MLogics

Automation for Industries

BatchXpert

RTF Reports and Printing Templates

English V1.2

Content

1 General

This Manual describes how the “Report Templates” for Batch Reports, Batch Summary Reports and Manual Operation Reports are edited. It describes how to create, modify and execute these Reports and how to extend them with customized Information. This guide is intended for advanced users and system integrators to the BatchXpert system.

Most of the Reporting and Analytical applications are based on customizable Report Templates that can be adjusted and customized in many ways to fit the projects and applications needs. These Reports are based on the “Rich Text Format” or “RTF” in short, which is a portable text file format that is supported by almost every text editing tool on the market.

This allows the Reporting functions to be easily extensible and adjustable by the end user of the System.

2 Requirements

The Report Templates are based on the “Rich Text Format” or “RTF” in short, which is a portable text file format that is supported by almost every text editing tool on the market. This means that they can be edited by almost every advanced Text editing tool available today such as

- WordPad", which is built into every windows version since Windows XP up to Windows 10
- Microsoft Word
- LibreOffice Write
- OpenOffice Write

Basic Reporting Templates are automatically installed on the System as part of the BatchXpert installation Process, and can be used as starting point for further customization.

3 Basics

Basically the Report templates are used to define different Layouts that control how the Reports will be printed and displayed. The templates can be defined independently for each available Report in the System

A distinction is made among two categories

1. RTF print templates
2. RTF reports

The print templates are used to print out the "step protocols" and the "manual intervention". These Reports generally consist of various Template files, which are then merged into the final document during printing.

Reports generally consist of just one single Template page that will be filled with information depending on the layout and functions defined in the template.

3.1 Basic principles

The RTF templates can be basically regarded as a normal text documents in which certain "placeholder" or "tokens" can be added, with which the Batch Reporting system is instructed to fill these tokens with the actual data of a batch. Graphics and other elements can be inserted at will into the documents. The colors and formatting of texts can be customized at will.

When you insert "wildcards" (token) in the templates, however, some rules must be followed. If these rules are not followed, the tokens are often not recognized as such, and ignored by the system. This manifests itself in the tokens are not replaced by the system of values or an error message, special simply remain so white they were inserted in the template.

1. A token is always encapsulated between "<%" and "%>". The content between these two characters is evaluated by the system.
2. String literals must always put within """(double quotation marks

TIP: if necessary, to reduce the text of the "token", one can apply the following trick. Since the templates and reports always the formatting of the first character of a token, usually always "<%", applied to the entire result of a token, you can reduce the font size of the following without any problems.

3.2 Simple Tokens

In principal, Tokens are dynamically evaluated formulas that are resolved during Report creation, which relate to certain objects. All tokens must always be encapsulated between "<%" and "%>". If a token is evaluated, then also all the mathematical rules (point before line) and brace are considered and evaluated in the correct order. Additionally, certain functions can be invoked which provider further processing of the formulas.

Here are a few examples of the token and its expected result.
<%2 + 2 * 2%> = 6 It will comply with all rules of arithmetic
<%(2 + 2) * 2%> = 8 There are also the staple rules observed
<%"Text"%> = Text All strings must put quotation marks
<%Pow (2,4)%> = 16 Pow is a function that calculates the exponent of two numbers
<%Pow (1 + 2.4)%> = 16 The arithmetic operations are resolved before passing it to the Pow Function

The type of Functions those are available in a Report depends fundamentally on the Report Templates Type and its intended use. All reports have different functions. However, there are some basic features which are available for all reports.

In addition, there are functions, that can be applied to all objects in an Reports object model, such as the "toString" function which converts an object to its string representation.

- <%Now.toString%> = the current date as a string. For example, "03/04/2012 14:03:01"
- <%Now.toString ("d")%> = Current date as pure date, untimed "03.04.2012
- <%Now.toString ("yyyy-mm-dd")%> = Own time format "2012-03-04"

The default formats for number representations can be viewed here:
http://msdn.microsoft.com/en-us/library/dwhawy9k.aspx

For date and time functions, the format display can be viewed here:
http://msdn.microsoft.com/en-us/library/az4se3k1.aspx
http://msdn.microsoft.com/en-us/library/8kb3ddd4.aspx

3.3 Conditionals and Repetition

The system also supports conditionals “if” and repetition “foreach” tokens. Using conditional tokens, it is possible to make entire report passages visible or indivisible depending on any legal Boolean condition.

Using the “foreach” token, one can define a text template that will be applied to every element in a given collection, list or array. “foreach” and “if” can be nested inside each other to form complex reports.

3.3.1 Conditionals

The “if” token is expressed as:
<%if condition%> and closed with <%endif%>.

Every if block must have a matching endif block, to mark the text passage that will be effected by the conditional. The Condition can be any form or mathematical expression that evaluates to an Boolean value such as:
<%if BatchNo >10%> this text is visible only for Batch Numbers greater that 10<%endif%>
<%if HasManualIntervention%> Only visible if there are Manual interventions in the batch<%endif%>

3.3.2 Repetition

The “foreach” token are express as:
<%foreach Batch.Units as Unit%>
this will be repeated for each Unit in Units
<%enforeach%>

Every if block must have a matching enforeach block, to mark the text passage that will be effected by the repetition. The repetition can be applied to any collection, list or array.
<%foreach Batch.Units as Unit%>
The unit name is <%Unit.Name%>:
<%enforeach%>

This will result in
The unit name is Mashtun:
The unit name is Lautertun:
The unit name is Wortkettle:
The unit name is Wortcooler:

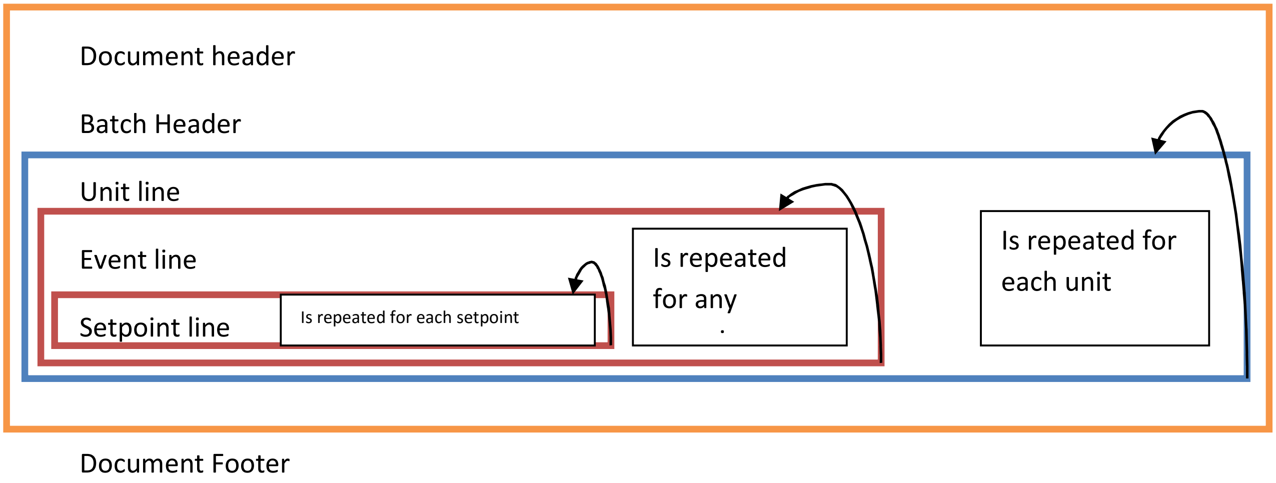
4 The Batch Report print templates

These templates are BatchXpert Batch Report Viewer used to "step protocols" print. For this purpose the various documents are dynamically filled with data and then to an entire document, in its dependencies of the underlying batch together.

The Batch Report print templates have additionally access to the functions relating to the selected batch to the basic functions.

4.1 The structure of the Batch Reports

In principle, the printing originals are divided into 6 sub-ranges, which is then assembled dynamically from BatchXpert an entire document. The Total Document divided into the following structure in which the respective areas are replaced with the appropriate data.



4.2 Document header

This template is added each time you print the first thing and only once in the document. This template has access only to the general functions and has no access to batch-specific data.

This template is meant to be here, the registered company logos and company addresses. Also other things like dates of expression, etc. can be specified here.

4.3 Batch Header

This template is added directly to the document header, and has access to the general batch data. This serves to add batch-specific data to the expression. Thus, here, for example, the batch number, program name, and start time of the batch shall be added. These objectives have access to the data of the selected batch.

4.4 Unit header

This template will be inserted once for each unit, which was active in the batch. This has access to the "Unit" specific data, and is added again for each unit, with whose data then refer to the corresponding browsing attached Unit.

It serves the data of the corresponding plant section output, followed by the subsequent events are related.

4.5 Event Line

This template is added for any occasion a unit. This template has access to the "event" specific data of the active event. It serves the actual events in the document involve.

4.6 Setpoint line

This template is added for each nominal value of an event. This template has access to the "setpoint" specific data. And serves to handle the presentation of the setpoint data.

4.7 Document Footer

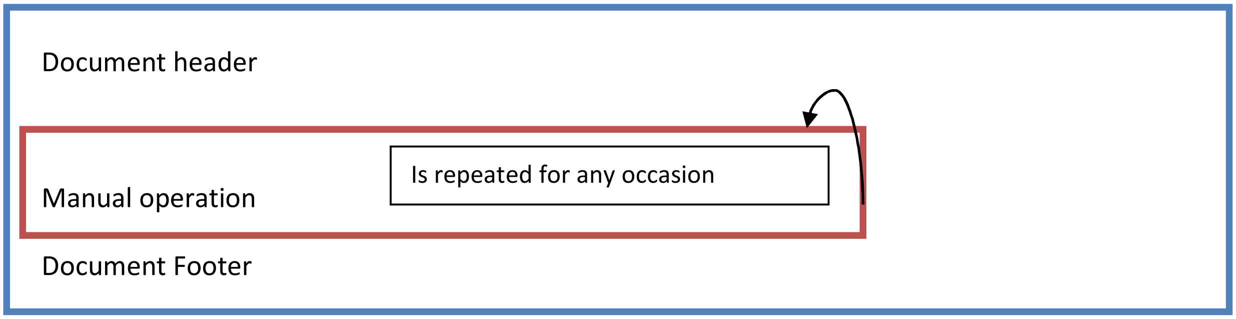
This template is added at the end of the document, and how the document header only access to common functions. Its purpose is to attach to any page numbers or sonsige information on the document.

5 The manual intervention print templates

The print templates for manual interventions are used to print out the data of manual intervention in the form are configured. The templates of Manual operations have basically the same function as the Batch Report Templates, but they are a much simpler structure.

5.1 The structure of the manual intervention

The structure of the manual intervention is divided into the following three templates, which are merged dynamically



5.2 Document header

This template is added each time you print the first thing and only once in the document. This template has access only to the general functions and does not have access to specific data.

This template is meant to be here, the registered company logos and company addresses. Also other things like dates of expression, etc. can be specified here.

5.3 Manual Operations

This template is added once for each Manual Operations event and has access to the data of the event, which is attached to the document. It serves to format the data of manual intervention in the document.

5.4 Document Footer

This template is added at the end of the document, and how the document header only access to common functions. Its purpose is to attach to any page numbers or various information on the document.

6 The Batch Summary Reports

The batch summary reports, or Batch Summary Reports, are used to display the data of the last batches in a more compact form. You basically contain the same data as the other reports; however, these are displayed in a filtered and summarized form.

The Batch Summary Reports are generally based on the same mechanism as the print templates for batch reports and manual interventions. There are basically the same token used, but the reports draw on a special set of functions. In addition, the Batch Summary reports will not be merged from multiple templates, but there are in principle only single templates. This template can also very well span several pages.

7 Script Reference

These functions are available at any of the aforementioned documents and reports at any time, regardless of its function. Those functions can be combined in conditionals, and simple tokens.

7.1 Date and Time

Now () As Date Time

Returns the current date and time.

Today () As Date Time

See Now ()

Date (year As Integer, month As Integer, Day As Integer) As DateTime

Erzugt a new date by which the respective components are passed.

Year (d As datetime) As Integer

Returns the year of the given date

Month (d As datetime) As Integer

Returns the month of the given date

Day (d As datetime) As Integer

Returns the day of the given date

WeekDay (d As datetime) As System.DayOfWeek

Returns the week of the given date

7.2 Mathematical Functions

Sin (v As Double) As Double

Calculates the sine of v

Cos (v As Double) As Double

Calculates the cosine of v

Tan (v As Double) As Double

Calculates the tangent of v

Asin (v As Double) As Double

Computes the Inverse of the sine of v

Acos (v As Double) As Double

Computes the Inverse of the cosine of v

Atan (v As Double) As Double

Computes the Inverse of the tangent of v

Ceiling (v As Double) As Double

Rounds V to the next straight ganzahligen worth

Floor (v As Double) As Double

Rounds V to the next straight ganzahligen value

Sign (v As Double) As Double

Returns 1 if it is a positive number, 0 for 0 and -1 if v negative

Log (v As Double) As Double

Calculates the natural logarithm of v

Log10 (v As Double) As Double

Calculates the 10th Logarithmums of v

Rnd (v As Double) As Double

Rounds V to nearest integer

Mod (x As Double, y As Double) As Double

Returns the remainder of a division. Which is also known as modulo.

Paragraph (val As Double) As Double

Returns the absolute warden back of v

Trunc (value As Double, Optional prec As Integer = 0) As Integer

Cuts all decimal places from which are less than 10-prec

Round (value As Object) As Double

See Rnd ()

Min (v1 As Double, v2 As Double, ...)

Returns the largest of the specified values. Up to 5 values are passed to the function.

Max (v1 As Double, v2 As Double, ...)

Returns the lowest of the specified values. Up to 5 values are passed to the function.

Exp (Base As Double, PEXP As Double) As Double

Calculates the exponent of the two values

Sqrt (v As Double) As Double

Calculates the square root of two values

Pow (v As Double, e As Double) As Double

See Exp

Pi As Double

Specifies the number pi back

7.3 Strings

Trim (str As String) As String

Removes all non-empty sign, tabulators and other non-printable characters from the beginning and end of the string

LeftTrim (str As String) As String

Removes all non-empty sign, tabulators and other non-displayable characters from the beginning of the string

RightTrim (str As String) As String

Removes all non-empty sign, tabulators and other non-displayable characters from the end of the string

Lower (value As String) As String

Converts all characters to lowercase

Upper (value As String) As String

Converts all characters to uppercase

Left (value As String, Length As Integer) As String

Returns a number of characters from the beginning of the string

Right (value As String, Length As Integer) As String

Returns a number of characters from the end of the string

Mid (value As String, Start As Integer) As String

Are all the characters back from a certain character position

Mid (value As String, Start As Integer, Length As Integer) As String

Returns a number of characters from a given character position

InStr (string1 As String, string2 As String) As Integer

Returns the position of the character on which the String2 found in String1

Len (Str As String) As Integer

Specifies the length of the string

Replace (Base As String, As String search, repl As String) As String

Replaces all existing search in base with repl