgName() As String  
BX\_Batch\_Time() As Date

## 6.2 Unit Functions

These functions return properties related to the selected reference Batch. Please refer to the “Unit Class” in the “BatchXpert Batch Summary Reference” Manual

Parameters:

- ClassNo (Integer): The Class Numbers of the requested Unit/Class
- StepNo (Integer): the Step Number in the Recipe of the unit to Check

BX\_Unit( ClassNo ) As String

BX\_Unit\_ID( ClassNo )

BX\_Unit\_Name( ClassNo ) As String

BX\_Unit\_ClassNo( ClassNo )

BX\_Unit\_Time( ClassNo ) As Date

BX\_Unit\_Info( ClassNo ) As String

BX\_Unit\_InfoID( ClassNo )

BX\_HasManualOpUnit( ClassNo ) As Boolean

BX\_HasManualOpStep( ClassNo , StepNo ) As Boolean

## 6.3 Parameter Functions

These functions return properties related to the selected reference Batch. Please refer to the “Parameter Class” in the “BatchXpert Batch Summary Reference” Manual

Parameters:

- ClassNo (Integer): The Class Numbers of the requested Unit/Class
- StepNo (Integer): the step number that contains the parameter
- ParameterNo (Integer): The Parameter number to load

BX\_Parameter( ClassNo , StepNo , PhaseNo , ParmeterNo ) As String

BX\_Parameter\_SP( ClassNo , StepNo , PhaseNo , ParmeterNo ) As Single

BX\_Parameter\_Val( ClassNo , StepNo , PhaseNo , ParmeterNo ) As Single

BX\_Parameter\_Name( ClassNo , StepNo , PhaseNo , ParmeterNo ) As String

BX\_Parameter\_UoM( ClassNo , StepNo , PhaseNo , ParmeterNo ) As String

BX\_Parameter\_Type( ClassNo , StepNo , PhaseNo , ParmeterNo ) As String



## 6.4 Total Time functions

Parameters:

- ClassNo (Integer): The Class Numbers of the requested Unit/Class
- StepNo (Integer): the step number that contains the requested step
- PhaseNo (Integer): The Phase number of the requested step

BX\_TotalTimeOfUnit( ClassNo ) As String

BX\_TotalTimeOfStep( ClassNo , StepNo , PhaseNo ) As String

BX\_TotalTimeOfSteps( ClassNo , StepFrom , StepTo ) As String

## 6.5 Event Functions

These functions return properties related to the selected reference Batch. Please refer to the “Unit Event Class” in the “BatchXpert Batch Summary Reference” Manual

Parameters:

- ClassNo (Integer): The Class Numbers of the requested Unit/Class
- StepNo (Integer): the step number that contains the requested step
- PhaseNo (Integer): The Phase number of the requested step

BX\_FirstStepEvent\_Time( ClassNo , StepNo , PhaseNo ) As String

BX\_FirstStepEvent\_InfoID( ClassNo , StepNo , PhaseNo ) As String

BX\_FirstStepEvent\_Info( ClassNo , StepNo , PhaseNo ) As String

BX\_FirstStepEvent\_PhaseName( ClassNo , StepNo , PhaseNo ) As String

BX\_LastStepEvent\_Time( ClassNo , StepNo , PhaseNo ) As String

BX\_LastStepEvent\_InfoID( ClassNo , StepNo , PhaseNo ) As String

BX\_LastStepEvent\_Info( ClassNo , StepNo , PhaseNo ) As String

BX\_LastStepEvent\_PhaseName( ClassNo , StepNo , PhaseNo ) As String

BX\_FirstPhaseEvent\_Time( ClassNo , PhaseNo ) As String

BX\_FirstPhaseEvent\_InfoID( ClassNo , PhaseNo ) As String

BX\_FirstPhaseEvent\_Info( ClassNo , PhaseNo ) As String

BX\_FirstPhaseEvent\_PhaseName( ClassNo , PhaseNo ) As String

BX\_LastPhaseEvent\_Time( ClassNo , PhaseNo ) As String

BX\_LastPhaseEvent\_InfoID( ClassNo , PhaseNo ) As String

BX\_LastPhaseEvent\_Info( ClassNo , PhaseNo ) As String

BX\_LastPhaseEvent\_PhaseName( ClassNo , PhaseNo ) As String