*Jo*PPS Windows Reporting

Index

1. Introduction

The J*o*PPS report generation phase uses a built-in report generator to present results to the end-user, which can be of various output types.

The J*o*PPS calculation phase dumps its results to temporary database tables on disk, called the *Result Database.*

The report generation phase will update all selected reports and format results according to predefined layout definitions.

The output of this process can be:
- HTML files
- Labels files (HTML based)
- Text files
- JoPPS-Script sources

For each report to be processed the built-in report generator is used at least once. The next chapter will discuss how the report generator is used to generate the result for a single report.

The report generation phase is invoked by:
- Recalculating, e.g. updating the result database.
- Regenerate a different selection of result reports.

Results are viewed using a built-in viewer capable of showing both HTML and plain text files. Only one result at a time can be displayed.

The internal J*o*PPS *SaveToDisk* flag determines whether or not results are written to disk. The state of this flag can be set in the JoPPS program.

The calculation and report generation phase can both be invoked manually or from within J*o*PPS-Script macro's. For documentation about J*o*PPS-Script, see the separate document available.

2. Filenames and locations

For every report that is requested, the report generator in J*o*PPS will "execute" certain files, files with a .QRF (Query Report File) extension. In short, QRF files contain the SQL query/queries to run on the Result Database and/or tell which layout file(s) take(s) part in the formatting of the result. Additionally, they can include QIF files (Query Information files) containing definitions usable in the QRF file itself or the *layout files*.

Some reports will have multiple QRF files, which each corresponds with a certain level in the project structure (batch/project level, assembly level, and frame level). The report generator will iterate through the project's structure in a hierarchical manner, and for every suitable component (batch/project, assembly, frame) a QRF file, if one exists for the current level, will be executed.

This chapter will explain the file types, where to search for the files and explain which files (if existent) will be executed for a given report.

2.1. Extensions

	Full name	*Description*
.QRF	Query Report File	main report file which is launched and takes control
.QIF	Query Information File	contains *defines* and *fields*, to be included by a QRF file
.H*n*	HTML Layout File	HTML templates (*n* represents the language)
.L*n*	Label Layout File	HTML label templates (*n* represents the language)
.L	Label Layout File	like L*n*, but not language dependent (e.g. label.l)
.T*n*	Text Layout File	Text templates (*n* represents the language)
.J	J*o*PPS-Script Layout File	J*o*PPS-Script templates
.BMP	Windows Bitmap File	
.GIF	GIF Image File	
.JPG	Jpeg Image File	

The *n* in .H*n*, .L*n* and .T*n* represents the language. By default, 1 = Dutch, 2 = French, 3 = German, 4 = English and 5 is not in use. For some countries, other combinations of languages are possible. Check this by viewing a *layout file* and look for fixed texts (usually between `<JOPPS>` and `</JOPPS>` tags), or by asking your supplier.

2.2. Location

Except for the order module reporting, all files associated with reports are searched for in the given search paths, set by *JoPPS Administrator* on the *Data* level (they are database dependent) under *File paths*.

For **J*o*PPS** result reports the paths after "Reports results" (default **`%PROGRAM_ROOT%\rpt`**`\custom;` **`%PROGRAM_ROOT%\rpt`**`\alu;` **`%PROGRAM_ROOT%\rpt`**`\common` where `%PROGRAM_ROOT%` is replaced by the path to the program files e.g. c:\joppswin) are being used.

For **Data Reporter** reports the paths after "Reports DReporter" (default **`%PROGRAM_ROOT%\template`**`\custom;` **`%PROGRAM_ROOT%\template`**`\alu;` **`%PROGRAM_ROOT%\template`**`\common`) and "Reports Results" are being used after each other.

The report files for the order module have their own default search path, set by *JoPPS Administrator* on the *General* level (they are being used for all databases and users) under *Order Module* (default **`%PROGRAM_ROOT%\rpt`**`\custom;` **`%PROGRAM_ROOT%\rpt`**`\order.rpt`).

Every file, needed by the report generator, will always be searched for in the following order:

1. The path(s) given in the optional -RPTPATH<paths> command-line parameter (see the documentation on command-line parameters) in their given order

2. The path(s) in J*o*PPS Administrator (after "Reports results" for J*o*PPS, after both "Reports DReporter" and "Reports results" for Data Reporter or under "Order Module" for the Order module) in their given order By default, the order for J*o*PPS Result Reports is:
   a. `%PROGRAM_ROOT%\rpt\custom`: this directory is always installed empty and is intended to store all customer specific and/or changed files
   b. `%PROGRAM_ROOT%\rpt\`**`alu`**: this directory contains all the files specific to the kind of material used (in this case ALU for Aluminium, can also be PVC or WOOD)
   c. `%PROGRAM_ROOT%\rpt\common`: this directory contains all files that are common to all report systems (all kind of materials). This directory contains most of the files. Some files are present here and also in a specific directory (to override the common files). Sometimes a file is not present in the common directory but is present in ALL specific directories (ALU,PVC,WOOD)

Under normal circumstances, you will have to search for a specific file in the specific directory (ALU, PVC or WOOD) and if it is not there, in the COMMON directory.

2.3. Filenames in JoPPS

Depending on which report and on which level of the project structure you want to report something, the name of the QRF file which can be run changes. Only QRF files that really exist are executed.

Filenames are generated as follows:

`<ReportName>_<ExecutionLevel>.QRF`

`<ReportName>` is 1 of the following:

Tab	*Report*	*<ReportName>*
HTML	Outlines	`outlines`
	Bill of material	`bill_of_material`
	Cutting list	`cutting_list`
	Accessories list	`accessories_list`
	Glass order	`glass_order` (Warning: different from v1)
	Window finishes list	`finishing_list`
	Part per page	`frame_per_page`
	Group per page	`assembly_per_page`
	Order list per supplier	`order_list_per_supplier`
	Order list per product	`order_list_per_product`
	Optimization	`optimization`
	Estimation detail	`estimation_detail`
	Estimation overview	`estimation_summary`
	Calculation detail	`calculation_detail`
	Calculation overview	`calculation_summary`
	Sections	`section`
	Offer	`offer`
	Offer sketches	`offer_outlines`
	Invoice	`invoice`
	Confirmation	`confirmation`
	Delivery note	`delivery_note`
	Statistics	`statistics`
	Operations	`machine`
	Glazing order	`filling_list`
	Profile list	`profile_list`
Labels	Cutting list	`lbl_cutting_list`
	Optimisation	`lbl_optimization`
Text	Order list per supplier	`order_list_per_supplier`
	Order list per product	`order_list_per_product`
	Own report 1	`user1`

	Own report 2	user2
	Own report 3	user3
	Own report 4	user4
	Own report 5	user5
	Operations	Machine
JoPPS-Script	Offer	Offer
	Order list per supplier	order_list_per_supplier
	Order list per product	order_list_per_product

Note: The actual list of available reports depends on the JoPPS license purchased and on the reports you are allowed to run by JoPPS Administrator on the *Accesses* level (user and database dependent) under *Access Rights* | *Reports*.

Some reports that exist in more then 1 tab have the same <ReportName>, which means they are using the same set of QRF filenames.

<ExecutionLevel> can take the following values during project(s) enumeration:
(the characters b and e stand for "begin" and "end")

<ExecutionLevel>	*Description*
b_project or b_batch	1x b_project for the project (not batchmode), or 1x b_batch for the batch (batchmode), BEFORE all groups of this project or batch
b_assembly	1x for each group of the current project or batch, BEFORE all frames of this group
b_frame	1x for each frame of the current group
e_frame	1x for each frame of the current group
e_assembly	1x for each group of the current project or batch, AFTER all frames of this group
e_project or e_batch	1x b_project for the project (not batchmode), or 1x b_batch for the batch (batchmode), AFTER all groups of this project or batch

When NOT calculating in batchmode (thus calculating a single report), the following order is maintained:
1) b_project for the current project
2) b_assembly for the first group of the project
3) b_frame and e_frame for the first frame of this group
4) repeat step 3) for all other frames of the same group
5) e_assembly for the current group
6) repeat steps 2) through 5) for all other groups of the project
7) e_project for the current project

When calculating in batchmode, the same order is maintained, but INSTEAD of b_project, **b_batch** is run, and INSTEAD of e_project, **e_batch** is run. Currently, most b_batch and e_batch QRF files will simply include b_project respectively e_project, so that these files do not have to be duplicated entirely. The order looks as follows:
1) b_batch
2) b_assembly for the first group of the batch
3) b_frame and e_frame for the first frame of this group
4) repeat step 3) for all other frames of the same group
5) e_assembly for the current group
6) repeat steps 2) through 5) for all other groups of the batch
7) e_batch

Some reports are unavailable in batchmode.

When calculating a single group, this group is being calculated as if it was the only group of the current project, while the calculation mode (batchmode or not) is taken into account.

2.4. Filenames in the Order Module

(yet to be described)

2.5. Filenames in Data Reporter

Currently, there is one QRF file for each table of basic data, which will operate on the current selected database.

2.6. Continuation sign

The following files are handled for continuation signs before actual execution: the QRF file being executed, any file included via the `include` keyword, any *layout file* used via the `using` keyword, any *layout file* included via the `%INCLUDE{}` function. These keywords and functions will be explained in the next chapters.

Before being processed, all lines in such a file are handled for continuation signs. Whenever the last non-blank character of a line is a `#`, the next line is treated as if it was on the same line on the position of the `#` character (instead of `#`), starting from the first non-blank character of that next line.

Example

```
This line is built from 2 lines but is han#
    dled as 1 line.
```

is handled as:

```
This line is built from 2 lines but is handled as 1 line.
```

2.7. Comment sign ;

In a QRF file, comments are allowed: if a line starts with a semicolon `;`, then the remainder of the line is treated as comment.

3. Keywords and sections in QRF files

A QRF file contains different sections separated by keywords.

```
include
defines
fields
report
calculate
using
next
```

These keywords are only recognised when at the first non-blank character of the line. Character case does not matter.

The `using` section is the only mandatory section. A QRF file can contain more then 1 `using` section: in this case the `next` keyword subdivides the QRF file, so that all other sections can be associated with their own `using` section.

The `include` keyword does not start or end a section, and the `next` keyword only ends the current one (it must be followed by another keyword which starts a section).

3.1. include

Syntax: **include <filename>**

<filename>

Includes the lines in the file `<filename>` (substitutable text) as if they were inserted on the line of the `include` statement. If `<filename>` is `nul`, then nothing is included.

Example 1

```
include jopps.qif
```

Example 2

```
include %REPORTNAME%_B_PROJECT.QRF
```

3.2. defines

Syntax: **defines**
<define1> = <value1>
<define2> = <value2>
…

<define*n*>, <value*n*>

Declares the *defines* on the lines following `defines`, until another keyword is read. 0 or more *defines* are allowed. `<define1>` is immediately set to `<value1>`, `<define2>` is immediately set to `<value2>`, etc. The values are substitutable texts and can use all previously defined *defines* and *parameters*. Whenever `%<define1>%` is found in substitutable parts during further execution of the report, `<value1>`, etc will replace it.

Use uppercase names for *defines*.

Example

```
defines
  C_REAL_LOSS        =2
  FD_PROFILE         =LEFT(4)
  FM_MEASURE         =mm
  FD_MEASURE         =DOUBLE(8,1,"%%s%FM_MEASURE%")
  DB_PIECE_WIDTH     =BREEDTE
  DB_PIECE_THICKNESS =DIKTE
```

3.3. fields

Syntax: **fields**
<field1> = <value1> display as <formatting1>
<field2> = <value2> display as <formatting2>
…

<field*n*>, <value*n*>

Declares the *fields* on the lines following `fields`, until another keyword is read. 0 or more *fields* are allowed. Unlike *defines*, *fields* are re-evaluated when they are needed for every record of the main query's result and for all *break headers* and *footers*, and are formatted. This means that the values are not immediately assigned to their field, but rather that the complete definition of the value (substitutable) is stored without immediate substitution, along with the desired formatting (which is immediately substituted). Whenever a field is requested (e.g. as `%<field1>%`), the value (e.g. `<value1>`) is substituted, formatted and inserted into the result. The values also get substituted and formatted only once per record for performance reasons. The result is stored for future use within the same record.

The values can use all *parameters*, *defines* and *fields* known at the moment they are first requested. Double quote characters can enclose the values **""** to denote a string value rather then a numeric value. This way, the values will be correctly interpreted when comparing them, etc. Without double quote characters, a value containing only numeric digits will be handled as a numeric value.

You cannot use *fields* inside *defines*, but you can use *fields* inside other *fields*. For formatting, you can only use all previously defined *defines* and *parameters*.

Use uppercase names for *fields*.

<formatting*n*>

The substituted values can be formatted in the following ways:

<formattingn>	*Description*
EXTERNAL	Calls J*o*PPS to generate a block of HTML code (see below).
ENUM(*n*)	Gets a text from a list of texts (the *fields'* value is the index in the list, *n* indicates from which list).
BLOB(*n1*,*n2*,*n3*,*n4*,*n5*)	Calls J*o*PPS to generate a block of HTML code, based on a table of basic data, depending on the fieldname specified as the value to format: *n1* = show this blob (1) or not (0) *n2* = detail level: no descriptions (0 or 1) or full descriptions (2) *n3* = borders on (1) / off (0) *n4* = currently not used (specify 0) *n5* = tabled-id from which to read the blob-field specified in the *fields'* value (110=accessories product, 111=accessories table, 108=glazing, 109=window finish, 107=profile glazing bead, 102=profile product, 105=profile properties, 116=profile reinforcement)
IMGREF(*n1*,*n2*,*n3*)	Generates a bitmap image depending on the *fields'* value (see below): *n1*,*n2*,*n3* = currently not used (specify 0 for all).
MAIL	E-mail address as HTML hyperlink.
MEMO	Large string (does no additional formatting).
LEFT(*n*)	Left-aligns the string by adding spaces and limits it to *n* characters.

RIGHT(*n*)	Right-aligns the string by adding spaces and limits it to *n* characters.
CENTER(*n*)	Centres the string by adding spaces and limits it to *n* characters.
INT(*n*) or **INT**(*n*,*f*)	Right-aligns the integer value and limits it to *n* characters, then (2nd form) applies formatting *f* (see below) to the result.
INT0(*n*) or **INT0**(*n*,*f*)	Right-aligns the integer value and limits it to *n* characters or expands it with leading zeroes to *n* characters, then (2nd form) applies formatting *f* (see below) to the result.
DOUBLE(*n1*,*n2*) or **DOUBLE**(*n1*,*n2*,*f*)	Rights-aligns the numeric value and limits it to *n1* characters, of which *n2* decimals and a decimal symbol (as specified in Regional Settings) if necessary, then (2nd form) applies formatting *f* (see below) to the result.
MONEY(*n1*,*n2*) or **MONEY**(*n1*,*n2*,*f*)	Rights-aligns the numeric value and limits it to *n1* characters, of which *n2* decimals, a decimal symbol and digit grouping symbol(s) (as specified in Regional Settings) where necessary, then (2nd form) applies formatting *f* (see below) to the result.
DATE(*f*)	Formats a date/time value applying the formatting *f* (see below); the date/time value is actually a floating point number, where the integral part is the number of days that have passed since 12/30/1899, and the fractional part is the fraction of a 24 hour day that has elapsed.

EXTERNAL formatting

If the substituted value is 1 of the following, the following will be generated:

value	*replaced by …*
BLOCKINFO:#<Prj>;<Grp>;<Frm>	Generates a HTML table containing price block information for the given frame part <Prj><Grp><Frm>
BATCHFRAMEORDER:#<Prj>;<Grp>;<Frm>	[Batchmode only] Generates a code representing the order in the batch of the given frame (project <Prj> group <Grp> frame <Frm>), which consists of a single number (e.g. 7) or a range of numbers (e.g. 8..10).
CFVALUE:<Prj>;<Grp>;#<Part>;<Fld>;<Filter>#<Format>[;<Type>[;<DepNo>[;<SeqNo>[;<Route>]]]]	Report function "CFVALUE:" allows you to easily retrieve the value of certain fields from the results table PART in a report. <Fld> 1 = light transmission factor 2 = solar factor 3 = surface 4 = surface 1 5 = surface 2 6 = 1x width + 1x height 7 = 2x height 8 = 2x width + 1x height 9 = width 10 = height 11 = perimeter 12 = 1x width + 2x height 13 = 2x width 14 = total weight 15 = frame weight 16 = glazing weight 17 = profile surface 1 18 = profile surface 2 19 = profile surface 1+2 20 = sound reduction index <Filter> filter 201 = project information 202 = group information 203 = frame part information (default if not specified) 204 = frame profile information 205 = frame opening information

	207 = sash part information 208 = sash profile information 209 = sash opening information 401 = frame model information 402 = sash model information <Format> accuracy (standard 2 decimal places) <Type> profile, filling,frame or sash type (if not specified = all) 0..25 = profile function - PROFILE PROPERTIES (0=profile, …) 0..6 = filling function - GLAZING (…, 1=Glasss, …) 0..1 = frame function (0=create, 1=buy) 0..17 = sash function (0=Fixed, …) <Department> DepNo profile, filling, frame or sahs departement (if not specified = all) <Serial number> SeqNo profile, filling, frame or sash serial number (if not specified = all) <Route> profile, filling, frame or sash route (if not specified = all) 0 = assembled 1 = separately 2 = direct 3 = purchased 4 = accessories 5 = deliver
DXFEXISTS:<Usg>;<Dxf>	Whether the specified DXF file is loaded or not (returns 1 or 0): <Usg> is 0 for section and 1 for view, <Dxf> is the actual code to look up.
INFO:<Prj>;<Grp>;<Frm>;#<Pie>;<Acc>;<Gla>;<Fin>	Generates a HTML table containing information about the given group (project <Prj> group <Grp>) if <Frm> is empty, or frame (project <Prj> group <Grp> frame <Frm>); also contains piece (<Pie>), accessory (<Acc>), filling (<Gla>) and window finish (<Fin>) information for all except the given department numbers (comma-delimited list of department numbers).
PREVIEW:<Id>;<Code>;<Mask>;<Size>;<Usage>	Creates a bitmap of the article. The additional information to be provided for this function: article type (105=profile, 110=accessories, 108=filling or 109=finishing), article code, article mask (possibly extra code, for profile series), width and height, type sketch (0=cross-section drawing, 1=view drawing).
SECTION:<Prj>;<Grp>	Generates a HTML table containing a list of the available sections for the given group (project <Prj> group <Grp>)
SECTIONPLD:<Prj>;<Grp>	Generates JoPPS-Script code containing a list of the available sections for the given group (project <Prj> group <Grp>), reserved for future use
SHOWVIEWPOINT:<Prj>;<Grp>;<Frm>;<Scenario>;<Side>	Checks whether visual information is requested. The additional information to be provided for this function: project code, group code, part code, sketch scenario and side (1=left, 2=Right, 3=Under and 4=Above)
VIEWPNT:<Prj>;<Grp>;<Frm>;<Width>;<Height>;<Size>;<Side>;<Dim>;<Color>;<Scale>;<Colsim>;<Mode>;<Resol>;<Scenario>	Generates bitmap/metafile of sight information. The additional information to be provided for this function: project code, group code, part code, width and height sketch, space visibility information, side view information (1=left, 2=Right, 3=Under and 4=Above), measuring lines (0=No, 1=Yes), sketches in color (0=No, 1=Yes), scale, color simulation (0=No, 1=Yes), inside/outside view (-1=according to group, 0=inside view and 1=outside view), resoltution (defaqult=120dpi), sketch scenario
SOUND:<Prj>;<Grp>;<Frm>#[;<Open>[;<VentPart>[;<VentOpen>[;<Format>]]]	The function "SOUND:" allows the user to retrieve the sound insulation value of a filling.

SOUNDEXISTS:<Prj>;# <Grp>;<Part>[;<Open># [;<VentPart>[;<VentOpen># [;<Show>]]]]	The function "SOUNDEXISTS:" allows the user to check whether the sound insulation value of a filling is available.
SECTIONPLD:<Prj>;<Grp>	Generates J*o*PPS-Script code containing a list of the available sections for the given group (project <Prj> group <Grp>), reserved for future use
UVALUE:<Prj>;<Grp>;# <Frm>	Function to report the available U-value
UVALUEEXISTS:<Prj>;# <Grp>;<Frm>;<show>	The function " UVALUEEXISTS:" allows the user to check whether the U-value is available.

means the line is not finished here: the next line should be appended at this position

IMGREF formatting

If the substituted value starts with 1 of the following, the requested bitmap is created and the following code will be generated:

Value	*replaced by …*
DXF_CAD:<Dxf>	When SaveToDisk is on, creates the DXF section for <Dxf> as a bitmap and generates HTML code to include the bitmap file. When SaveToDisk is off, generates HTML code which forces the internal HTML Viewer to draw the DXF section when needed.
DXF_DRW:<Dxf>	When SaveToDisk is on, creates the DXF view for <Dxf> as a bitmap and generates HTML code to include the bitmap file. When SaveToDisk is off, generates HTML code which forces the internal HTML Viewer to draw the DXF view when needed.
DRAWING:<Prj>;<Grp>;# <Frm>;<Wid>;<Hei>;<Mea>;# <Col>;<Sca>;<Cod>;<Ang>;# <Rei>;<GlB>;<Sim>;<Vie>;# <Res>;<Atm>;<Scn>;<Mrk>;# <Btp>;<Fdr>;<Vdr>;<Vis>	When SaveToDisk is on, creates the drawing of the given group (if <Frm> is left empty) or frame as a bitmap and generates HTML code to include the bitmap. When SaveToDisk is off, generates HTML code which forces the internal HTML Viewer to draw the drawing when needed. The drawing is generated for the given group (project <Prj> group <Grp>) or frame (project <Prj> group <Grp> frame <Frm>), with a width of <Wid> * 1 mm, a height of <Hei> * 1 mm, measures <Mea> on (1) or off (0), colours <Col> on (1) or black-and-white (0), a scale of <Sca>, show codes <Cod> for none (0), glazing (1), profiles (2) or both (3), display of angles <Ang> on (1) or off (0), reinforcements <Rei> on (1) or off (0), glazing beads <GlB> on (1) or off (0), real world color simulation <Sim> on (1) or off (0), view mode <Vie> on (1) or off (0), resolution <Res> default 120dpi, marked atom <Atm> is primary element to be highlighted, scenario <Scn> used for drawing.(0..8), marked atom <Mrk> is secondary atom to be highlighted, additional filter <Btp>, frame reference <Fdr> for dimensions (1=frame, 2=part), vent reference <Vdr> for dimensions (0=vent, 1=frame, 2=part),generated outline visible <Vis> on (1) or off (0)
PICTURE: . . . (same parameters as DRAWING:)	Does the same as DRAWING:, but is always handled as if SaveToDisk is on.
BMPCASE:<Prj>;<Grp>;# <Deel>;<Width>;# <Height>;<Measure># <Color>;<Scale># <Codes>;<Angle># <Steel>;<Bead># <Simul>;<View># <Resol>;<Mark># <Scenario>;<Highlight># <Batch>;<Frame># <Vent>;<Show>#	Creates a bitmap with framing part information without the sashes
BMPSASH:<Prj>;<Grp># <Deel>;<Open># <VlDeel>;<Width># <Height>;<Measure># <Color>;<Scale># <Codes>;<Angle># <Steel>;<Bead># <Simul>;<View># <Resol>;<Mark># <Scenario>;<Highlight># <Batch>;<Frame># <Vent>;<Show>#	Creates a bitmap with sash (part) information without the case frame
EMFCASE: . . . (same parameters as **BMPCASE:**)	Creates a metafile with framing part information without the sashes
EMFSASH: . . . (same parameters as **BMPSASH:**)	Creates a metafile with sash (part) information without the case frame
METAFIL: . . . (same parameters as DRAWING:)	Does the same as PICTURE:, but generates a Windows Metafile (EMF extension) instead of a bitmap.

FILLING:<Prj>;<Grp>;# <Frm>;<Ope>;<VPa>;<VOp>;# <Wid>;<Hei>;<Mea>;<Col>;# <Sca>;<Cod>;<Sim>	When SaveToDisk is on, creates the drawing of a given filling as a bitmap and generates the HTML code to include the bitmap file. When SaveToDisk is off, generates HTML code which forces the internal HTML Viewer to draw the drawing when needed. The drawing is generated for the given opening (project <Prj> group <Grp> frame <Frm> opening <Ope> and <VPa> and <VOp> left empty) or vent opening (project <Prj> group <Grp> frame <Frm> opening <Ope> vent part <VPa> vent opening <VOp>), with a width of <Wid> * 1 mm, a height of <Hei> * 1 mm, measures <Mea> on (1) or off (0), colours <Col> on (1) or black-and-white (0), a scale of <Sca>, show codes <Cod> on (1) or off (0), real world color simulation <Sim> on (1) or off (0).
BARCODE:<Typ>;<Sho>;# <Wid>;<Hei>;<Re1>;<Re2>;# <Co1>;<Co2>;<Co3>	Generates a barcode bitmap
BCDG244: . . . (same parameters as BARCODE:)	Does the same as BARCODE:, but
BCJELGO: . . . (same parameters as BARCODE:)	Does the same as BARCODE:, but

means the line is not finished here: the next line should be appended at this position

Special formatting *f* for INT(*n*,*f*), INT0(*n*,*f*), DOUBLE(*n1*,*n2*,*f*), MONEY(*n1*,*n2*,*f*)

f must be a double quote delimited formatting string containing %s, which will be substituted by the numeric value. An example is DOUBLE(8,1,"%%s%FM_MEASURE%"). Note that %s is written as %%s, which prohibits %s% to be substituted if the format string would be %s%FM_MEASURE%.

Special formatting *f* for DATE(*f*)

f must be a double quote delimited formatting string and can be built from the following specifiers:

Specifier	*Displays*
c	Displays the date using the format given by the short date style (Regional Settings), followed by the time using the format given by the long time style (Regional Settings). The time is not displayed if the fractional part of the date/time value is zero.
d	Displays the day as a number without a leading zero (1-31).
dd	Displays the day as a number with a leading zero (01-31).
ddd	Displays the day as an abbreviation (Sun-Sat) depending on the current locale in the Regional Settings.
dddd	Displays the day as a full name (Sunday-Saturday) depending on the current locale in the Regional Settings.
ddddd	Displays the date using the format given by the short date style (Regional Settings).
dddddd	Displays the date using the format given by the long date style (Regional Settings).
m	Displays the month as a number without a leading zero (1-12). If the m specifier immediately follows an h or hh specifier, the minute rather than the month is displayed.
mm	Displays the month as a number with a leading zero (01-12). If the mm specifier immediately follows an h or hh specifier, the minute rather than the month is displayed.
mmm	Displays the month as an abbreviation (Jan-Dec) depending on the current locale in the Regional Settings.
mmmm	Displays the month as a full name (January-December) depending on the current locale in the Regional Settings.
yy	Displays the year as a two-digit number (00-99).
yyyy	Displays the year as a four-digit number (0000-9999).
h	Displays the hour without a leading zero (0-23).
hh	Displays the hour with a leading zero (00-23).
n	Displays the minute without a leading zero (0-59).
nn	Displays the minute with a leading zero (00-59).
s	Displays the second without a leading zero (0-59).
ss	Displays the second with a leading zero (00-59).
t	Displays the time using the format given by the short time style (Regional Settings).
tt	Displays the time using the format given by the long time style (Regional Settings).

am/pm	Uses the 12-hour clock for the preceding h or hh specifier, and displays 'am' for any hour before noon, and 'pm' for any hour after noon. The am/pm specifier can use lower, upper, or mixed case, and the result is displayed accordingly.
a/p	Uses the 12-hour clock for the preceding h or hh specifier, and displays 'a' for any hour before noon, and 'p' for any hour after noon. The a/p specifier can use lower, upper, or mixed case, and the result is displayed accordingly.
ampm	Uses the 12-hour clock for the preceding h or hh specifier, and displays the contents of the AM Symbol (Regional Settings) for any hour before noon, and the contents of the PM Symbol (Regional Settings) for any hour after noon.
/	Displays the date separator character given by the date separator (Regional Settings).
:	Displays the time separator character given by the time separator (Regional Settings).
'xx'	Characters enclosed in single quotes are displayed as-is, and do not affect formatting.

Format specifiers may be written in upper case as well as in lower case letters--both produce the same result.
If the formatting string is empty (""), "c" is supposed.

Examples

```
defines
  FE_CLIENTTYPE   =ENUM(-9733)
  FD_EMAILADDRESS =MAIL
  FD_MEMO         =MEMO
  FD_QUANTITY     =INT(10,"%%s%FM_QUANTITY%")
  FD_MEASURE      =DOUBLE(8,1,"%%s%FM_MEASURE%")
  FD_DATE         =DATE("d/m/yyyy")

fields
  DSP_INFOASSEMBLY ="INFO:%PROJECT%;%ASSEMBLY%;;%VIMODE%;%AIMODE%" #
                                          DISPLAY AS EXTERNAL
  DSP_CLIENT_TYPE  = @%DB_CLIENT_TYPE%    DISPLAY AS %FE_CLIENTTYPE%
  DSP_ACC_VARIETY  =  %DB_ACC_VARIETY% #
                   DISPLAY AS BLOB(%SHOW_ACC_VARIETY%,%DETAIL_LEVEL%,%DETAIL_BBORDERS%,0,110)
  DSP_DRAWASSEMBLY ="DRAWING:%PROJECT%;%ASSEMBLY%;;%ASSEMBLY_PER_PAGE_WIDTH%;#
                     %ASSEMBLY_PER_PAGE_HEIGHT%;%GLOBAL_SCALES%;%GLOBAL_COLOREDOUTLINES%;;#
                     %ASSEMBLY_PER_PAGE_CODEINFO%;%ASSEMBLY_PER_PAGE_ANGLEINFO%;#
                     %ASSEMBLY_PER_PAGE_FORCEINFO%" #
                                          DISPLAY AS IMGREF(0,0,0)
  DSP_DRAWFILL     ="FILLING:@%DB_GLASS_PROJECT%;@%DB_GLASS_ASSEMBLY%;@%DB_GLASS_FRAME%;#
                     @%DB_GLASS_OPENING%;@%DB_GLASS_VENTPART%;@%DB_GLASS_VENTOPENING%;#
                     %ASSEMBLY_PER_PAGE_FILLING_WIDTH%;%ASSEMBLY_PER_PAGE_FILLING_HEIGHT%;1;#
                     %GLOBAL_COLOREDOUTLINES%;;%ASSEMBLY_PER_PAGE_CODEINFO%;0" #
                                          DISPLAY AS IMGREF(0,0,0)

  DSP_TEXT_EMAIL   ="@%DB_TEXT_EMAIL%"    DISPLAY AS %FD_EMAILADDRESS%
  DSP_FILL_MEMO    = @%DB_FILL_MEMO%      DISPLAY AS %FD_MEMO%
  DSP_PIECE_ID     ="@%DB_PIECE_ID%"      DISPLAY AS LEFT(40)
  DSP_PIECE_DEPNO  = @%DB_PIECE_DEPNO%    DISPLAY AS RIGHT(10)
  DSP_PIECE_NO     = @%DB_PIECE_NO%       DISPLAY AS %FD_QUANTITY%
  DSP_PIECE_LABS   = @%DB_PIECE_LABS%     DISPLAY AS %FD_MEASURE%
  DSP_TEXT_BEGIN   = @%DB_TEXT_BEGIN%     DISPLAY AS %FD_DATE%
```

3.4. report

Syntax: **report**
<first line of SQL query>
<second line of SQL query>
…

The report keyword is used to run a SQL query on the calculation results. All lines following the line with report, until another keyword is read, are handled as substitutable strings, which together make up the SQL statement. The result tables can be found in the path pointed to by the %PATH_TMP% system parameter.

The substitution of these lines must generate a valid SQL statement that upon execution returns a result table, otherwise an error is returned and further execution of the QRF file is aborted. Normally, the **SQL select statement** is used for this purpose. The query's result is then processed record by record using the various section of the *layout* (see the using keyword).

For a description of the SQL select statement syntax, see the separate chapter *SQL select statement*.

The fields in the query's result can be accessed by the query's *layout* using the `@` character (see next chapter).

Example

```
report
  select   min(%DB_PIECE_BATCH%) as %DB_PIECE_BATCH%,
           sum(%DB_PIECE_NO%) as %DB_PIECE_NO%,
           %DB_PIECE_SERIE%,
           min(%DB_PIECE_PROFILEDESC%) as %DB_PIECE_PROFILEDESC%,
           min(%DB_PIECE_LABS%) as %DB_PIECE_LABS%,
           min(%DB_PIECE_LMAX%) as %DB_PIECE_LMAX%,
           min(%DB_PIECE_LMIN%) as %DB_PIECE_LMIN%,
           %DB_PIECE_ANGLEB%,
           %DB_PIECE_ANGLEE%,
           min(%DB_PIECE_VARIETYDESC%) as %DB_PIECE_VARIETYDESC%,
           %DB_PIECE_PRODUCT%,
           %DB_PIECE_PROJECT%,
           %DB_PIECE_ASSEMBLY%,
           %DB_PIECE_FRAME%,
           min(%DB_PIECE_OPENING%) as %DB_PIECE_OPENING%,
           min(%DB_PIECE_VENTPART%) as %DB_PIECE_VENTPART%,
           min(%DB_PIECE_SYSTEM%) as %DB_PIECE_SYSTEM%,
           min(%DB_PIECE_PROFILE%) as %DB_PIECE_PROFILE%,
           %DB_PIECE_VARIETY%,
           min(%DB_PIECE_ID%) as %DB_PIECE_ID%,
           %DB_PIECE_CFCHAR%,
           %DB_PIECE_PIECENO%
  from     %TBL_PIECE%
  where    %DB_PIECE_PROJECT%  like '%PROJECT%'
  and      %DB_PIECE_ASSEMBLY% like '%ASSEMBLY%'
  and      %DB_PIECE_FRAME%    like '%FRAME%'
  and      %DB_PIECE_DEPNO% not in %FRAME_PER_PAGE_FILTER_PIECE%
  and      %DB_PIECE_NO% > 0
  group by %DB_PIECE_PROJECT%, %DB_PIECE_ASSEMBLY%, %DB_PIECE_FRAME%, %DB_PIECE_PRODUCT%,
           %DB_PIECE_VARIETY%, %DB_PIECE_SERIE%, %DB_PIECE_CFCHAR%, %DB_PIECE_PIECENO%,
           %DB_PIECE_LOUTPUT%, %DB_PIECE_ANGLEB%, %DB_PIECE_ANGLEE%
  having   count(*) > 0
```

3.5. calculate

Syntax: **calculate**
<break definition 1>
<break definition 2>
…

<break definition *n*> syntax:
[<SQL select fields phrase>] break on <Field1>[,<Field2>[,…]]

After a `report` section, a `calculate` section can occur. This can contain 1 or more *break definitions* of which each causes the report generator to split up the result into groups based on the contents of the fields specified after the `break on` keyword. The texts before and after `break on` are substitutable. When multiple *break definitions* are present, subsequent *break definitions* define subgroups of the previous group.

Each *break definition* corresponds with a *break header* and a *break footer* section in the *layout file* (see next chapter). Before each *break footer*, if an `<SQL select fields phrase>` is present, a sub-query is run on the current group of records.

<Field*n*>

While a query's result is processed on a record-per-record basis, the contents of all `<Fieldn>` is compared against the same `<Fieldn>` in the previous record. If the contents of 1 or more fields has changed, or if the contents of the fields from a higher-level *break* (closer to the `calculate` keyword) has changed, the condition for this *break* has been met and the *break level* becomes active.

<SQL select fields phrase>

On a group, when the corresponding *break footer* is about to be executed, a query is run on the records of the group only, to calculate totals and other information based on this group of records only. Specify the fields you want to use in the *break footer* (*layout file*) as `<SQL select fields phrase>`. Treat this as the part of the SQL select statement between the SQL `select` and `from` keywords, and as if you are selecting them from the main query's result. The sub-query is expected to return exactly 1 record (multiple fields allowed). It's fields are accessible via the `@` character, while the main query's fields stay accessible via `@@` (see next chapter).

Execution order

When a *break condition* has been met, the following actions will occur:

1) It is determined which *breaks* are becoming active, that is the highest level on which the change of field contents was detected, and all lower-level *breaks*.
2) The last record containing the old values is made the current one.
3) The lowest *break level* is made the current one.
4) If a `<SQL select fields phrase>` is present, a sub-query selecting these fields is run on the current group.
5) The *break level's* corresponding *break footer* section of the *layout file* (see next chapter) is executed.
6) Steps 4) and 5) are repeated for all higher *break levels* which also became active (low-to-high).
7) The first record of the next group becomes active (the record that is about to be processed).
8) The highest active *break* is made the current one.
9) The *break level's* corresponding *break header* section of the *layout file* (see next chapter) is executed.
10) Step 9) is repeated for all lower level *breaks* (high-to-low)

Example

```
report
  select   min(%DB_PIECE_BATCH%) as %DB_PIECE_BATCH%,
           sum(%DB_PIECE_CFLENGTH%) as %DB_PIECE_CFLENGTH%,
           %DB_PIECE_PRODUCT%,
           %DB_PIECE_SERIE%,
           min(%DB_PIECE_PROFILEDESC%) as %DB_PIECE_PROFILEDESC%,
           %DB_PIECE_ARTICLE%,
           sum(%DB_PIECE_CFPRICE%) as %DB_PIECE_CFPRICE%,
           sum(%DB_PIECE_TIME1%) as %DB_PIECE_TIME1%,
           sum(%DB_PIECE_TIME2%) as %DB_PIECE_TIME2%,
           %DB_PIECE_LINK%,
           %DB_PIECE_VARIETY%,
           %DB_PIECE_PROJECT%,
           %DB_PIECE_ASSEMBLY%,
           %DB_PIECE_FRAME%,
           min(%DB_PIECE_OPENING%) as %DB_PIECE_OPENING%,
           min(%DB_PIECE_VENTPART%) as %DB_PIECE_VENTPART%
  from     %TBL_PIECE%
  where    %DB_PIECE_DEPNO% > 0
  and      %DB_PIECE_NO% > 0
  group by %DB_PIECE_PROJECT%, %DB_PIECE_ASSEMBLY%, %DB_PIECE_FRAME%,
           %DB_PIECE_ARTICLE%, %DB_PIECE_PRODUCT%, %DB_PIECE_VARIETY%,
           %DB_PIECE_SERIE%, %DB_PIECE_LINK%
  having   count(*) > 0

calculate
  sum(%DB_PIECE_CFLENGTH%), sum(%DB_PIECE_CFPRICE%), sum(%DB_PIECE_TIME1%) as #
    %DB_PIECE_TIME1_SUM%, sum(%DB_PIECE_TIME2%) as %DB_PIECE_TIME2_SUM% #
    break on %DB_PIECE_BATCH%
  sum(%DB_PIECE_CFLENGTH%), sum(%DB_PIECE_CFPRICE%), sum(%DB_PIECE_TIME1%) as #
    %DB_PIECE_TIME1_SUM%, sum(%DB_PIECE_TIME2%) as %DB_PIECE_TIME2_SUM% #
    break on %DB_PIECE_PROJECT%, %DB_PIECE_ASSEMBLY%, %DB_PIECE_FRAME%
  sum(%DB_PIECE_CFLENGTH%), sum(%DB_PIECE_CFPRICE%), sum(%DB_PIECE_TIME1%) as #
    %DB_PIECE_TIME1_SUM%, sum(%DB_PIECE_TIME2%) as %DB_PIECE_TIME2_SUM% #
    break on %DB_PIECE_PROJECT%, %DB_PIECE_ASSEMBLY%, %DB_PIECE_FRAME%, %DB_PIECE_ARTICLE%
```

3.6. using

Syntax: **`using <filename>`**
or

```
using
  <layout line 1>
  <layout line 2>
  …
```

<filename>, <layout line *n*>

The `using` keyword is used to indicate the *layout file* `<filename>` (substitutable text) through which all records of the main query's result must be processed. If `<filename>` is `nul`, then no output will be generated. Instead of specifying `<filename>`, the entire *layout* can be added on the lines below the `using` keyword, until another keyword is read.

The next chapter will discuss the contents and sections of the *layout*. In all places where is referred to a *layout* file, it can also be the lines following the `using` keyword.

Example 1

```
using %REPORTNAME%_B_PROJECT.%REPORTTYPE%%LANGUAGE%
```

Example 2

```
using
  <page>
```

3.7. next

Syntax: **`next`**

The `next` keyword ends the current section (generally a `using` section), so that the following sections associate with the next `using` section.

4. Layout files

Layout files specified by the `using` keyword (and *layout lines* in a QRF after a `using` keyword) are subdivided into sections. The number of sections needed depends on the availability of a main query and the number of *break definitions* in the `calculate` section (QRF file) for this query.

To end a section a delimiter line must be used. For HTML reports (and HTML labels), this delimiter is a line beginning with `<!--%%`, for other reports a line beginning with `%%`. Other characters on these delimiter lines are ignored. The last section does not have to be ended by a delimiter line, except when it is empty. You must provide the exact number of sections needed.

4.1. Sections without a main query

When there is no main query available (no `report` section in the QRF file) **only 1 section** is used, which is output only once.

Example

```
<page>
```

4.2. Sections when a main query is available

When a main query is available, the following sections are output in the given order:
1) *header*: only once
2) *break header 1*, *break header 2*, …: if there are *break definitions* (specified by the `calculate` section), and if some *break levels* become active (when processing the first record, all levels are active), the *break header* sections for the corresponding active *break levels* will be output (from high to low)
3) *detail*: once per record
4) …, *break footer 2*, *break footer 1*: if there are *break definitions* (specified by the `calculate` section), and if some *break levels* become active (when the last record was processed, all levels are active), the *break footer* sections for the corresponding active *break levels* will be output (from low to high)
5) *footer*: only once

They are defined in the layout file in the following order:
1) *header*
2) *detail*
3) *break header 1*, *break footer 1*, *break header 2*, *break footer 2*, …
4) *footer*

Important: When the main query produces a result with no records at all, no single section is output, even not the header and footer sections!

Example

Table c:\windows\temp\test.db:

```
K1      K2      Val     Desc
--      --      ---     ----------------------
 0       0       12     Description of record 1
 0       1      201     Description of record 2
 0       1      128     Description of record 3
 0       3      -37     Description of record 4
 1       3       -7     Description of record 5
```

QRF file

```
report
  select *
  from "c:\windows\temp\test.db"

calculate
  max(Val) as MaxVal break on K1
  max(Val) as MaxVal break on K1, K2

using
This is the header of the example report<BR>
<!--%% detail------------------------------------------------------------>
......This is record %DETAILCNT%: @K1,@K2,@VAL,@DESC<BR>
<!--%% break header 1---------------------------------------------------->
..This is break header 1 (@K1)<BR>
<!--%% break footer 1---------------------------------------------------->
..This is break footer 1: the highest value for (@@K1) is @MAXVAL <BR>
<!--%% break header 2---------------------------------------------------->
....This is break header 2 (@K1,@K2)<BR>
<!--%% break footer 2---------------------------------------------------->
....This is break footer 2: the highest value for (@@K1,@@K2) is @MAXVAL<BR>
<!--%% footer------------------------------------------------------------>
This is the footer of the example report<BR>
```

The HTML output will look as follows:

This is the header of the example report
..This is break header 1 (0)
....This is break header 2 (0,0)
......This is record 0: 0,0,12,Description of record 1
....This is break footer 2: the highest value for (0,0) is 12
....This is break header 2 (0,1)
......This is record 1: 0,1,201,Description of record 2
......This is record 2: 0,1,128,Description of record 3
....This is break footer 2: the highest value for (0,1) is 201
....This is break header 2 (0,3)
......This is record 3: 0,3,-37,Description of record 4
....This is break footer 2: the highest value for (0,3) is -37
..This is break footer 1: the highest value for (0) is 201
..This is break header 1 (1)
....This is break header 2 (1,3)
......This is record 4: 1,3,-7,Description of record 5
....This is break footer 2: the highest value for (1,3) is -7
..This is break footer 1: the highest value for (1) is -7
This is the footer of the example report

5. Parameter syntax and function reference

5.1. Parameter syntax

What follows are some definitions, which upon successful substitution, replace the entire definition.

%PARAMNAME%, %FUNCNAME{argument1[,argument2[,…]]}, %%

All *parameters*, *defines* and *fields* are put between `%` characters: `%PARAMNAME%`. Use `%%` to output a single `%` character. Functions are written as `%FUNCNAME{argument1[,argument2[,…]]}` (see Function reference).

@*n*, @@*n*, @fieldname, @@fieldname

A `@` character is used to get a fields' contents of the main query. An exception is in a break footer section with a sub-query, where `@` is used to get a fields' contents from the sub-query, and `@@` to get a fields' contents from the main query. After `@` or `@@`, as long as only the characters `0` through `9`, `A` through `Z` and _ are used, they are treated as the fieldname or number of the field from which to return it's contents. If the characters can be treated as a number *n*, the content of the nth field is returned, otherwise the characters are treated as the fieldname of which to retrieve the contents. Use uppercase characters only for fieldnames.

Substitution order

When a substitution takes places, the following order of replacements is respected:

1) Functions are executed: functions nested inside other functions are executed first. Arguments of functions () will be substituted as needed while respecting this same substitution order. For every *field* that is needed somewhere and that has not yet been evaluated for the current record, this same substitution order will be respected.
2) The remaining parameters are substituted. For every *field* that is needed somewhere and that has not yet been evaluated for the current record, this same substitution order will be respected.
3) `@` and `@@` characters are processed to retrieve the requested field's contents.

In a QRF file, the texts that are substitutable are indicated (see the documentation on *Keywords and sections in QRF files* earlier in this document). In *layout files*, these substitutions occur on a line by line basis, while the continuation character `#` is preserved everywhere, as is the case for QRF files also.

5.2. Function reference

%EVAL{*expr*}

Passes *`expr`* to the internal expression evaluator, which is part of the *JoPPS-Script Language*. There is a separate document available describing its functionality. The result of the expression is returned.

E.g. `%EVAL{1+2*3}` returns `7`.

%TIMES{*count*,*text*}

Returns the text *`text`* *`count`* times. *`count`* may be zero.

E.g. `%TIMES{5,Hi}` returns `HiHiHiHiHi`.

%LOOKUP{*table*,**[***fieldname1*[;*fieldname2*[;…]]**]**,**[***value1*[;*value2*[;…]]**]**,*returnField*}

(The bold brackets **`[`** and **`]`** are part of the syntax and do not indicate and optional part of the syntax.)
Looks up a field *`returnField`* in the database table *`table`* and returns its contents, for which the fields' *`fieldname1`* contents equals *`value1`*, and the fields' *`fieldname2`* contents equals *`value2`*, etc. There should be as many *`value`* arguments as *`fieldname`* arguments. If no path information is specified for *`table`*, then the directory containing the basic data (such as `c:\joppswin\dbf\<DatabaseId>`) is supposed. The default extension is `.DB`.

E.g.
`%LOOKUP{%TBL_CLIENT%,[%DB_CLIENT_CODE%],["TECHWIN"],%DB_CLIENT_COEFFICIENT%}` returns the coefficient for client "TECHWIN".

%INCLUDE{*filename*}

Includes the contents of *filename* into the result, after handling its contents for the continuation sign # and substituting it line by line. If *filename* is nul, nothing happens. If the file is not found (it is searched for in the given search paths), an error occurs. When nesting %IF{} and %INCLUDE{} functions, the recommended practice is to use %IF{} inside %INCLUDE{} to generate the correct *filename*.

E.g. %INCLUDE{myfile.h} will return the 2 lines of the file "myfile.h" if this contains 2 lines.

%IF{*condition*,*doTrue*,*doFalse*}

Returns *doTrue* or *doFalse*, depending on the result of *condition*, which can be "True" respectively "False".
If *condition* contains a = or ! character, the parts before and after = / ! are evaluated. If both parts are of the same type (numeric or alphanumeric), the values are compared against each other, otherwise *condition* evaluates False. The comparison evaluates True when = was specified and both values are equal or when ! was specified and both values are different, in all other cases the comparison evaluates False.
If *condition* contains no = or ! character, then if it's a numeric value, *condition* evaluates True if the value is non-zero, or if it's an alphanumeric value, *condition* evaluates True if the value is not an empty string, in all other cases *condition* evaluates False.

If *condition* needs to be more complex, use the %EVAL{} function inside *condition*.

E.g. %IF{7,It's True,It's False} returns It's True.
E.g. %IF{0,It's True,It's False} returns It's False.
E.g. %IF{1=1,It's True,It's False} returns It's True.
E.g. %IF{A!A,It's True,It's False} returns It's False.
E.g. %IF{%EVAL{substr("test",2,1)="e"},It's True,It's False} returns It's True because the second character of "test" is an "e".

%FORMAT{*text*,**[***format***]**}

(The bold brackets **[** and **]** are part of the syntax and do not indicate and optional part of the syntax.)
The text *text* is formatted as indicated by *format* and returned. The possibilities for formatting are exactly the same as those after the display as clause of *field definitions*. Refer to the explanation of the fields keyword for a full description of its possibilities.

E.g. %FORMAT{12,[DOUBLE(8,2)]} returns ^^^12.00 or ^^^12,00 (depending on Regional Settings; every ^ represents a space).
E.g. %FORMAT{ABCDEFGHIJ,[LEFT(8)]} returns ABCDEFGH.
E.g. %FORMAT{DXF_CAD:123456,[IMGREF(0,0,0)]} returns the HTML code for DXF section 123456 and generates it as a bitmap (see IMGREF formatting for more details).

%LINE{*remark text field,line number*}

The %LINE{} function is provided to easily report a specific line from a multiple-line comment field.

E.g. %LINE{%ASSEMBLYREMARK_TEXT%,3} reports the third line from the Default Remark text field from the Group detail.

6. System and global parameters

6.1. System parameters

System parameters are parameters that cannot be changed by the user. They are either defined by JoPPS (Static system parameters) or maintained by the report generator during execution. Accessible for verification through Jopps menu help – System Information - %Parameters - System

Static system parameters

Parameter	Description	Example value
General paths		
%PROGRAM_ROOT%	JoPPS start-up directory	C:\JoPPSWIN
%SYSTEM_ROOT%	MS-Windows system directory	C:\WINDOWS
%SYSTEM_TMP%	Windows temporary directory	C:\WINDOWS\TEMP
%INI_FILE%	JoPPS Configuration settings file	C:\JoPPSWIN\JOPPS.INI
Version strings		
%VERSION%	Executable version	1
%DLLVERSION%	DLL version	1
%REVISION%	Executable revision	2
%BUILD%	Build date	1/04/99
%STATION%	Build station	0
Regional Settings (from Control Panel)		
%CURRENCYSTRING%	Currency string	Euro
%CURRENCYFMT%	Position of currency symbol	3
%NEGCURRFMT%	Negative number format	8
%THOUSANDSEP%	Thousand separator	.
%DECIMALSEP%	Decimal separator	,
%CURRENCYDECIMALS%	Currency decimals	2
%DATESEPARATOR%	Date separator	/
%SHORTDATEFMT%	Short date format	d/MM/yy
%LONGDATEFMT%	Long date format	dddd d MMMM yyyy
%TIMESEP%	Time separator	:
%TIMEAMSTRING%	AM time string	AM
%TIMEPMSTRING%	PM time string	PM
%SHORTTIMEFMT%	Short time format	h:mm
%LONGTIMEFMT%	Long time format	h:mm:ss
%LISTSEP%	List separator	;
Current date & time strings (start of report generation)		
%TIME%	Current time	8:57:30
%DATE%	Current date	1/04/99
%WEEKDAY%	Day of the week	4
%DOW%	Day of the week	4
%DAY%	Day of the year	91
%DOY%	Day of the year	91
%WEEK%	Week of the year	13
%WOY%	Week of the year	13
%D%	Day of the month	1
%DD%	Day of the month	1
%M%	Month of the year	4
%MM%	Month of the year	04

%YYYY%	Year (4 digits)	1999
%YY%	Year (2 digits)	99
%HOUR%	Hour	8
%MIN%	Minutes	57
%SEC%	Seconds	30

Current database related		
%DB_ID%	Database identification	TS
%DB_DESC%	Database identification description	TechWIN Software
%DB_PATH%	Database directory	C:\JoPPSWIN\DBF\TS
%DB_USER%	Database username	ADMIN
%DB_USERDESC%	Database username description	Administrator

Paths (settable by JoPPS Administrator or jopps.ini)		
%PATH_DBX%	Default directory import-export files	C:\JoPPSWIN\DBx
%PATH_JP%	Default directory project files	C:\JoPPSWIN\JP
%PATH_CAD%	Default directory section DXF files	C:\JoPPSWIN\DXF\CAD\…TS
%PATH_DRW%	Default directory drawing DXF files	C:\JoPPSWIN\DXF\DRW
%SEARCH_DATA%	Effective search path DATA files, equal to PATH_DATA corrected by commandline parameters	C:\JoPPSWIN\DATA;C:\JoPPSWIN\DATA\CUSTOM
%PATH_DATA%	Search path DATA files	C:\JoPPSWIN\DATA;C:\JoPPSWIN\DATA\CUSTOM
%PATH_TXT%	Default directory TEXT files	C:\JoPPSWIN\TXT
%PATH_OUTPUT%	Default directory OUTPUT files	C:\JoPPSWIN\OUTPUT
%PATH_TMP%	Temporary directory RESULT tables. Static system parameter	C:\WINDOWS\TEMP\12345678.TMP
%SEARCH_REPORT%	Effective search path REPORT files for results equal to PATH_REPORT corrected by commandline parameters	C:\JoPPSWIN\RPT\CUSTOM; C:\JoPPSWIN\RPT\COMMON; C:\JoPPSWIN\RPT\ALU(or PVC or Wood)
%PATH_REPORT% *or* %REPORTPATH%	Search path REPORT files for results	C:\JoPPSWIN\RPT\CUSTOM; C:\JoPPSWIN\RPT\COMMON; C:\JoPPSWIN\RPT\ALU(or PVC or Wood)
%PATH_HTML%	Default directory HTML output	C:\JoPPSWIN\RESULT
%PATH_RESULT%	Default directory OTHER output	C:\JoPPSWIN\RESULT
%PATH_HLP%	Default directory HELP files	C:\JoPPSWIN\HLP
%PATH_MSETUP%	Default directory for machine setup	C:\JOPPSWIN\MSETUP

System parameters maintained by the report generator during execution

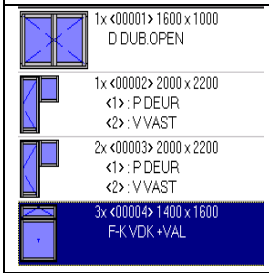
Parameter	***Description***	***Example value***

Initial parameters for current report		
%LANGUAGE%	Language for reports, language JoPPS is running in for normal reports, customer language for commercial reports	1 = Dutch, 2 = French, 3 = German, 4 = English, 5 = other
%NAME%	Name of the report	offer_b_project
%DESC%	Description regarding report	Offer
%REPORTTYPE%	Type of current report	H,T,L,J
%REPORTNAME%	Name of current report	OUTLINES
%REPORTDOC%	Filename current report document	C:\…\NEW PROJECT Outlines.htm
%RESULTSAVE%	State of SaveToDisk button	1

Calculation dependent parameters		
%PROJECTCNT%	Total number of projects to be processed	1
%MODUS%	Calculation modus (project, batch or price table)	1…3

Project level related parameters		
%BATCHMODE%	Batchmode enabled	0 or 1
%BATCH%	Batch identification	New Project

Assembly level related parameters (only available for assembly and lower levels)		
%ASSEMBLYCNT%	Assembly's sequence number in project or batch	Starts at 0, see below
%ASSEMBLYWIDTH%	Width current assembly in mm	50
%ASSEMBLYHEIGHT%	Height current assembly in mm	50
%ASSEMBLYTOT%	[Available for all levels] Number of assemblies in project or batch	Minimum 1, see below
%ASSEMBLYSUM%	[Available for all levels] Sum of assemblies in project or batch taking into account repetition factor of assembly	Minimum 1, see below

	ASSEMBLY COUNT	ASSEMBLY CNT	ASSEMBLY TOT	ASSEMBLY SUM	FRAME CNT	FRAME TOT	ALLFRAME CNT	ALLFRAME TOT
	1	0	4	7	0	1	0	6
	1	1	4	7	0	2	1	6
				7	1	2	2	
	2	2	4	7	0	2	3	6
				7	1	2	4	
	3	3	4	7	0	1	5	6

Frame level related parameters (only available for frame level)		
%FRAMECNT%	Frame's sequence number in current assembly	Starts at 0, see above
%FRAMETOT%	[Available for assembly and lower levels] Number of frames calculated in current assembly	Minimum 1, see above
%FRAMEWIDTH%	Width current frame in mm	50
%FRAMEHEIGHT%	Height current frame in mm	50
%FRAMEAREA%	Surface current frame in mm²	2500
%FRAMEEDGES%	Length equals 2 x height + 1 x width of frame	
%FRAMESIDES%	Length equals 2 x height + 2 x width of frame	
%ALLFRAMECNT%	Frame's sequence number in project or batch	Starts at 0, see above
%ALLFRAMETOT%	[Available for all levels] Number of frames calculated in current project or batch	Minimum 1, see above

Parameters depending on the main query's current record		
%DETAILCNT%	Counter for record number	Starts at 0
%SUBDETAILCNT%	Record counter since last BREAK	
%TOTRECCNT%	Number of records in query	Minimum 1

6.2. Global parameters

The user can change global parameters. They can be found in *Edit | Settings* in the tabs *Calculations* (Report: *Global*) and *Reports*.

Parameter	***Description***	***Example value***
HTML table settings		
%BACKGROUND%	Background colour of complete page	Standard = white
%COLOR_BACK%	Background colour intro header	
%COLOR_FORE%	Colour for everything in foreground like text	
%COLOR_LINK%	Colour of HTML-link	Blue
%COLOR_VISITED%	Colour of HTML-link already clicked on	
%COLOR_BORDER%	Border colour	
%WIDTH_BORDER%	Border width	
%COLOR_INFO%	Background colour info table	
%COLOR_BREAK%	Not for general use	
%COLOR_ALTBACK%	Alternate Background colour	
%COLOR_TITLEBORDER%	Title border colour	
%WIDTH_TITLEBORDER%	Title border width	
%COLOR_ANNOTATE%	Colour for special annotation	If group number is more then 1
%COLOR_TABLEHEAD%	Table header colour	
%COLOR_TABLEDATA%	Table data colour	
%COLOR_BDATA%	Blob data colour	
%COLOR_BHEAD%	Blob header colour	
%COLOR_BBORDER%	Blob border colour	
%WIDTH_BBORDER%	Blob border width	
%COLOR_RDATA%	Result data colour	
%COLOR_RBORDER%	Result border colour	
%WIDTH_RBORDER%	Result border width	
%DETAIL_LEVEL%	Depth of description	0 = no descriptions 1 = only the main descriptions 2 = As many descriptions as possible
%DETAIL_BORDERS%	Border on/off	
%DETAIL_BBORDERS%	Blob borders on/off	
%DETAIL_BLOBSON%	Blob data on/off	
%DPI%	DPI for sketches, fixed value	120
%SMALLLOGO%	Logo to be used in report headers	techwin16.bmp

Formatting		
%DEC_PRICE%	Number of decimals for prices	0
%FM_PRECURRENCY%	Format pre currency	Euro
%FM_POSTCURRENCY%	Format post currency	Euro
%FM_QUANTITY%	Format quantity suffix	x
%FM_PIECES%	Format pieces suffix	st.
%PREC_PRICE%	Price precision	0.01

%PROJECT%	Project name	NEW PROJECT
%PROJECTDESC%	Project description	
%PROJECTKIND%	Project kind (= private, seller, contractor, architect or supplier	0…4

%PROJECTSTATE%	Project state (= unknown, offer, order, production, delivery, invoice or settled)	0…6
%PROJECTCURRENCY%	Project currency	Euro
%PROJECTADDRESS1%	First line project address	TechWIN Software
%PROJECTADDRESS2%	Second line project address	Brusselse Steenweg 267
%PROJECTADDRESS3%	Third line project address	2800 MECHELEN
%PROJECTPHONE%	Telephone	
%PROJECTFAX%	Fax	
%PROJECTEMAIL%	E-mail	
%PROJECTDAYWEEK%	Delivery day or week	0
%PROJECTCOMMENT%	Project comments	
%PROJECTSERIES%	Default project series	CONCEPT SYSTEM 68
%PROJECTPROFILE%	Default profile colour on project level	REYNAERS WHITE
%PROJECTFRAMEPROFILE%	Default profile colour on frame level	REYNAERS WHITE
%PROJECTVENTPROFILE%	Default profile colour on vent level	REYNAERS WHITE
%PROJECTBEAD%	Default glazingbead colour on project level	REYNAERS WHITE
%PROJECTFRAMEBEAD%	Default glazingbead colour on frame level	REYNAERS WHITE
%PROJECTVENTBEAD%	Default glazingbead colour on vent level	REYNAERS WHITE
%PROJECTACCESS%	Default accessories colour on project level	REYNAERS WHITE
%PROJECTFRAMEACCESS%	Default accessories colour on frame level	REYNAERS WHITE
%PROJECTVENTACCESS%	Default accessories colour on vent level	REYNAERS WHITE
%PROJECTFILLING%	Default project glazing	DOUBLE GLASS 20MM
%PROJECTNORM%	Wind norm used for this project	NORMAL SITE
%ASSEMBLY%	Assembly identification	00001
%ASSEMBLYDESC%	Assembly description	FRONT DOOR
%ASSEMBLYCOUNT%	Number of times the assembly is asked for	Minimum 1, see above
%ASSEMBLYREMARK%	General remark with the assembly	Kitchen door second floor
%SECTIONCNT%	Internal use only	
%FRAME%	Frame identification	
%FRAMEDESC%	Frame description	
%BLOCK_LOSS%	Priceblock used for profile loss	90
%BLOCK_TIME1%	Priceblock used for first time	91
%BLOCK_TIME2%	Priceblock used for second time	92
%BLOCK_GLAZING%	Priceblock used for placement glass	93
%CLIENT%	Project clients identification	
%CLIENT_NAME%	Project clients description	
%CLIENT_CONTACT%	Project clients contact person	
%CLIENT_ADDRESS%	Project clients address	
%CLIENT_ZIP%	Project clients ZIP code	
%CLIENT_CITY%	Project clients city	
%CLIENT_PHONE%	Project clients telephone	
%CLIENT_FAX%	Project clients fax	
%CLIENT_EMAIL%	Project clients E-mail	

%GLOBAL_… (also used as %GLOBAL_..._TEXT%		
…COMPANY%	Company name	TechWIN Software BVBA
…STREET%	Company address	Brusselse Steenweg 267

…PLACE%	Company city	Mechelen
…ZIP%	Company ZIP code	B-2800
…PHONE%	Company telephone	(32)-015/44.64.64
…FAX%	Company fax	(32)-015/44.64.66
…EMAIL%	Company E-mail	info@techwin.be
…TAXNUMBER%	Tax number company	
…TRADEREGISTER%	Trade register number company	
…REGISTRATION%	Registration number company	
…ACCOUNT%	Account number company	
_DELSTREET%	Company streetname at delivery address	Brusselse Steenweg 267
…DELPLACE%	Company city at delivery address	Mechelen
…DELZIP%	Company ZIP code at delivery address	B-2800
_DELPHONE%	Company phone at delivery address	(32)-015/44.64.64
…DELFAX%	Company fax at delivery address	(32)-015/44.64.66
…DELEMAIL%	Company E-mail at delivery address	info@techwin.be
_DELMAILBOXADDRESS%	Mailbox streetname at delivery address	
_DELMAILBOXCODE%	Mailbox zipcode at delivery address	
_DELMAILBOXPLACE%	Mailbox city at delivery address	

Global Report Parameters **%GLOBAL_…**		
…LOSS%	Loss calculation (=fixed, absolute, real)	0…2
…LABELS%	Enables/disables labels (internal use only)	1 or 0
…SCALES%	Enables/disables scaling of outlines	1 or 0
…COLOREDOUTLINES%	Enables/disables colours in outlines	1 or 0
…OPTIMIZATION%	Enables/disables optimisation	1 or 0
…OPTMULTI%	Enables/disables multiple length combinations	default 2
…OPTMULTITOTLOSS%	Percentage loss tolerated for all bars	1 or 0
…OPTMULTICMPLVL%	Memory use for optimisation	0=low, .., 3=high
…OPTLOSSDEC%	Incremental step from allowed loss when using one length	default 2
%LABEL_OFSX%	Horizontal spacing to first label in mm	0
%LABEL_OFSY%	Vertical spacing to first label in mm	0
%LABEL_GAP%	Space between labels in mm	0
%LABEL_LEFT%	Left margin within label in mm	5
%LABEL_TOP%	Upper margin within label in mm	5
%LABEL_W%	Label width in mm	70
%LABEL_H%	Label height in mm	42.5
%LABEL_COLS%	Number of labels horizontally in mm	3
%LABEL_ROWS%	Number of labels vertically in mm	7
%LABEL_COLOR%	Background colour	$0080FFFF
%LABEL_BORDER%	Folding line	0 or 1
%SECTION_SCOPE%	Aperture of one section	200mm
%SECTION_FORMAT%	Separately, with view or in view	0…2
%SECTION_EXPLODE%	Enables/disables exploding blocks	1 or 0
%SECTION_VERSION%	DXF version (-,10,11,12-13,14)	0…4
%TEMPLATE_…		
…OUTLINE_SCALE%	Scale view	1/12
…HORIZONTAL_SCALE%	Scale horizontal sections	1/3
…VERTICAL_SCALE%	Scale vertical sections	1/3
%DIMENSION_…		
…FRAME%	Frame dimensions	0 or 1
…STIJL%	Frame profile positions	0 or 1
…VENT%	Vent dimensions	0 or 1
…POST%	Vent profile positions	0 or 1

…GLAS%	Glazing & profile dimensions	0 or 1
…HORIZONTAL%	Horizontal alignment measures	0…2
…VERTICAL%	Vertical alignment measures	0…3
…FONT%	Dimension font style	Arial
…SIZE%	Dimension font size	4mm
…SCALE%	Internal use only	
%GLOBAL_… This group of parameters is set in function of customer type		
…USEDISCOUNT%	Apply discount or not	0 or 1
…COEFEXTERNAL%	Method of coefficient calculation	0 = internal, 1= external
…USECOEF1%	Apply coefficient 1 or not	0 or 1
…CALCCOEF1%	How must coefficient 1 be applied	0 = on gross price, 1 = after calculation of discount
…DESCCOEF1%	Description coefficient 1	Coefficient 1
…USECOEF2%	Apply coefficient 2 or not	0 or 1
…CALCCOEF2%	How must coefficient 2 be applied	0 = on gross price, 1 = after calculation of discount, 2 = after calculation of coefficient 1
…DESCCOEF2%	Description coefficient 2	Coefficient 2
…USECOEF3%	Apply coefficient 3 or not	0 or 1
…CALCCOEF3%	How must coefficient 3 be applied	0 = on gross price, 1 = after calculation of discount, 2 = after calculation of coefficient 1, 3 = after calculation of coefficient 2
…DESCCOEF3%	Description coefficient 3	Coefficient 3
…USETIME1%	Show clause time 1 or not	0 or 1
…DESCTIME1%	Description time 1 if time 1 different from 0	included
…DESCTIME1_NINC%	Description time 1 if time 1 = 0	not included
…USETIME2%	Show clause time 2 or not	0 or 1
…DESCTIME2%	Description time 2 if time 1 different from 0	included
…DESCTIME2_NINC%	Description time 2 if time 1 = 0	not included

7. Jopps.qif

This file contains information, which is needed in all the report generation activities.

The file contains the following sections:

- defines:
  - indicates how certain values have to be presented
  - mostly regarding measurements
- floating-point formatting:
  - not used
  - are all set to zero
- field definitions:
  - sets length of commonly used string values
  - mostly regarding general project information
- field enums:
  - gets a text from a field of values
  - interpreted by program, language database dependent
- dialog references:
  - interpreted by program and data reporter, which generates a URL for which the appropriate dialog/record will be shown
- company information:
  - Global parameters transformed into easy to used fieldnames
  - personalising certain variables, like name of logo
- show blob fields:
  - to indicate of blob-fields should be shown or not
  - for description of blob-fields, see point 3.3 fields
- macro's:
  - predefines macro's for later use inside the lay out files
- constants:
  - pre-defines a range of constants used in all reports

8. Result database tables and QIF files

8.1. Acc.qif

Resides in Common. Used for templates.

8.2. Accset.qif

Resides in Common. Used for templates.

8.3. Acctbl.qif

Resides in Common. Used for templates.

8.4. Attrib.qif

Resides in Common.

DB_ATTRIB_BATCHNO	=BATCHNR	; Order number in batch of project
DB_ATTRIB_BATCH	=BATCH	; Batchname, typically first project in batch or chosen by user
DB_ATTRIB_ACCDESC	=BESLAGNAAM	; Accessory description
DB_ATTRIB_ACC	=BESLAG	; Accessory code
DB_ATTRIB_SERIE	=SERIE	; Accessory order code
DB_ATTRIB_LENGTH	=LENGTE	; Accessory length, typically for rubbers
DB_ATTRIB_PROJECT	=PROJECT	; Project name in which accessory is used
DB_ATTRIB_ASSEMBLY	=GROEP	; Number of the group where accessory is used
DB_ATTRIB_FRAME	=DEEL	; Number of the part where accessory is used
DB_ATTRIB_OPENING	=OPEN	; Number of the opening where accessory is used
DB_ATTRIB_VENTPART	=VLDEEL	; Number of the ventpart where accessory is used
DB_ATTRIB_BC	=BC	; For future use with barcodes
DB_ATTRIB_SUPPLIER	=LEVERANCIER	; Supplier code
DB_ATTRIB_SIZE	=SIZ	; Number of items to order in one go
DB_ATTRIB_RATE	=RATE	; Length or number of items, with rubbers, total length of rubbers
DB_ATTRIB_UNIT	=UNIT	; Purchase price
DB_ATTRIB_VARIETYDESC	=VARIANTNAAM	; Finish description
DB_ATTRIB_VARIETY	=VARIANT	; Finish code
DB_ATTRIB_VARIETYSERIE	=VARIANTSERIE	; Finish code
DB_ATTRIB_LINK	=LINK	; Link through which accessory is connected to accessory set or one profile to another (for example glass bead linked to vent profile)
DB_ATTRIB_FLAG	= FLAG	;
DB_ATTRIB_TYPE	=KIND	; Function of attribute 0 = profile, 1 = accessory, 2 = gasket
DB_ATTRIB_COLOR	=KLEUR	; Index for finish, internal use
DB_ATTRIB_EQUAL	=EQUAL	; Position of profile in window 0= angled, 1= --, 2 = |
DB_ATTRIB_DEPNO	=AFDNR	; Department number of attribute
DB_ATTRIB_SEQNO	=VOLGNR	; Sequence number within department
DB_ATTRIB_SURFACE1	=OPPERVLAK1	; Is used for price calculation in function of profile surface
DB_ATTRIB_SURFACE2	=OPPERVLAK2	; Is used for price calculation, typically anodising
DB_ATTRIB_WEIGHT	=GEWICHT	; Weight of attribute
DB_ATTRIB_ITMPRICE	=ITMPRIJS	; Purchase price per unit
DB_ATTRIB_ITMOFFER	=ITMOFFER	; Selling price per unit
DB_ATTRIB_QTYPRICE	=QTYPRIJS	; Graduated purchase price
DB_ATTRIB_QTYOFFER	=QTYOFFER	; Graduated selling price
DB_ATTRIB_SETPRICE	=SETPRIJS	; Packing purchase price
DB_ATTRIB_SETOFFER	=SETOFFER	; Packing selling price
DB_ATTRIB_NO	=NR	; Number of items of attribute
DB_ATTRIB_STOCK	=STOCK	; Stock code

DB_ATTRIB_MINSIZE	=MINSIZE	; Graduated quantity
DB_ATTRIB_PACKSIZE	=PACKSIZE	; Items per pack
DB_ATTRIB_PRICE	=PRIJS	; Purchase price
DB_ATTRIB_OFFER	=OFFER	; Selling price for a length or number
DB_ATTRIB_TIME1	=TIJD1	; Value of time 1
DB_ATTRIB_TIME2	=TIJD2	; Value of time 2
DB_ATTIB_ARTICLE1	=ARTIKEL1	; Price block time 1
DB_ATTIB_ARTICLE2	=ARTIKEL2	; Price block time 2
DB_ATTRIB_INFO	=INFO	; 0 = Included, 1 = Charge, 2 = price, 3 = option, 4 = write, 5 = report
DB_ATTRIB_ARTICLE	=ARTIKEL	; Price block
DB_ATTRIB_REBATE	=KORTING	; Rebate on prices
DB_ATTRIB_LOSS	=Verlies	; Fixed loss
DB_ATTRIB_ERROR	=CFERROR	; = 1 when neg. packaching number (size<0)
DB_ATTRIB_PACK	=CFCEILRATE	; Number of packages to order rounded to next higher integer
DB_ATTRIB_CFLENGTH	=CFLENGTE	; length*number/1000
DB_ATTRIB_RATE	=CFCEILRATE	; Number of packages to order rounded to next higher integer
DB_ATTRIB_PEAKSIZE	=CFPEAKSIZE	;
DB_ATTRIB_PCKPEAKSIZE	=CFPCKPEAKSIZE	;
DB_ATTRIB_CFPRICE	=CFPRIJS	;
DB_ATTRIB_LABEL	=CFLABEL	; Projectcode*partcode
DB_ATTRIB_CFBATCH	=CFBATCH	; Batchcode*partcode
DB_ATTRIB_ATOM	=ATOM	;
DB_ATTRIB_HHI	=HHI	;
DB_ATTRIB_AUX	=AUX	;
DB_ATTRIB_BTYPE	=BATCHTYPE	; Batchtype for article

8.5. Block.qif

Resides in Common. Used for templates.

8.6. Client.qif

Resides in Common. Used for templates.

8.7. Combin.qif

Resides in Common. Used for templates.

8.8. Cost.qif

Resides in Common. Database contains all window finishes in the pool of projects.

DB_COST_BATCH	=BATCH	; Batchname, typically first project in batch or chosen by user
DB_COST_PROJECT	=PROJECT	; Project name in which part is used
DB_COST_ASSEMBLY	=GROEP	; Number of the group in which part is used
DB_COST_FRAME	=DEEL	; Frame in which part is used
DB_COST_BATCHNO	=BATCHNR	; Sequential number inside batch
DB_COST_BLOCK	=BLOK	; Price block
DB_COST_ARTICLE	=ARTIKEL	; Price code
DB_COST_DESC	=NAAM	; Description of price code
DB_COST_SERIE	=SERIE	; Order code
DB_COST_INFO	=INFO	; 0 = Included, 1 = Charge, 2 = price, 3 = option, 4 = write, 5 = report
DB_COST_CURRENCY	=MUNT	; Currency rate, related to price block
DB_COST_DISCOUNT	=KORTING	; Discount rate, related to price block
DB_COST_LOSS	=VERLIES	; Percent loss permitted in optimisation calculation, related to price block
DB_COST_PROFIT	=WINST	; Requested margin, related to price block
DB_COST_REBATE	=REBATE	; True or false, rebate to be calculated for article

		or not
DB_COST_PRICE	=PRIJS	; Prize information, coming from fixed price, price tables, other prices, …
DB_COST_TIME	=TIJD	; Cost related to time
DB_COST_RATIO	=RATIO	; Take coefficient into account or not
DB_COST_FACTOR	=FAKTOR	; Price coefficient finish if set in price block
DB_COST_CFGROSS	=CFBRUTO	; 1+loss/100
DB_COST_CFNETTO	=CFNETTO	; 1-rebate/100
DB_COST_CFRATIO	=CFRATIO	; Netto*factor
DB_COST_CFMARGIN	=CFMARGE	; Ratio + profit/100
DB_COST_CFTOTAL	=CFTOTAAL	; Ratio + margin
DB_COST_CFLOSS	=CFVERLIES	; 100*(gross/price-1)
DB_COST_CFDISCOUNT	=CFKORTING	; 100*(netto/gross-1)
DB_COST_CFFACTOR	=CFFAKTOR	; 100*(ratio/netto-1)
DB_COST_CFPROFIT	=CFWINST	; 100*(total/ratio-)

8.9. Extra.qif

Resides in Common.

DB_EXTRA_BATCHNO	=BATCHNR	; Order number in batch of project
DB_EXTRA_BATCH	=BATCH	; Batchname, typically first project in batch or chosen by user
DB_EXTRA_PROJECT	=PROJECT	; Project name in which extra is used
DB_EXTRA_ASSEMBLY	=GROEP	; Number of the group where extra is used
DB_EXTRA_FRAME	=DEEL	; Number of the part where extra is used
DB_EXTRA_OPENING	=OPEN	; Number of the opening where extra is used
DB_EXTRA_VENTPART	=VLDEEL	; Number of the ventpart where extra is used
DB_EXTRA_BC	=BC	; For future use with barcodes
DB_EXTRA_SUPPLIER	=LEVERANCIER	; Supplier code
DB_EXTRA_FINISH	=AFWERKING	; Window finish code
DB_EXTRA_DESC	=NAAM	; Description of extra part
DB_EXTRA_VARIETY	=VARIANT	; Color order code
DB_EXTRA_VARIETYDES	=VARIANTNAAM	; Color description
DB_EXTRA_VARIETYSERIE	=VARIANTSERIE	; Color code
DB_EXTRA_SERIE	=SERIE	;
DB_EXTRA_TYPE	=KIND	; Kind of Window finish 0 = none, 1 = left, 2=right, 3 = centre, 4 = bow, 5 = basket, 6 = round, 7 = width, 8 = height, 9 = side 10 = edge, 11 = perimeter, 12 = Insect screen, 13 = Napoleon hat, 14 = half perimeter, 15 = pieces
DB_EXTRA_DEPNO	=AFDNR	; Department number of extra
DB_EXTRA_SEQNO	=VOLGNR	; Sequence number within department
DB_EXTRA_WIDTH	=BREEDTE	; Depends on function in window finishes and corrections, here function put on width or surface related
DB_EXTRA_HEIGHT	=HOOGTE	; Depends on function in window finishes and corrections, here function put on width or surface related
DB_EXTRA_LENGTH	=LENGTE	; Depends on function in window finishes and corrections, here function put length related
DB_EXTRA_SURFACE	=OPPERVLAK	; Depends on function in window finishes and corrections, here function surface related
DB_EXTRA_ITMPRICE	=ITMPRIJS	; Purchase price/unit
DB_EXTRA_ITMOFFER	=ITMOFFER	; Selling price/unit
DB_EXTRA_MINPRIJS	=MINPRIJS	; Minimum purchase price
DB_EXTRA_MINOFFER	=MINOFFER	; Minimum selling price
DB_EXTRA_ INCPRIJS	=INCPRIJS	; Increment purchase price
DB_EXTRA_ INCOFFER	=INCOFFER	; Increment selling price
DB_EXTRA_NO	=NR	; Number of items
DB_EXTRA_PRICE	=PRIJS	; Purchase price

DB_EXTRA_OFFER	=OFFER	; Selling price
DB_EXTRA_TIME1	=TIJD1	; Value of time 1
DB_EXTRA_TIME2	=TIJD2	; Value of time 2
DB_EXTRA_ARTICLE1	=ARTIKEL1	; Price block time 1
DB_EXTRA_ARTICLE2	=ARTIKEL2	; Price block time 2
DB_EXTRA_INFO	=INFO	; 0 = Included, 1 = Charge, 2 = price, 3 = option, 4 = write, 5 = report
DB_EXTRA_REBATE	=KORTING	; Rebate from price block
DB_EXTRA_LOSS	=VERLIES	; Fixed loss
DB_EXTRA_LABEL	=CFLABEL	; Groupcode*partcode
DB_EXTRA_CFBATCH	=CFBATCH	; Batchcode*partcode
DB_EXTRA_CFPRICE	=CFPRIJS	; Selling or purchase price
DB_EXTRA_ATOM	=ATOM	;
DB_EXTRA_HHI	=HHI	;
DB_EXTRA_AUX	=AUX	;
DB_EXTRA_BTYPE	=BATCHTYPE	; Batchtype

8.10. Fill.qif

Resides in Common. and Wood

8.11. Finish.qif

Resides in Common. Used for templates.

8.12. Glass.qif

Resides in Common. Database contains all fillings in the pool of projects.

DB_GLASS_BATCHNO	=BATCHNR	; Order number in batch of project
DB_GLASS_BATCH	=BATCH	; Batchname, typically first project in batch or chosen by user
DB_GLASS_PROJECT	=PROJECT	; Project name in which filling is used
DB_GLASS_ASSEMBLY	=GROEP	; Number of the group where filling is used
DB_GLASS_FRAME	=DEEL	; Number of the part where filling is used
DB_GLASS_OPENING	=OPEN	; Number of the opening where filling is used
DB_GLASS_VENTPART	=VLDEEL	; Number of the ventpart where filling is used
DB_GLASS_VENTOPENING	=VLOPEN	; Number of the opening in the vent where filling is used
DB_GLASS_BC	=BC	; For future use with barcodes
DB_GLASS_SUPPLIER	=LEVERANCIER	; Supplier code
DB_GLASS_FILLING	=VULLING	; Filling code
DB_GLASS_DESC	=NAAM	; Description of filling
DB_GLASS_VARIETY	=VARIANT	; Color order code
DB_GLASS_VARIETYDESC	=VARIANTNAAM	; Color description
DB_GLASS_VARIETYSERIE	=VARIANTSERIE	; Color code
DB_GLASS_SERIE	=SERIE	;
DB_GLASS_TYPE	=KIND	; 0 = none, 1 = glass, 2 = cover profile, 3 = Panel, 4 = Flat, 5 = Diamond, 6 = Grill
DB_GLASS_DEPNO	=AFDNR	; Department number
DB_GLASS_SEQNO	=VOLGNR	; Sequence number within department
DB_GLASS_WIDTH	=BREEDTE	; Width of particular filling
DB_GLASS_HEIGHT	=HOOGTE	; Height of particular filling
DB_GLASS_MEASURE	=AFMETING	; Width*height/1000000 (actual dimensions)
DB_GLASS_PLACEMENT	=PLAATSING	; Price placement
DB_GLASS_SURFACE	=OPPERVLAK	; Width*height/1000000 (corrected to minimum dimensions and taken round off values into account)
DB_GLASS_WEIGHT	=GEWICHT	; Weight of the panel
DB_GLASS_ITMPRICE	=ITMPRIJS	; Purchase price/unit
DB_GLASS_ITMOFFER	=ITMOFFER	; Selling price/unit

```
DB_GLASS_UNTPRICE       =UNTPRIJS           ; Purchase price/m2
DB_GLASS_UNTOFFER       =UNTOFFER           ; Selling price/m2
DB_GLASS_NO             =NR                 ; Number of items
DB_GLASS_PRICE          =PRIJS              ; Purchase price
DB_GLASS_OFFER          =OFFER              ;
DB_GLASS_SLANTPRICE     =OFFER1             ;
DB_GLASS_ARCPRICE       =OFFER2             ;
DB_GLASS_TIME1          =TIJD1              ; Cost price 1
DB_GLASS_TIME2          =TIJD2              ; Cost price 2
DB_GLASS_INFO           =INFO               ; Way to calculate and present price in offer 0 =
DB_GLASS_ARTICLE        =ARTIKEL            ; Price block
DB_GLASS_REBATE         =KORTING            ; Rebate
DB_GLASS_LOSS           =VERLIES            ; Fixed loss
DB_GLASS_LABEL          =CFLABEL            ;
                              Groupcode*partcode:openingcode.ventcode.ventopeningcode
DB_GLASS_CFBATCH        =CFBATCH            ;
                              Groupcode*partcode:openingcode.ventcode.ventopeningcode
                              within batch
DB_GLASS_CFBATCH        =CFREMARK           ;
DB_GLASS_CFPRICE        =CFPRIJS            ; Price+placement
DB_GLASS_BATCHPC        =BATCHPC            ; niet in glas tabel
DB_GLASS_BATCHFC        =BATCHFC            ; niet in glas tabel
DB_GLASS_BATCHCC        =BATCHCC            ; niet in glas tabel
DB_GLASS_PROJECTPC      =PROJECTPC          ; niet in glas tabel
DB_GLASS_PROJECTFC      =PROJECTFC          ; niet in glas tabel
DB_GLASS_PROJECTCC      =PROJECTCC          ; niet in glas tabel
DB_GLASS_ASSEMBLYPC     =ASSEMBLYPC         ; niet in glas tabel
DB_GLASS_ASSEMBLYFC     =ASSEMBLYFC         ; niet in glas tabel
DB_GLASS_ASSEMBLYCC     =ASSEMBLYCC         ; niet in glas tabel
DB_GLASS_FRAMEPC        =FRAMEPC            ; niet in glas tabel
DB_GLASS_FRAMEFC        =FRAMEFC            ; niet in glas tabel
DB_GLASS_FRAMECC        =FRAMECC            ; niet in glas tabel
DB_GLASS_VENTPC         =VENTPC             ; niet in glas tabel
DB_GLASS_VENTFC         =VENTFC             ; niet in glas tabel
DB_GLASS_VENTCC         =VENTCC             ; niet in glas tabel
DB_GLASS_ATOM           =ATOM               ;
DB_GLASS_HHI            =HHI                ;
DB_GLASS_AUX            =AUX                ;
DB_GLASS_PROJECTATOM    =PROJECTATOM        ;
DB_GLASS_GRADE          =GRAAD              ;
DB_GLASS_BTYPE          =BATCHTYPE          ; Window batch type
```

8.13. Glazbead.qif

Resides in Common. Used for templates.

8.14. Hintinfo.qif

Resides in Common.

8.15. Model.qif

Resides in Common.

8.16. Mop.qif

Resides in Common. Used for templates.

8.17. Mops.qif

Resides in Common. Database contains all fillings in the pool of projects.

DB_MOPS_ID	=ID	; Sequence number within list of Macining OPerationS
DB_MOPS_ENABLED	=ENABLED	; Is operation enabled or not
DB_MOPS_PROJECT	=PROJECT	; Project name in which operation is activated
DB_MOPS_ASSEMBLY	=GROEP	; Number of the group where for operation is activated
DB_MOPS_FRAME	=DEEL	; Number of the part where for operation is activated
DB_MOPS_CODE	=MOP	; Machine operating code for use in setup
DB_MOPS_DESC	=NAAM	; Machine operating code description
DB_MOPS_KIND	=FUNCTIE	;
DB_MOPS_ROUTINE	=ROUTINE	;
DB_MOPS_POSITION	=POSITION	; Side of machiningoperation?
DB_MOPS_OFFSET	=OFFSET	; Position of machining operation from start profile
DB_MOPS_CORRECT	=CORRECT	; Correction on position of machining operation from start profile
DB_MOPS_LPNT	=LPNT	; JoPPS internal profile length, connected to the model build up
DB_MOPS_LMAX	=LMAX	; Overmeasured length of piece
DB_MOPS_LMIN	=LMIN	; Piece length
DB_MOPS_LABS	=LABS	; Lmax corrected with shorten and round off values from profile parameters
DB_MOPS_LOPT	=LOPT	; Bar length used to optimise piece
DB_MOPS_CMAX1	=CMAX1	; Correction at start of profile to go from LPNT to LMAX
DB_MOPS_CMAX2	=CMAX2	; Correction at end of profile to go from LPNT to LMAX
DB_MOPS_CMIN1	=CMIN1	; Correction at start of profile to go from LMAX to LMIN
DB_MOPS_CMIN2	=CMIN2	; Correction at end of profile to go from LMAX to LMIN
DB_MOPS_CABS1	=CABS1	; Correction at start of profile to go from LABS to LMAX
DB_MOPS_CABS2	=CABS2	;Correction at end of profile to go from LABS to LMAX
DB_MOPS_DEPNO	=AFDNR	; Department number
DB_MOPS_SEQNO	=VOLGNR	; Sequence number within department
DB_MOPS_TIME1	=TIJD1	; Cost price 1
DB_MOPS_TIME2	=TIJD2	; Cost price 2
DB_MOPS_ARTICLE1	=ARTIKEL1	; Price block 1
DB_MOPS_ARTICLE2	=ARTIKEL2	; Price block 2
DB_MOPS_A	=A	; First parameter to go from Jopps machining instruction to MCMan machine code, numeric value
DB_MOPS_B	=B	; Second parameter to go from Jopps machining instruction to MCMan machine code, numeric value
DB_MOPS_C	=C	; Third parameter to go from Jopps machining instruction to MCMan machine code, numeric value
DB_MOPS_D	=D	; Fourth parameter to go from Jopps machining instruction to MCMan machine code, numeric value
DB_MOPS_E	=E	; Fifth parameter to go from Jopps machining instruction to MCMan machine code, numeric value
DB_MOPS_F	=F	; Sixth parameter to go from Jopps machining

		instruction to MCMan machine code, numeric value
DB_MOPS_G	=G	; Seventh parameter to go from Jopps machining instruction to MCMan machine code, numeric value
DB_MOPS_H	=H	; Eight parameter to go from Jopps machining instruction to MCMan machine code, numeric value
DB_MOPS_S	=S	; String parameter to go from Jopps machining instruction to MCMan machine code
DB_MOPS_ATOM	=ATOM	;
DB_MOPS_COMMENT	=COMMENT	;
DB_MOPS_BTYPE	=BATCHTYPE	;

8.18. Norm.qif

Resides in Common. Used for templates.

8.19. Offer.qif

Resides in Common.

DB_OFFER_PROJECT	=PROJECT	; Name of project
DB_OFFER_ASSEMBLY	=GROEP	; Assembly number within project
DB_OFFER_FRAME	=DEEL	; Part number within assembly
DB_OFFER_LEVEL	=LVL	; Internal use only
DB_OFFER_INDEX	=NDX	;
DB_OFFER_INFO	=INFO	; 0 = Included, 1 = Charge, 2 = price, 3 = option, 4 = write, 5 = report
DB_OFFER_NO	=NR	; Number of items
DB_OFFER_DESC	=NAAM	; Description of, assembly, part, window finishing
DB_OFFER_SYSTEM	=REEKS	; System code of item
DB_OFFER_SYSTEMDESC	=REEKSNAAM	; Description of item
DB_OFFER_VARIETY	=VARIANT	; Color order code
DB_OFFER_VARIETYDESC	=VARIANTNAAM	; Color description
DB_OFFER_WIDTH	=BREEDTE	; Width relating to object
DB_OFFER_HEIGHT	=HOOGTE	; Height relating to object
DB_OFFER_LENGTH	=LENGTE	; Length relating to object
DB_OFFER_FACTOR	=FAKTOR	; Number of parts
DB_OFFER_ARTICLE	=ARTIKEL	; Price block
DB_OFFER_REBATE	=REBATE	; Rebate on prices
DB_OFFER_REFUND	=REFUND	; Percentage loss
DB_OFFER_PRICE	=PRIJS	; Price of object
DB_OFFER_TIME	=TIJD	; Time related to object
DB_OFFER_LOSS	=LOSS	; Loss percentage coming from price block
DB_OFFER_CFPIECES	=CFSTUKS	; Value depends on level and number, internal use
DB_OFFER_CFAFFPIECES	=CFAFFSTUKS	; As above, but for confirmation
DB_OFFER_CFPRICE	=CFPRIJS	;
DB_OFFER_SUBLEVEL	=CFLVLCOUNT	;
DB_OFFER_ASSITEM	=CFGROEPITEM	;

8.20. Param.qif

Resides in Common. Used for templates.

8.21. Part.qif

Resides in Common.

```
DB_PART_PROJECT          =PROJECT          ; Project name in which element is used
DB_PART_ASSEMBLY         =GROEP            ; Number of the group where element is used
DB_PART_ FRAME           =DEEL             ; Number of the part where element is used
DB_PART_OPENING          =OPEN             ; Number of the opening where element is used
DB_PART_VENTPART         =VLDEEL           ; Number of the ventpart where element is
                                             used
DB_PART_VENTOPENING      =VLOPEN           ; Number of the ventpart opening where
                                             element is used
DB_PART_ID               =ID               ;
DB_PART_KIND             =KIND             ;
DB_PART_FLAG             =FLAG             ;
DB_PART_FACTOR           =FAKTOR           ;
DB_PART_CODE             =CODE             ; Element code
DB_PART_DESC             =NAAM             ; Element description
DB_PART_SYSTEM           =REEKS            ; System code of element
DB_PART_SYSTEMDESC       =REEKSNAAM        ; System description
DB_PART_VARIETY          =VARIANT          ; Order code
DB_PART_VARIETYDESC      =VARIANTNAAM      ; Order code description
DB_PART_WIDTH            =WIDTH            ; Window width (also vent and filling)
DB_PART_HEIGTH           =HEIGHT           ; Window height (also vent and filling)
DB_PART_ACCESSWIDTH      =WIDTHEX          ; Window width with vent correction
DB_PART_ACCESSHEIGHT     =HEIGHTEX         ; Window height with vent correction
DB_PART_GASKETWIDTH1     =GASKETW1         ;
DB_PART_GASKETHEIGHT1    =GASKETH1         ;
DB_PART_GASKETWIDTH2     =GASKETW2         ;
DB_PART_GASKETHEIGHT2    =GASKETH2         ;
DB_PART_GASKETWIDTH3     =GASKETW3         ;
DB_PART_GASKETHEIGHT3    =GASKETH3         ;
DB_PART_TOTALWEIGHT      =TOTALWEIGHT      ; Total weight of window, not for other elements
DB_PART_FRAMEWEIGHT      =FRAMEWEIGHT      ; Frame weight of window, not for other elements
DB_PART_GLASSWEIGHT      =GLASSWEIGHT      ; Glass weight of window, not for other elements
DB_PART_SURFACE          =SURFACE          ; Window surface (also vent and filling)
DB_PART_PERIMETER        =PERIMETER        ; Window perimeter (also vent)
DB_PART_LENGTH           =LENGTH           ;
DB_PART_CFWIDTH          =CFWIDTH          ;
DB_PART_CFHEIGHT         =CFHEIGHT         ;
DB_PART_CFEDGE           =CFEDGE           ;
DB_PART_CFSIZE           =CFSIZE           ;
DB_PART_HANDLEPOS        =HANDLE           ; Position handle on vent
DB_PART_BATCHREF         =BATCHREF         ;
DB_PART_BATCHCNT         =BATCHCNT         ;
DB_PART_PROJECTREF       =PROJECTREF       ;
DB_PART_PROJECTCNT       =PROJECTCNT       ;
DB_PART_ASSEMBLYREF      =GROEPREF         ;
DB_PART_ASSEMBLYCNT      =GROEPCNT         ;
DB_PART_FRAMEREF         =FRAMEREF         ;
DB_PART_FRAMECNT         =FRAMECNT         ;
DB_PART_VENTREF          =VENTREF          ;
DB_PART_VENTCNT          =VENTCNT          ;
DB_PART_ATOM             =ATOM             ;
DB_PART_PROJECTATOM      =PROJECTATOM      ;
DB_PART_CHILDCNT         =CHILDCNT         ;
DB_PART_MOPCNT           =MOPCNT           ;
DB_PART_REFID            =CFLABEL          ;
```

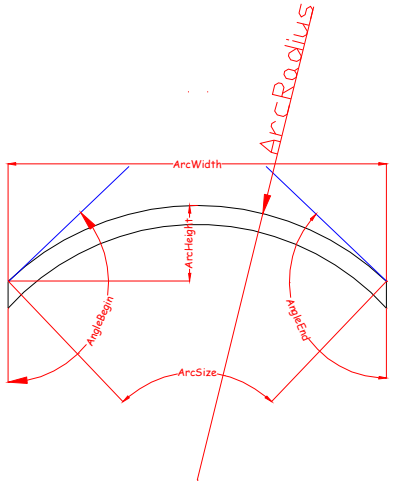
8.22. Piece.qif

Resides in Alu, Common., PVC and Wood. Table contains all pieces (of profiles) in the pool of projects.

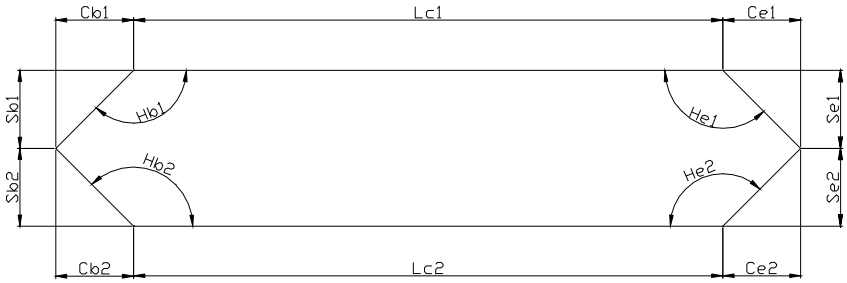
DB_PIECE_BATCHNO	=BATCHNR	; Sequential number of piece in pool
DB_PIECE_BATCH	=BATCH	; Name of the batch run, mostly first project name in pool
DB_PIECE_PRODUCT	=PRODUKT	; Order code of piece
DB_PIECE_VARIETYDESC	=VARIANTNAAM	; Variant description of part
DB_PIECE_VARIETY	=VARIANT	; Code of variant
DB_PIECE_LABS	=LABS	; Absolute length of piece
DB_PIECE_PROJECT	=PROJECT	; Project in pool to which piece is related
DB_PIECE_ASSEMBLY	=GROEP	; Group in which the piece exists
DB_PIECE_FRAME	=DEEL	; Frame in which the piece exists
DB_PIECE_OPENING	=OPEN	; Opening to which piece is related
DB_PIECE_VENTPART	=VLDEEL	; Ventpart to which is piece is related
DB_PIECE_BC	= BC	; For future use with barcodes
DB_PIECE_SUPPLIER	=LEVERANCIER	; Code of the supplier
DB_PIECE_SYSTEM	=REEKS	; System to which piece belongs
DB_PIECE_PROFILE	=PROFIEL	; Profile code
DB_PIECE_PROFILEDESC	=PROFIELNAAM	; Description of profile from which piece is cut
DB_PIECE_VARIETYSERIE	=VARIANTSERIE	; Color order code
DB_PIECE_SERIE	=SERIE	; Order code of profile
DB_PIECE_PRODUCTDESC	=PRODUKTNAAM	; Profile description
DB_PIECE_LINK	=LINK	; Link to other profile
DB_PIECE_PART	=PART	; Number of reinforcements in piece (0,1,2,3)
DB_PIECE_PARTCODE1	=PARTCODE1	; Partcode of first reinforcement
DB_PIECE_PARTLABS1	=PARTLABS1	; Absolute length first reinforcement
DB_PIECE_PARTLBGN1	=PARTLBGN1	; Distance from begin piece to begin first reinforcement
DB_PIECE_PARTLEND1	=PARTLEND1	; Distance from end piece to end first reinforcement
DB_PIECE_PARTCODE2	=PARTCODE2	; Partcode of second reinforcement
DB_PIECE_PARTLABS2	=PARTLABS2	; Absolute length second reinforcement
DB_PIECE_PARTLBGN2	=PARTLBGN2	; Distance from begin piece to begin second reinforcement
DB_PIECE_PARTLEND2	=PARTLEND2	; Distance from end piece to end second reinforcement
DB_PIECE_PARTCODE3	=PARTCODE3	; Partcode of third reinforcement
DB_PIECE_PARTLABS3	=PARTLABS3	; Absolute length third reinforcement
DB_PIECE_PARTLBGN3	=PARTLBGN3	; Distance from begin piece to begin third reinforcement
DB_PIECE_PARTLEND3	=PARTLEND3	; Distance from end piece to end third reinforcement
DB_PIECE_COLOR	=KLEUR	; Index for finish, internal use
DB_PIECE_PIECENO	=LABEL	; Show label or not 0 = not, 2 = show label
DB_PIECE_EQUAL	=EQUAL	; Position of profile in window 0= angled, 1= --, 2 = I
DB_PIECE_PLACE	=PLACE	; 5=links, 6=rechts, 3=onder, 4=boven, 0=schuin, 1 horizontaal, 2 verticaal
DB_PIECE_LABEL	=ETIKET	; 0= no label, 1= label, -1= already evaluated
DB_PIECE_FACTOR	=FAKTOR	;
DB_PIECE_CHAR	=LETTER	; Letter or code to be printed with editor or on documents, visualised on the drawing
DB_PIECE_PROGRAM	=PROGRAM	; Program code specified with profile
DB_PIECE_SYSTEM1	=REEKS1	; System to which profile belongs coming to start of profile
DB_PIECE_PROFILE1	=PROFIEL1	; Profile code
DB_PIECE_SYSTEM2	=REEKS2	; System to which profile belongs coming to start of profile
DB_PIECE_PROFILE2	=PROFIEL2	; Profile code

DB_PIECE_DEPNO	=AFDNR	; Department number
DB_PIECE_SEQNO	=VOLGNR	; Sequence number within department
DB_PIECE_WIDTH	=BREEDTE	; Width of piece
DB_PIECE_OFFSET1	=OFFSET1	; Offset side one coming from profile parameters
DB_PIECE_OFFSET2	=OFFSET2	; Offset side two coming from profile parameters
DB_PIECE_THICKNESS	=DIKTE	; Thickness of piece
DB_PIECE_W1	=W1	; Parameter W1 from profile parameters
DB_PIECE_W2	=W2	; Parameter W2 from profile parameters
DB_PIECE_W3	=W3	; Parameter W3 from profile parameters
DB_PIECE_W4	=W4	; Parameter W4 from profile parameters
DB_PIECE_W5	=W5	; Parameter W5 from profile parameters
DB_PIECE_W6	=W6	; Parameter W6 from profile parameters
DB_PIECE_W7	=W7	; Parameter W7 from profile parameters
DB_PIECE_W7	=W8	; Parameter W8 from profile parameters
DB_PIECE_W9	=W9	; Parameter W9 from profile parameters
DB_PIECE_W10	=W10	; Parameter W10 from profile parameters
DB_PIECE_X1	=X1	; Parameter X1 from profile parameters
DB_PIECE_X2	=X2	; Parameter X2 from profile parameters
DB_PIECE_X3	=X3	; Parameter X3 from profile parameters
DB_PIECE_X4	=X4	; Parameter X4 from profile parameters
DB_PIECE_X5	=X5	; Parameter X5 from profile parameters
DB_PIECE_X6	=X6	; Parameter X6 from profile parameters
DB_PIECE_X7	=X7	; Parameter X7 from profile parameters
DB_PIECE_X7	=X8	; Parameter X8 from profile parameters
DB_PIECE_X9	=X9	; Parameter X9 from profile parameters
DB_PIECE_X10	=X10	; Parameter X10 from profile parameters
DB_PIECE_Y1	=Y1	; Parameter Y1 from profile parameters
DB_PIECE_Y2	=Y2	; Parameter Y2 from profile parameters
DB_PIECE_Y3	=Y3	; Parameter Y3 from profile parameters
DB_PIECE_Y4	=Y4	; Parameter Y4 from profile parameters
DB_PIECE_Y5	=Y5	; Parameter Y5 from profile parameters
DB_PIECE_Y6	=Y6	; Parameter Y6 from profile parameters
DB_PIECE_Y7	=Y7	; Parameter Y7 from profile parameters
DB_PIECE_Y7	=Y8	; Parameter Y8 from profile parameters
DB_PIECE_Y9	=Y9	; Parameter Y9 from profile parameters
DB_PIECE_Y10	=Y10	; Parameter Y10 from profile parameters
DB_PIECE_Z1	=Z1	; Parameter Z1 from profile parameters
DB_PIECE_Z2	=Z2	; Parameter Z2 from profile parameters
DB_PIECE_Z3	=Z3	; Parameter Z3 from profile parameters
DB_PIECE_Z4	=Z4	; Parameter Z4 from profile parameters
DB_PIECE_Z5	=Z5	; Parameter Z5 from profile parameters
DB_PIECE_Z6	=Z6	; Parameter Z6 from profile parameters
DB_PIECE_Z7	=Z7	; Parameter Z7 from profile parameters
DB_PIECE_Z7	=Z8	; Parameter Z8 from profile parameters
DB_PIECE_Z9	=Z9	; Parameter Z9 from profile parameters
DB_PIECE_Z10	=Z10	; Parameter Z10 from profile parameters
DB_PIECE_SURFACE1	=OPPERVLAK1	; Surface of piece, circumference to distance (Standard)
DB_PIECE_SURFACE2	=OPPERVLAK2	; Surface of piece, circumference to distance (Anodising)
DB_PIECE_WEIGHT	=GEWICHT	; Weight of piece to unit length
DB_PIECE_VOLUME	=VOLUME	;Volume of piece to unit length
DB_PIECE_LPNT	=LPNT	; Length, value from editor (start point-end point)
DB_PIECE_LMAX	=LMAX	; Length of piece relative to outside of frame
DB_PIECE_LMIN	=LMIN	; Length of piece relative to inside
DB_PIECE_LOPT	=LOPT	; Length of bar to which piece is related after optimise
DB_PIECE_LPEN	=LPEN	; Standard penlength for wood
DB_PIECE_LEXT	=LEXT	; Standard extention profile when stop
DB_PIECE_CORRECTB	=CB	; Correction value begin piece equal to width of

		the arriving piece
DB_PIECE_CORRECTE	=CE	; Correction value end piece equal to width of the arriving piece
DB_PIECE_ANGLEB	=HB	; Cutting angle begin of piece
DB_PIECE_ANGLEE	=HE	; Cutting angle end of piece
DB_PIECE_VISIBLEB	=SB	, For Wood, Pen visible start profile
DB_PIECE_VISIBLEE	=SE	, For Wood, Pen visible end profile
DB_PIECE_STOPB	=VB	; For Wood, Pen or no Pen start profile
DB_PIECE_STOPE	=VE	; For Wood, Pen or no Pen end profile
DB_PIECE_ARCHEIGHT	=HARC	; Height of arc
DB_PIECE_ARCWIDTH	=WARC	; Width of arc
DB_PIECE_ARCRADIUS	=RARC	; Radius of arc
DB_PIECE_ARCSIZE	=SARC	;



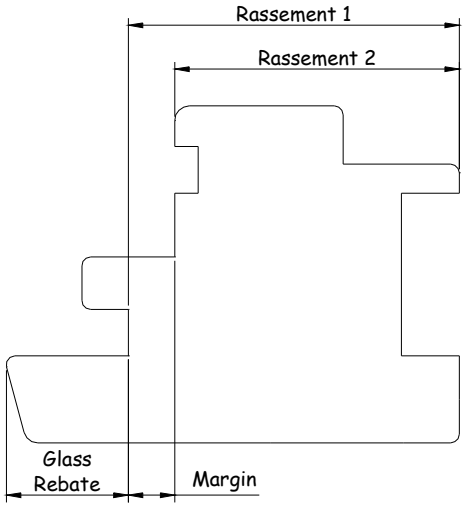
DB_PIECE_LCUT0	=LC0	;



DB_PIECE_LCUT1	=LC1	; See drawing above
DB_PIECE_LCUT2	=LC2	; See drawing above
DB_PIECE_ANGLEB1	=HB1	; See drawing above
DB_PIECE_ANGLEE1	=HE1	; See drawing above
DB_PIECE_CORRECTB1	=CB1	; See drawing above
DB_PIECE_CORRECTE1	=CE1	; See drawing above
DB_PIECE_SIZEB1	=SB1	; See drawing above
DB_PIECE_SIZEE1	=SE1	; See drawing above
DB_PIECE_ANGLEB2	=HB2	; See drawing above
DB_PIECE_ANGLEE2	=HE2	; See drawing above
DB_PIECE_CORRECTB2	=CB2	; See drawing above
DB_PIECE_CORRECTE2	=CE2	; See drawing above

DB_PIECE_SIZEB2	=SB2	; See drawing above
DB_PIECE_SIZEE2	=SE2	; See drawing above
DB_PIECE_COST1	=KOST1	; Cost of item per length
DB_PIECE_COST2	=KOST2	; Cost of item per weight
DB_PIECE_COST3	=KOST3	; Cost of item per surface
DB_PIECE_COST4	=KOST4	;
DB_PIECE_COST5	=KOST5	;
DB_PIECE_COST6	=KOST6	;
DB_PIECE_COST7	=KOST7	;
DB_PIECE_COST8	=KOST8	;
DB_PIECE_MEASURE1	=MAAT1	; Minimum length
DB_PIECE_MEASURE2	=MAAT2	; Minimum weight
DB_PIECE_MEASURE3	=MAAT3	; Minimum surface
DB_PIECE_MEASURE4	=MAAT4	;
DB_PIECE_MEASURE5	=MAAT5	;
DB_PIECE_MEASURE6	=MAAT6	;
DB_PIECE_MEASURE7	=MAAT7	;
DB_PIECE_MEASURE8	=MAAT8	;
DB_PIECE_NO	=NR	; number of times profile has to be used, specified in editor by user
DB_PIECE_STOCK	=STOCK	; Stock code
DB_PIECE_MINSIZE	=MINSIZE	; Graduated quantity
DB_PIECE_PACKSIZE	=PACKSIZE	; Items per pack
DB_PIECE_PRICE	=PRIJS	; Purchase price
DB_PIECE_OFFER	=OFFER	; Selling price
DB_PIECE_TIME1	=TIJD1	; Cost time 1
DB_PIECE_TIME2	=TIJD2	; Cost time 2
DB_PIECE_ARTICLE1	=ARTIKEL1	; Price block time 1
DB_PIECE_ARTICLE2	=ARTIKEL2	; Price block time 2
DB_PIECE_INFO	=INFO	; 0 = Included, 1 = Charge, 2 = price, 3 = option, 4 = write, 5 = report
DB_PIECE_ARTICLE	=ARTIKEL	; Price block
DB_PIECE_REBATE	=KORTING	; Rebate
DB_PIECE_LOSS	= VERLIES	; Fixed loss
DB_PIECE_OPTREPNO	=REPNR	; Repetition number, number of times the same combination of pieces resulting to a bar occurs
DB_PIECE_OPTITEMNO	=ITEMNR	; Sequence number of the bar within bars of same profile code, restarts for every new barcode or colour
DB_PIECE_OPTSEQNO	=SEQNR	; Sequence number within bar
DB_PIECE_OPTSAWNO	=ZAAGNR	; Sequence number of the bar within bars of same profile code, goes up with the same value as the number of bars taken together e.g. 1 – 3 – 5
DB_PIECE_BARSPERCUT	=BPC	; Number of bars taken together
DB_PIECE_LST_NO	=L_NR	;
DB_PIECE_LST_REPNO	=L_REPNR	;
DB_PIECE_LST_ITEMNO	=L_ITEMNR	;
DB_PIECE_LST_SEGNO	=L_SEQNR	;
DB_PIECE_CARRIERNO	=KARNR	; Number of carrier, used in batchmode
DB_PIECE_CABINNO	=VAKNR	; Cabin number on carrier, e.g. 1 for first window frame profiles, 2 first window vent profiles
DB_PIECE_CFCHAR	=CFLETTER	; how profile is positioned in frame | = vertical, - = horizontal
DB_PIECE_ID	=CFLABEL	; Text to be printed on label
DB_PIECE_ID_PRJ	=CFLABELPRJ	; Text to be printed on label if in batch
DB_PIECE_NUMBER	=CFNUMMER	; Identical to Label, except when rest piece then = 0
DB_PIECE_CFLENGTH	=CFLENGTE	; Lmax*number/1000 if selling price, else Labs*number/1000
DB_PIECE_CFPRICE	=CFPRIJS	; Selling or purchase price

DB_PIECE_CFREMLENGTH	=CFRESTLENGTE	; Lmax*Repnumber/1000 if selling price, else Labs*Repnumber/1000
DB_PIECE_BATCHREF	=BATCHREF	;
DB_PIECE_BATCHCNT	=BATCHCNT	;
DB_PIECE_PROJECTREF	=PROJECTREF	;
DB_PIECE_PROJECTCNT	=PROJECTCNT	;
DB_PIECE_ASSEMBLYREF	=ASSEMBLYREF	;
DB_PIECE_ASSEMBLYCNT	=ASSEMBLYCNT	;
DB_PIECE_FRAMEREF	=FRAMEREF	;
DB_PIECE_FRAMECNT	=FRAMECNT	;
DB_PIECE_VENTREF	=VENTREF	;
DB_PIECE_VENTCNT	=VENTCNT	;
DB_PIECE_ATOM	=ATOM	;
DB_PIECE_HHI	=HHI	;
DB_PIECE_AUX	=AUX	;
DB_PIECE_PARENTATOM	=PARENTATOM	;
DB_PIECE_PROJECTATOM	=PROJECTATOM	;
DB_PIECE_COUNT	=CNT	;
DB_PIECE_REVERTED	=REVERTED	; Profile is reverted related to normal profile direction
DB_PIECE_CMAX1	=CMAX1	;
DB_PIECE_CMAX2	=CMAX2	;
DB_PIECE_CMIN1	=CMIN1	;
DB_PIECE_CMIN2	=CMIN2	;
DB_PIECE_CABS1	=CABS1	;
DB_PIECE_CABS2	=CABS2	;
DB_PIECE_RB0	=RB0	;
DB_PIECE_RE0	=RE0	;



DB_PIECE_R1B1	=R1B1	; Rassement 1 Begin profile side 1, see drawing
DB_PIECE_R2B1	=R2B1	; Rassement 2 Begin profile side 1, see drawing
DB_PIECE_R1E1	=R1E1	; Rassement 1 End profile side 1, see drawing
DB_PIECE_R2E1	=R2E1	; Rassement 2 End profile side 1, see drawing
DB_PIECE_R1B2	=R1B2	; Rassement 1 Begin profile side 2, see drawing
DB_PIECE_R2B2	=R2B2	; Rassement 2 Begin profile side 2, see drawing
DB_PIECE_R1E2	=R1E2	; Rassement 1 End profile side 2, see drawing
DB_PIECE_R2E2	=R2E2	; Rassement 2 End profile side 2, see drawing
DB_PIECE_COMMENT	=COMMENT	;
DB_PIECE_MOPCNT	=MOPCNT	;
DB_PIECE_RACKID	=RACKID	; Rack identification in which profile is placed, is given by user in sertup

```
 DB_PIECE_RACKNO          =RACKNO          ; Rack sequence number which profile is placed,
for each type of rack
 DB_PIECE_RACKX           =RACKX           ; Rack identification in which profile is placed in
X-direction, takes alphanumeric into account
 DB_PIECE_RACKY           =RACKY           ; Rack identification in which profile is placed in
Y-direction, takes alphanumeric into account
 DB_PIECE_RACKXI          =RACKXI          ; Rack identification in which profile is placed in
X-direction, integer value
 DB_PIECE_RACKYI          =RACKYI          ; Rack identification in which profile is placed in
Y-direction, integer value
 DB_PIECE_RACK            =RACK            ; Combination code to identify rack.
Combination of RackNO, RackID,RackX,RackY
 DB_PIECE_BTYPE           =BATCHTYPE       ;  Batchtype of chassis
```

8.23. Product.qif

Resides in Common. Used for templates.

8.24. Profile.qif

Resides in Common. Information is recalled from inside layout files from cutting list. Is the table describing the profile parameters.

```
 DB_PROFILE_CODE_SERIE          =CODE_REEKS      ; Series code, first part unique code
 DB_PROFILE_CODE_PROFILE        =CODE_PROFIEL    ; Profile code, second part unique code
 DB_PROFILE_DESC                =%REC_DESC%      ; Profile description
 DB_PROFILE_OWNER               =OWNER           ; JoPPS key of dongle that last changed
                                                   the record
 DB_PROFILE_MODIFIED            =MODIFIED        ; Time of modifivation
 DB_PROFILE_DXFCAD              =DXF_CAD         ; DXF reference for cross section
 DB_PROFILE_DXFDRW              =DXF_DRW         ; DXF reference for head on view
 DB_PROFILE_UDS                 =UDS             ; Show label (profile code) or not
                                                   (for cross sections)
 DB_PROFILE_DRWWIDTH            =DRW_WIDTH       ; Positioning in X of label (profile code)
 DB_PROFILE_DRWHEIGHT           =DRW_HEIGHT      ; Positioning in Y of label (profile code)
 DB_PROFILE_LINK                =LINK            ; For creating cross sections
 DB_PROFILE_FUNCTION            =FUNCTIE         ; 0 = profile, 1 = guide, 2 = widen,
                                                   3 =  sun,  4 = centre, 5 = circle,
                                                   6 = sun centre, 7 = sun glass,
                                                   8 = centre glass, 9 = compartment,
                                                   10 = cross, 11 = on glass, 12 = in glass,
                                                   13 = none, 14 = weather bar profile,
                                                   15 = sill, 16 = joining profile,
                                                   17 = Ventilation, 18 = reinforcement,
                                                   19 = cross, 20 = glazing bead
 DB_PROFILE_SUPPLIER            =LEVERANCIER     ; Supplier code
 DB_PROFILE_PRODUCT             =PRODUKT         ; Order code, related to profile product
                                                   code
 DB_PROFILE_WIDTH               =BREEDTE         ; Profile width, cross section
 DB_PROFILE_THICKNESS           =DIKTE           ; Profile height, cross section
 DB_PROFILE_GEOMETRY            =GEOMETRIE       ; Array of 4x10 for general coefficients
                                                   in tab geometry
 DB_PROFILE_WEIGHT              =GEWICHT         ; Profile weight/length
 DB_PROFILE_MOMENT1             =MOMENT1         ; Moment of inertia Ix
 DB_PROFILE_MOMENT2             =MOMENT2         ; Moment of inertia Iy
 DB_PROFILE_SURFACE1            =OPPERVLAK1      ; Developed surface of profile, for
                                                   anodising for example
 DB_PROFILE_SURFACE2            =OPPERVLAK2      ; Developed surface of profile, for
                                                   anodising for example
 DB_PROFILE_MINUEND             =AFTREKMAAT      ; Value to be subtracted from profile
                                                    length for each end
```

DB_PROFILE_SHORTEN	=AFKORTMAAT	; Value to be added to profile length for each end, typically used in wood to leave some material for cutting slots
DB_PROFILE_ROUNDOFF	=AFRONDMAAT	; incremental value to round profile length to
DB_PROFILE_REBATE	=GLASSLAG	; Distance glass goes into profile
DB_PROFILE_REBATE1	=GLASSLAG1	; Distance glass goes into profile side 1
DB_PROFILE_REBATE2	=GLASSLAG2	; Distance glass goes into profile side 2
DB_PROFILE_MARGIN	=SPELING	; Correction value on profile length for profile that butts onto this profile
DB_PROFILE_MARGIN1	=SPELING1	; Correction value on profile length for profile that butts onto this profile side 1
DB_PROFILE_MARGIN2	=SPELING2	; Correction value on profile length for profile that butts onto this profile side 2
DB_PROFILE_OFFSET	=OFFSET	; Correction value for going from vent Sizes to accessorie sizes
DB_PROFILE_ARTICLE	=ARTIKEL	; Price block code
DB_PROFILE_ARTICLE1	=ARTIKEL1	; Price block code for time 1
DB_PROFILE_ARTICLE2	=ARTIKEL2	; Price block code for time 2
DB_PROFILE_TIME1	=TIJD1	; Value for time 1 for this profile
DB_PROFILE_TIME2	=TIJD2	; Value for time 2 for this profile
DB_PROFILE_DEPNUM	=AFDNR	; Department number used for organising profiles
DB_PROFILE_SEQNUM	=VOLGNR	; Sequence number for extra organising within department
DB_PROFILE_MINLEN	=MINLEN	; Minimum length for profile for price
DB_PROFILE_INCLEN	=INCLEN	; Incremental length for pricing
DB_PROFILE_MINPRICE	=MINPRIJS	; Minimum price for minimum length
DB_PROFILE_INCPRICE	=INCPRIJS	; Incremental price with incremental length
DB_PROFILE_CHAR	=LETTER	; Character that can be output to screen in editor or to the report in results
DB_PROFILE_PROGRAM	=PROGRAM	; 4 position program code
DB_PROFILE_GLAZINGBEAD	=GLASLAT	; Code glazing bead set related to this profile
DB_PROFILE_GLAZINGBEAD1	=GLASLAT1	; Code glazing bead set related to this profile for side 1
DB_PROFILE_GLAZINGBEAD2	=GLASLAT2	; Code glazing bead set related to this profile for side 2
DB_PROFILE_STIFFENER	=VERSTERK	; Code for reinforcement set related to this profile
DB_PROFILE_COMBIN_SYSTEM	=COMBIN_REEKS	; Series of profile that combines as this profile
DB_PROFILE_COMBIN_PROFILE	=COMBIN_PROFIEL	; Series of profile that combines as this profile
DB_PROFILE_OVERMEASURE	=INROLMAAT	; Measure to be added to each extreme to be able to make a bend
DB_PROFILE_MINRADIUS	=MINSTRAAL	; Minimum banding radius
DB_PROFILE_PENSIZE	=PENMAAT	; Standard length of pen (only for wood)
DB_PROFILE_MORTISELENGTH	=SLISMAAT	; Depth of pen when profile junction is of type stop (only for wood)
DB_PROFILE_INTERNALLENGTH	=ZIELBREEDTE	; Extra width side 1 of profile (only for wood)
DB_PROFILE_INTERNALMEASURE	=ZIELMAAT	; Extra width side 2 of profile (only for wood)
DB_PROFILE_RESIZE	=RESIZE	; Is user allowed to change width of profile in editor
DB_PROFILE_GASKET	=DICHTING	; Blobfield relating to gaskets defined

		with the profile
DB_PROFILE_ADDON	=ADDON	; Blobfield relating to extra profiles defined with the profile
DB_PROFILE_CRIMP	=PERSEN	; Blobfield relating to the information regarding crimp junctions
DB_PROFILE_SCREW	=SCHROEVEN	; Blobfield relating to the information regarding screw junctions
DB_PROFILE_CLAMP	=KLEMMEN	; Blobfield relating to the information regarding clamp junctions
DB_PROFILE_ACC	=BESLAG	; Blobfield relating to accessories defined with the profile
DB_PROFILE_ACTION	=ACTIE	; Blobfield of information regarding machining operations
DB_PROFILE_MASK	=MASK	;
DB_PROFILE_FILTER	=FLTR	;
DB_PROFILE_HELPTOPIC	=HELP_TOPIC	;
DB_PROFILE_MEMO	=MEMO	; Memo field

8.25. Serie.qif

Resides in Common. Used for templates.

8.26. Standard.qif

Resides in Common. Used for templates.

8.27. Stiff.qif

Resides in Common. Used for templates.

8.28. Tariff.qif

Resides in Common. Used for templates.

8.26. Text.qif

Resides in Common.

DB_TEXT_CODE	=CODE	; Project code, or customer/supplier code
DB_TEXT_NAME	=TEXT_NAAM	; If project code, project description
DB_TEXT_ADDRESS1	=ADRES1	; Project address line 1
DB_TEXT_ADDRESS2	=ADRES2	; Project address line 2
DB_TEXT_ADDRESS3	=ADRES3	; Project address line 3
DB_TEXT_SYSTEM	=REEKS	; Standard system code
DB_TEXT_PROFILE	=PROJECTPROFIEL	; Standard finish of profiles
DB_TEXT_FRAMEPROFILE	=FRAMEPROFIEL	; Standard finish of frame profiles
DB_TEXT_VENTPROFILE	=VENTPROFIEL	; Standard finish of vent profiles
DB_TEXT_GLAZBEAD	=PROJECTGLASLAT	; Standard finish of glass bead
DB_TEXT_FRAMEGLAZBEAD	=FRAMEGLASLAT	; Standard finish of frame glass bead
DB_TEXT_VENTGLAZBEAD	=VENTGLASLAT	; Standard finish of vent glass bead
DE_TEXT_STIFF	=PROJECTVERSTERK	; Standard finish of stiffning profiles
DE_TEXT_FRAMESTIFF	=FRAMEVERSTERK	; Standard finish of frame stiffning profiles
DE_TEXT_VENTSTIFF	=VENTVERSTERK	; Standard finish of vent stiffning profiles
DB_TEXT_ACC	=PROJECTBESLAG	; Standard finish of accessory
DB_TEXT_FRAMEACC	=BESLAG	; Standard finish of frame accessory
DB_TEXT_VENTACC	=BESLAG	; Standard finish of vent accessory
DB_TEXT_GLAS	=PROJECTVULLING	; Standard finish of glas
DB_TEXT_FRAMEGLAS	=FRAMEVULLING	; Standard finish of frame glas
DB_TEXT_VENTGLAS	=VENTVULLING	; Standard finish of vent glas
DB_TEXT_XTR	= PROJECTAFWERK	, Standard finish for window finishing
DB_TEXT_FRAMEXTR	= FRAMEAFWERK	, Standard finish for frame window

```
                                                    finishing
DB_TEXT_VENTXTR         = VENTAFWERK                , Standard finish for vent window
                                                    finishing
DB_TEXT_FILL            =VULLING                    ; Standard filling
DB_TEXT_BEGIN           =INPUT                      ; Input date, start project
DB_TEXT_PERIOD          =TERMIJN                    ; Delivery week
DB_TEXT_ARCHITEKT       =ARCHITECT                  ; Code of architect for customer/supplier
DB_TEXT_SELLER          =VERKOPER                   ; Code of seller for client
                                                    customer/supplier
DB_TEXT_VIEW            =POSITIE                    ;
DB_TEXT_LOCATION        =LIGGING                    ; Location, in regard to wind norm
DB_TEXT_BLOCK           =PRIJSGROEP                 ;
DB_TEXT_COEF1           =COEFFICIENT1               ; Cost coefficient 1
DB_TEXT_COEF2           =COEFFICIENT2               ; Cost coefficient 2
DB_TEXT_COEF3           =COEFFICIENT3               ; Cost coefficient 3
DB_TEXT_COST1           =KOST1                      ; Text to be used with Time 1
DB_TEXT_COST2           =KOST2                      ; Text to be used with Time 2
DB_TEXT_CURRENCY        =MUNT                       ; Currency kind, not value, ex BF
DB_TEXT_FACTOR          =FAKTOR                     ; Exchange rate currency
DB_TEXT_AMOUNT          =BEDRAG                     ; Sales value whole project
DB_TEXT_TAXTARIFF       =BTWBEDRAG                  ; VAT
DB_TEXT_DISCOUNT        =KORTING                    ; Discount on project
DB_TEXT_TRANSPORT       =VERVOER                    ; Transport cost
DB_TEXT_TAXTRANSPORT    =BTWVERVOER                 ; VAT transport
DB_TEXT_CONDITIONS      =VOORWAARDEN                ; Sales conditions, to be printed on offer
DB_TEXT_CLIENT          =KLANT                      ; Customer/supplier name
DB_TEXT_CONTACT         =KONTAKT                    ; Customer/supplier contact
DB_TEXT_CONTACT2        =KONTAKT2                   ; Customer/supplier contact second
DB_TEXT_CONTACT3        =KONTAKT3                   ; Customer/supplier contact third
DB_TEXT_STREET          =STRAAT                     ; Customer/supplier street name
DB_TEXT_ZIP             =POSTCODE                   ; Customer/supplier zip code
DB_TEXT_PLACE           =PLAATS                     ; Customer/supplier location
DB_TEXT_COUNTRY         =LAND                       ; Customer/supplier country
DB_TEXT_BOXADDRESS      =POSTBUSADRES               ; Customer/supplier street name mailbox
                                                    address
DB_TEXT_BOXCODE         =POSTBUSCODE                ; Customer/supplier zip code mailbox
                                                    address
DB_TEXT_BOXPLACE        =POSTBUSPLAATS              ; Customer/supplier location mailbox
                                                    address
DB_TEXT_PHONE           =TELEFOON                   ; Customer/supplier phone number
DB_TEXT_PHONE2          =TELEFOON2                  ; Customer/supplier phone number
                                                    contact 2
DB_TEXT_PHONE3          =TELEFOON3                  ; Customer/supplier phone number
                                                    contact 3
DB_TEXT_GSM             =GSM                        ; Customer/supplier GSM number
                                                    contact
DB_TEXT_GSM2            =GSM2                       ; Customer/supplier GSM number
                                                    contact 2
DB_TEXT_GSM3            =GSM3                       ; Customer/supplier GSM number
                                                    contact 3
DB_TEXT_TELEFAX         =TELEFAX                    ; Customer/supplier Facsimile number
DB_TEXT_TELEFAX2        =TELEFAX2                   ; Customer/supplier Facsimile number
                                                    contact 2
DB_TEXT_TELEFAX3        =TELEFAX3                   ; Customer/supplier Facsimile number
                                                    contact 3
DB_TEXT_EMAIL           =EMAIL                      ; Customer/supplier email address
DB_TEXT_EMAIL2          =EMAIL2                     ; Customer/supplier email address
                                                    contact 2
DB_TEXT_EMAIL3          =EMAIL3                     ; Customer/supplier email address
                                                    contact 3
```

DB_TEXT_TAXNUMBER	=BTWNUMMER	; Customer/supplier VAT number
DB_TEXT_ACCOUNT	=REKENING	; Customer/supplier account number
DB_TEXT_FUNCTION	=FUNCTIE	; Customer/supplier type 0 = private, 1 = seller, 2 = contractor, 3 = architect, 4 = supplier
DB_TEXT_R0	=R0	; If seller or contractor this is = 1, all others, multiplication of 3 correction values

8.27. Variety.qif

Resides in Common. Used for templates.

9. Report descriptions

Parameter	***Description***	***Example value***

9.0. Command parameters

%TEMPLATE_…		
…PLDFILENAME%	Not used	
%PAPER_…		
…TYPE%	Paper size	A4
…ORIENTATION%	Landscape or portrait	
…MARGINLEFT%		
…MARGINRIGHT%		
…MARGINTOP%		
…MARGINBOTTOM%		

User Parameters		
GetParam		%EVAL{getparam(“COMPANY”)} = %COMPANY%
SetParam		%EVAL{setparam(“VAR_FOUND”,”1”)}

9.1. Header data

These files do not create reports themselves. These parameters are available where the files are listed in a report's file list.

Files

header_project.h1
header_project.h2
header_project.h3
header_project.h4

Specific parameters

%HEADER_ENABLED%	Header enabled in slot	0 or 1
%HEADER_DATE%	System date enabled in header	0 or 1
%HEADER_CLIENT_…		
…CODE%	Customer code enabled in header	0 or 1
…NAME%	Customer name enabled in header	0 or 1
…CONTACT%	Contact person name enabled in header	0 or 1
…ADDRESS1%	Customer address first line enabled in header	0 or 1
…ADDRESS2%	Customer address second line enabled in header	0 or 1
…ADDRESS3%	Customer address third line enabled in header	0 or 1
…CITY%	Customer city enabled in header	0 or 1
…PHONE%	Customer phone enabled in header	0 or 1
…FAX%	Customer facsimile enabled in header	0 or 1
…EMAIL%	Customer email enabled in header	0 or 1
%HEADER_PROJECT_…		
…CODE%	Project name enabled in header	0 or 1
…NAME%	Project description enabled in header	0 or 1
…ADDRESS%	Project address enabled in header	0 or 1
…SERIES%	Project series enabled in header	0 or 1
…PROFILE%	Project profile colour enabled in header	0 or 1
…FRAME_PROFILE%	Project frame profile colour enabled in header	0 or 1
…VENT_PROFILE%	Project vent profile colour enabled in header	0 or 1
…BEAD%	Project glazing bead colour enabled in header	0 or 1
…FRAME_BEAD%	Project frame glazing bead colour enabled in header	0 or 1
…VENT_BEAD%	Project vent glazing bead colour enabled in header	0 or 1
…ACCESS%	Project accessories colour enabled in header	0 or 1
…FRAME_ACCESS%	Project frame accessories colour enabled in header	0 or 1
…VENT_ACCESS%	Project vent accessories colour enabled in header	0 or 1
…FORCE%	Project reinforcement colour enabled in header	0 or 1
…FRAME_FORCE%	Project reinforcement colour enabled in header	0 or 1
…VENT_FORCE%	Project reinforcement colour enabled in header	0 or 1
…GLAZING%	Project filling description enabled in header	0 or 1
…FILLING%	Project filling colour enabled in header	0 or 1
…FRAME_FILLING%	Project frame filling colour enabled in header	0 or 1
…VENT_FILLING%	Project vent filling colour enabled in header	0 or 1
…FINISH%	Project window finish colour enabled in header	0 or 1
…FRAME_FINISH%	Project frame window finish colour enabled in header	0 or 1

…VENT_FINISH%	Project vent window finish colour enabled in header	0 or 1
…NORM%	Project wind norm enabled in header	0 or 1
…DAYWEEK%	Project delivery date enabled in header	0 or 1
…MEMO%	Project info from settings enabled in header	0 or 1
…COMMENT%	Project extra information enabled in header	0 or 1

9.2. Outlines

Files

outlines_b_assembly.h1
outlines_b_assembly.h2
outlines_b_assembly.h3
outlines_b_assembly.h4
outlines_b_assembly.qrf
outlines_b_batch.qrf
outlines_b_frame.h1
outlines_b_frame.h2
outlines_b_frame.h3
outlines_b_frame.h4
outlines_b_frame.qrf
outlines_b_project.h1
outlines_b_project.h2
outlines_b_project.h3
outlines_b_project.h4
outlines_b_project.qrf
outlines_e_batch.qrf
outlines_e_project.h1
outlines_e_project.h2
outlines_e_project.h3
outlines_e_project.h4
outlines_e_project.qrf
JoPPS.QIF
header_project.h1
header_project.h2
header_project.h3
header_project.h4

Specific parameters

%OUTLINES_…		
…FRAMEMODE%	Do different parts in a group stay together	0 = per group 1 = per part
…DETAILS%	Is info block shown or not	0 = No, 1 = Yes
…COLUMNS%	No of columns in report	1 to 3
…WIDTH%	Width of the sketch in mm	70
…HEIGHT%	Height of the sketch in mm	70
…STYLE%	Sketch scaled or absolute	0 = scale (1/20) 1 = absolute(70x70mm)
…SCALE%	Scale value for sketches	1/20
…CODEINFO%	How codes should be shown	0= none, 1= glazing 2= profiles, 3= both
…ANGLEINFO%	Show angle if different from 90 degrees or not	0 = no, 1 = yes
…FORCEINFO%	Show reinforcements or not	0 = no, 1 = yes
…VIMODE%	Show info glazing or not	0 = default, 1 = never, 2 = always
…AIMODE%	Show info window finish	0 = default, 1 = never, 2 = always

9.3. Bill of material

Files

bill_off_material_b_batch.qrf (in Common)
bill_off_material_b_project.h1 (in Alu, PVC, Wood)
bill_off_material_b_project.h2 (in Alu, PVC, Wood)
bill_off_material_b_project.h3 (in Alu, PVC, Wood)
bill_off_material_b_project.h4 (in Alu, PVC, Wood)
bill_off_material_b_project.qrf (in Alu, PVC, Wood)
JoPPS.QIF
Piece.QIF
header_project.h1
header_project.h2
header_project.h3
header_project.h4

Specific parameters

%BILL_OF_MATERIAL_…		
…LENGTH%	How must profile length be represented	0 = sawing length, including extra length for PVC 1 = length 2 = inside length
…FILTER%	Department numbers for articles to be excluded from list	ex. (0,999)

9.4. Cutting list

Files

cutting_list_b_batch.qrf (in Common)
cutting_list_b_project.h1 (in Alu, PVC, Wood)
cutting_list_b_project.h2 (in Alu, PVC, Wood)
cutting_list_b_project.h3 (in Alu, PVC, Wood)
cutting_list_b_project.h4 (in Alu, PVC, Wood)
cutting_list_b_project.qrf (in Alu, PVC, Wood)
JoPPS.QIF
Piece.QIF
Profile.Qif
header_project.h1
header_project.h2
header_project.h3
header_project.h4

Specific parameters

%CUTTING_LIST_…		
…LENGTH%	How must profile length be represented	0 = sawing length, including extra length for PVC 1 = length 2 = inside length
…FILTER%	Department numbers to be excluded from list	ex. (0,999)
…BATCHREF%	Profile reference in batchmode	0 = Project code 1 = Batch code

9.5. Accessories list

Files

accessories_list_b_batch.qrf
accessories_list_b_project.qrf
accessories_list_b_project_1.h1
accessories_list_b_project_1.h2
accessories_list_b_project_1.h3
accessories_list_b_project_1.h4
accessories_list_b_project_2.h1
accessories_list_b_project_2.h2
accessories_list_b_project_2.h3
accessories_list_b_project_2.h4
JoPPS.QIF
Attrib.QIF
header_project.h1
header_project.h2
header_project.h3
header_project.h4

Specific parameters

%ACCESSORIES_LIST_…		
…FILTER%	Department numbers for articles to be excluded from list	ex. (0,999)

9.6. Glass order

Files

filling_list_b_batch.qrf
filling_list_b_project.qrf
filling_list_b_project_1.h1
filling_list_b_project_1.h2
filling_list_b_project_1.h3
filling_list_b_project_1.h4
filling_list_b_project_2.h1
filling_list_b_project_2.h2
filling_list_b_project_2.h3
filling_list_b_project_2.h4
JoPPS.qif
Glass.qif
Text.qif

Specific parameters

%FILLING_LIST_…		
…NOCOLORS%	Must background be used or not	0 = Yes, 1 = no
…WIDTH%	Width of the sketch in mm	100
…HEIGHT%	Height of the sketch in mm	100
…CODEINFO%	Show codes in filling or not	0 = no, 1 = Yes
…PRICE%	Show price or not	0 = no, 1 = Yes
…REBATE%	If price yes, show total price or with rebate	0 = no rebate, 1 = rebate
…WEIGHT%	Show weight or not	0 = Yes, 1 = no
…DETAIL%	Glazing reference	0 = divided per project (batch) 1 = Divided per group 2 = Detailed
…FILTER%	Department numbers for articles to be excluded from list	ex. (0,999)
…EXCELCOMPAT%	Is compatibility with Excel necessary or not	0 = no, 1 = Yes

9.7. Window finishes list

Files

accessories_list_b_batch.qrf
accessories_list_b_project.qrf
accessories_list_b_project_1.h1
accessories_list_b_project_1.h2
accessories_list_b_project_1.h3
accessories_list_b_project_1.h4
accessories_list_b_project_2.h1
accessories_list_b_project_2.h2
accessories_list_b_project_2.h3
accessories_list_b_project_2.h4
JoPPS.QIF
Attrib.QIF
header_project.h1
header_project.h2
header_project.h3
header_project.h4

Specific parameters

%FINISHING_LIST_…		
…NOCOLORS%	Must background be used or not	0 = Yes, 1 = no
…PRICE%	Show price or not	0 = no price, 1 = show price
…REBATE%	If price yes, show total price or with rebate	0 = no rebate, 1 = rebate
…FILTER%	Department numbers for articles to be excluded from list	ex. (0,999)
…EXCELCOMPAT%	Is compatibility with Excel necessary or not	0 = no, 1 = Yes

9.8. Part per page

Files

frame_per_page_b_assembly.qrf (Common)
frame_per_page_b_assembly.h1 (Common)
frame_per_page_b_assembly.h2 (Common)
frame_per_page_b_assembly.h3 (Common)
frame_per_page_b_assembly.h4 (Common)
frame_per_page_b_frame.qrf (Alu, PVC, Wood)
frame_per_page_b_frame_1.h1 (Alu, PVC, Wood)
frame_per_page_b_frame_1.h2 (Alu, PVC, Wood)
frame_per_page_b_frame_1.h3 (Alu, PVC, Wood)
frame_per_page_b_frame_1.h4 (Alu, PVC, Wood)
frame_per_page_b_frame_2.h1 (Common)
frame_per_page_b_frame_2.h2 (Common)
frame_per_page_b_frame_2.h3 (Common)
frame_per_page_b_frame_2.h4 (Common)
frame_per_page_b_frame_3.h1 (Common)
frame_per_page_b_frame_3.h2 (Common)
frame_per_page_b_frame_3.h3 (Common)
frame_per_page_b_frame_3.h4 (Common)
frame_per_page_b_frame_4.h1 (Common)
frame_per_page_b_frame_4.h2 (Common)
frame_per_page_b_frame_4.h3 (Common)
frame_per_page_b_frame_4.h4 (Common)
frame_per_page_b_frame_5.h1 (Common)
frame_per_page_b_frame_5.h2 (Common)
frame_per_page_b_frame_5.h3 (Common)
frame_per_page_b_frame_5.h4 (Common)
frame_per_page_b_frame_6.h1 (Common)
frame_per_page_b_frame_6.h2 (Common)
frame_per_page_b_frame_6.h3 (Common)
frame_per_page_b_frame_6.h4 (Common)
frame_per_page_b_frame_drw.h1 (Common)
frame_per_page_b_frame_drw.h2 (Common)
frame_per_page_b_frame_drw.h3 (Common)
frame_per_page_b_frame_drw.h4 (Common)
JoPPS.QIF
Piece.QIF
Attrib.QIF
Glass.QIF
Extra.QIF

Specific parameters

%FRAME_PER_PAGE_…		
…WIDTH%	Width of the sketch in mm	80
…HEIGHT%	Height of the sketch in mm	80
…FILLING_WIDTH%	Width of the sketch for special form in mm	100
…FILLING_HEIGHT%	Height of the sketch for special form in mm	100
…CODEINFO%	How codes should be shown	0= none, 1= glazing 2= profiles, 3= both
…ANGLEINFO%	Show angle if different from 90 degrees or not	0 = no, 1 = yes
…FORCEINFO%	Show reinforcements or not	0 = no, 1 = yes
…INFOBELOW%	Position extra info	0 = side, 1 = below
…LENGTH%	How must profile length be represented	0 = sawing length, including extra length for PVC 1 = length 2 = inside length
…FILTER_PIECE%	Filter on departments relating to profile pieces	(0,999)
…FILTER_ATTRIB%	Filter on departments relating to attributes	(0,999)
…FILTER_GLASS%	Filter on departments relating to filling	(0,999)
…FILTER_EXTRA%	Filter on departments relating to extra's	(0,999)
…VIMODE%	How information on filling must be shown	0 = standard, 1 = never, 2 = always
…AIMODE%	How information on finish must be shown	0 = standard, 1 = never, 2 = always

9.9. Group per page

Files

assembly_per_page_b_assembly.qrf (Alu, PVC, Wood)
assembly_per_page_b_assembly_1.h1 (Alu, PVC, Wood)
assembly_per_page_b_assembly_1.h2 (Alu, PVC, Wood)
assembly_per_page_b_assembly_1.h3 (Alu, PVC, Wood)
assembly_per_page_b_assembly_1.h4 (Alu, PVC, Wood)
assembly_per_page_b_assembly_2.h1 (Common)
assembly_per_page_b_assembly_2.h2 (Common)
assembly_per_page_b_assembly_2.h3 (Common)
assembly_per_page_b_assembly_2.h4 (Common)
assembly_per_page_b_assembly_3.h1 (Common)
assembly_per_page_b_assembly_3.h2 (Common)
assembly_per_page_b_assembly_3.h3 (Common)
assembly_per_page_b_assembly_3.h4 (Common)
assembly_per_page_b_assembly_4.h1 (Common)
assembly_per_page_b_assembly_4.h2 (Common)
assembly_per_page_b_assembly_4.h3 (Common)
assembly_per_page_b_assembly_4.h4 (Common)
assembly_per_page_b_assembly_5.h1 (Common)
assembly_per_page_b_assembly_5.h2 (Common)
assembly_per_page_b_assembly_5.h3 (Common)
assembly_per_page_b_assembly_5.h4 (Common)
assembly_per_page_b_assembly_6.h1 (Common)
assembly_per_page_b_assembly_6.h2 (Common)
assembly_per_page_b_assembly_6.h3 (Common)
assembly_per_page_b_assembly_6.h4 (Common)
assembly_per_page_b_assembly_drw.h1 (Common)
assembly_per_page_b_assembly_drw.h2 (Common)
assembly_per_page_b_assembly_drw.h3 (Common)
assembly_per_page_b_assembly_drw.h4 (Common)
JoPPS.QIF
Piece.QIF
Attrib.QIF
Glass.QIF
Extra.QIF

Specific parameters

%ASSEMBLY_PER_PAGE_…		
…WIDTH%	Width of the sketch in mm	80
…HEIGHT%	Height of the sketch in mm	80
…FILLING_WIDTH%	Width of the sketch for special form in mm	100
…FILLING_HEIGHT%	Height of the sketch for special form in mm	100
…CODEINFO%	How codes should be shown	0= none, 1= glazing 2= profiles, 3= both
…ANGLEINFO%	Show angle if different from 90 degrees or not	0 = no, 1 = yes
…FORCEINFO%	Show reinforcements or not	0 = no, 1 = yes
…INFOBELOW%	Position extra info	0 = side, 1 = below
…LENGTH%	How must profile length be represented	1 = sawing length, including extra length for PVC 2 = length 3 = inside length
…FILTER_PIECE%	Filter on departments relating to profile pieces	(0,999)
…FILTER_ATTRIB%	Filter on departments relating to attributes	(0,999)
…FILTER_GLASS%	Filter on departments relating to filling	(0,999)
…FILTER_EXTRA%	Filter on departments relating to extra's	(0,999)
…VIMODE%	How information on filling must be shown	0 = standard, 1 = never, 2 = always
…AIMODE%	How information on finish must be shown	0 = standard, 1 = never, 2 = always

9.10. Order list per supplier

This applies to both the HTML, Text and JoPPS-Script versions of this report.

Files

order_list_per_supplier_b_batch.qrf
order_list_per_supplier_b_broject.qrf
order_list_per_supplier_b_project_0.h1
order_list_per_supplier_b_project_0.h2
order_list_per_supplier_b_project_0.h3
order_list_per_supplier_b_project_0.h4
order_list_per_supplier_b_project_0.J
order_list_per_supplier_b_project_0.t1
order_list_per_supplier_b_project_0.t2
order_list_per_supplier_b_project_0.t3
order_list_per_supplier_b_project_0.t4
order_list_per_supplier_b_project_1.h1 (Common and PVC)
order_list_per_supplier_b_project_1.h2 (Common and PVC)
order_list_per_supplier_b_project_1.h3 (Common and PVC)
order_list_per_supplier_b_project_1.h4 (Common and PVC)
order_list_per_supplier_b_project_1.J
order_list_per_supplier_b_project_1.t1 (Common and PVC)
order_list_per_supplier_b_project_1.t2 (Common and PVC)
order_list_per_supplier_b_project_1.t3 (Common and PVC)
order_list_per_supplier_b_project_1.t4 (Common and PVC)
order_list_per_supplier_b_project_2.h1 (Common and PVC)
order_list_per_supplier_b_project_2.h2 (Common and PVC)
order_list_per_supplier_b_project_2.h3 (Common and PVC)
order_list_per_supplier_b_project_2.h4 (Common and PVC)
order_list_per_supplier_b_project_2.J
order_list_per_supplier_b_project_2.t1 (Common and PVC)
order_list_per_supplier_b_project_2.t2 (Common and PVC)
order_list_per_supplier_b_project_2.t3 (Common and PVC)
order_list_per_supplier_b_project_2.t4 (Common and PVC)
order_list_per_supplier_b_project_3.h1 (Common and PVC)
order_list_per_supplier_b_project_3.h2 (Common and PVC)
order_list_per_supplier_b_project_3.h3 (Common and PVC)
order_list_per_supplier_b_project_3.h4 (Common and PVC)
order_list_per_supplier_b_project_3.J
order_list_per_supplier_b_project_3.t1 (Common and PVC)
order_list_per_supplier_b_project_3.t2 (Common and PVC)
order_list_per_supplier_b_project_3.t3 (Common and PVC)
order_list_per_supplier_b_project_3.t4 (Common and PVC)
order_list_per_supplier_e_batch.qrf
order_list_per_supplier_e_broject.j
order_list_per_supplier_e_broject.qrf
JoPPS.QIF
Attrib.QIF
Text.QIF

Specific parameters

%ORDER_LIST_PER_SUPPLIER_… (HTML)		
…PRICE%	How price should be shown	0 = no price, 1 = unit price, 2 = price/packing unit
…REBATE%	Show price rebate or not	0 = no rebate, 1 = rebate
…VARIANT%	What code to show	0 = order code, 1 = colour code
…NOCOLORS%	Show background colours	0 = no, 1 = yes
…EXCELCOMPAT%	Excel compatible or not (Excel can't handle all information in standard HTML report)	0 = no, 1 = yes
%ORDER_LIST_PER_SUPPLIER_… (Text)		
…PRICE%	How price should be shown	0 = no price, 1 = unit price, 2 = price/packing unit
…REBATE%	Show price with or without rebate	0 = no rebate, 1 = rebate
…VARIANT%	What code should be printed	0 = order code, 1 = colour code
%ORDER_LIST_PER_SUPPLIER_… (JScript)		
…PRICE%	How price should be shown	0 = no price, 1 = unit price, 2 = price/packing unit
…REBATE%	Show price with or without rebate	0 = no rebate, 1 = rebate
…VARIANT%	What code should be printed	0 = order code, 1 = colour code

9.11. Order list per product

This applies to both the HTML, Text and JoPPS-Script versions of this report.

Files

order_list_per_product_b_batch.qrf
order_list_per_product_b_broject.qrf
order_list_per_product_b_project_0.h1
order_list_per_product_b_project_0.h2
order_list_per_product_b_project_0.h3
order_list_per_product_b_project_0.h4
order_list_per_product_b_project_0.J
order_list_per_product_b_project_0.t1
order_list_per_product_b_project_0.t2
order_list_per_product_b_project_0.t3
order_list_per_product_b_project_0.t4
order_list_per_product_b_project_1.h1 (Common and PVC)
order_list_per_product_b_project_1.h2 (Common and PVC)
order_list_per_product_b_project_1.h3 (Common and PVC)
order_list_per_product_b_project_1.h4 (Common and PVC)
order_list_per_product_b_project_1.J
order_list_per_product_b_project_1.t1 (Common and PVC)
order_list_per_product_b_project_1.t2 (Common and PVC)
order_list_per_product_b_project_1.t3 (Common and PVC)
order_list_per_product_b_project_1.t4 (Common and PVC)
order_list_per_product_b_project_2.h1 (Common and PVC)
order_list_per_product_b_project_2.h2 (Common and PVC)
order_list_per_product_b_project_2.h3 (Common and PVC)
order_list_per_product_b_project_2.h4 (Common and PVC)
order_list_per_product_b_project_2.J
order_list_per_product_b_project_2.t1 (Common and PVC)
order_list_per_product_b_project_2.t2 (Common and PVC)
order_list_per_product_b_project_2.t3 (Common and PVC)
order_list_per_product_b_project_2.t4 (Common and PVC)
order_list_per_product_b_project_3.h1 (Common and PVC)
order_list_per_product_b_project_3.h2 (Common and PVC)
order_list_per_product_b_project_3.h3 (Common and PVC)
order_list_per_product_b_project_3.h4 (Common and PVC)
order_list_per_product_b_project_3.J
order_list_per_product_b_project_3.t1 (Common and PVC)
order_list_per_product_b_project_3.t2 (Common and PVC)
order_list_per_product_b_project_3.t3 (Common and PVC)
order_list_per_product_b_project_3.t4 (Common and PVC)
order_list_per_product_b_project_4.h1
order_list_per_product_b_project_4.h2
order_list_per_product_b_project_4.h3
order_list_per_product_b_project_4.h4
order_list_per_product_b_project_4.J
order_list_per_product_b_project_4.t1
order_list_per_product_b_project_4.t2
order_list_per_product_b_project_4.t3
order_list_per_product_b_project_4.t4
order_list_per_product_e_batch.qrf
order_list_per_product_e_broject.j
order_list_per_product_e_broject.qrf
JoPPS.QIF
Attrib.QIF

Specific parameters

%ORDER_LIST_PER_PRODUCT_… (HTML)

…PRICE%	How price should be shown	0 = no price, 1 = unit price, 2 = price/packing unit
…REBATE%	Show price rebate or not	0 = no rebate, 1 = rebate
…VARIANT%	What code to show	0 = order code, 1 = colour code
…NOCOLORS%	Show background colours	0 = no, 1 = yes
…EXCELCOMPAT%	Excel compatible or not (Excel can't handle all information in standard HTML report)	0 = no, 1 = yes
%ORDER_LIST_PER_PRODUCT_… (Txt)		
…PRICE%	How price should be shown	0 = no price, 1 = unit price, 2 = price/packing unit
…REBATE%	Show price with or without rebate	0 = no rebate, 1 = rebate
…VARIANT%	What code should be printed	0 = order code, 1 = colour code
%ORDER_LIST_PER_PRODUCT_… (JScript)		
…PRICE%	How price should be shown	0 = no price, 1 = unit price, 2 = price/packing unit
…REBATE%	Show price with or without rebate	0 = no rebate, 1 = rebate
…VARIANT%	What code should be printed	0 = order code, 1 = colour code

9.12. Optimization

Does not apply to wood.

Files

optimization_b_batch.qrf (in Common)
optimization_b_project.h1 (in Common , PVC)
optimization_b_project.h2 (in Common , PVC)
optimization_b_project.h3 (in Common , PVC)
optimization_b_project.h4 (in Common , PVC)
optimization_b_project.qrf (in Alu, PVC, Wood)
JoPPS.QIF
Piece.QIF
Profile.Qif
header_project.h1
header_project.h2
header_project.h3
header_project.h4

Specific parameters

%OPTIMIZATION_…		
…LENGTH%	How must profile length be represented	0 = sawing length, including extra length for PVC 1 = length 2 = inside length
…FILTER%	Filter on departments numbers	(0,999)
…BATCHREF%	How must the reference be shown when running in batch	0 = project code, 1 = batch code

9.13. Estimation detail

Files

estimation_detail_b_batch.qrf
estimation_detail_b_frame.h1
estimation_detail_b_frame.h2
estimation_detail_b_frame.h3
estimation_detail_b_frame.h4
estimation_detail_b_frame.qrf
estimation_detail_b_project.qrf
JoPPS.QIF
Cost.QIF
header_project.h1
header_project.h2
header_project.h3
header_project.h4

Specific parameters

%ESTIMATION_DETAIL_…		
…OUTLINES%	Show outline or not	0 = no, 1 = yes
…WIDTH%	Width of outline	30mm
…HEIGHT%	Height of outline	30mm

9.14. Estimation overview

Files

estimation_summary_b_batch.qrf
estimation_summary_b_frame.h1
estimation_summary_b_frame.h2
estimation_summary_b_frame.h3
estimation_summary_b_frame.h4
estimation_summary_b_frame.qrf
estimation_summary_b_project.qrf
JoPPS.QIF
Cost.QIF
header_project.h1
header_project.h2
header_project.h3
header_project.h4

Specific parameters

none

9.15. Calculation detail

Files

calculation_detail_b_frame.qrf (Common and Wood)
calculation_detail_b_frame_1.h1 (Common and Wood)
calculation_detail_b_frame_1.h2 (Common and Wood)
calculation_detail_b_frame_1.h3 (Common and Wood)
calculation_detail_b_frame_1.h4 (Common and Wood)
calculation_detail_b_frame_2.h1
calculation_detail_b_frame_2.h2
calculation_detail_b_frame_2.h3
calculation_detail_b_frame_2.h4
calculation_detail_b_frame_3.h1
calculation_detail_b_frame_3.h2
calculation_detail_b_frame_3.h3
calculation_detail_b_frame_3.h4
calculation_detail_b_frame_4.h1
calculation_detail_b_frame_4.h2
calculation_detail_b_frame_4.h3
calculation_detail_b_frame_4.h4
calculation_detail_b_frame_5.h1
calculation_detail_b_frame_5.h2
calculation_detail_b_frame_5.h3
calculation_detail_b_frame_5.h4
calculation_detail_b_frame_drw.h1
calculation_detail_b_frame_drw.h2
calculation_detail_b_frame_drw.h3
calculation_detail_b_frame_drw.h4
JoPPS.Qif
Piece.Qif
Attrib.Qif
Glass.Qif
Extra.Qif

Specific parameters

%CALCULATION_DETAIL_…		
…OUTLINES%	Show outline or not	0 = no, 1 = yes
…WIDTH%	Width of outline	30mm
…HEIGHT%	Height of outline	30mm
…REBATE%	Show price with or without rebate	0 = no rebate, 1 = rebate

9.16. Calculation overview

Files

calculation_summary_b_batch.qrf
calculation_summary_b_project.qrf (Common and Wood)
calculation_summary_b_project_1.h1 (Common and Wood)
calculation_summary_b_project_1.h2 (Common and Wood)
calculation_summary_b_project_1.h3 (Common and Wood)
calculation_summary_b_project_1.h4 (Common and Wood)
calculation_summary_b_project_2.h1
calculation_summary_b_project_2.h2
calculation_summary_b_project_2.h3
calculation_summary_b_project_2.h4
calculation_summary_b_project_3.h1
calculation_summary_b_project_3.h2
calculation_summary_b_project_3.h3
calculation_summary_b_project_3.h4
calculation_summary_b_project_4.h1
calculation_summary_b_project_4.h2
calculation_summary_b_project_4.h3
calculation_summary_b_project_4.h4
calculation_summary_b_project_5.h1
calculation_summary_b_project_5.h2
calculation_summary_b_project_5.h3
calculation_summary_b_project_5.h4
calculation_summary_b_project_6.h1
calculation_summary_b_project_6.h2
calculation_summary_b_project_6.h3
calculation_summary_b_project_6.h4
JoPPS.Qif
Piece.Qif
Attrib.Qif
Glass.Qif
Extra.Qif

Specific parameters

%CALCULATION_SUMMARY_…		
…REBATE%	Show price with or without rebate	0 = no rebate, 1 = rebate

9.17. Sections

Files

section_b_assembly.h1
section_b_assembly.h2
section_b_assembly.h3
section_b_assembly.h4
section_b_assembly.j
section_b_assembly.qrf
section_b_assembly_addsections.j
section_b_project.h1
section_b_project.h2
section_b_project.h3
section_b_project.h4
section_b_project.j
section_b_project.qrf
section_e_project.j
section_e_project.qrf
JoPPS.Qif

Specific parameters

none

9.18. Offer

This applies to both the HTML and JoPPS-Script versions of this report.

Files

offer_b_project.qrf
offer_b_project_1.h1
offer_b_project_1.h2
offer_b_project_1.h3
offer_b_project_1.h4
offer_b_project_1.j
offer_b_project_1_architect.h1 (Common and Wood)
offer_b_project_1_architect.h2 (Common and Wood)
offer_b_project_1_architect.h3 (Common and Wood)
offer_b_project_1_architect.h4 (Common and Wood)
offer_b_project_1_architect.j
offer_b_project_1_contractor.h1 (Common and Wood)
offer_b_project_1_contractor.h2 (Common and Wood)
offer_b_project_1_contractor.h3 (Common and Wood)
offer_b_project_1_contractor.h4 (Common and Wood)
offer_b_project_1_contractor.j
offer_b_project_1_private.h1 (Common and Wood)
offer_b_project_1_private.h2 (Common and Wood)
offer_b_project_1_private.h3 (Common and Wood)
offer_b_project_1_private.h4 (Common and Wood)
offer_b_project_1_private.j
offer_b_project_1_seller.h1 (Common and Wood)
offer_b_project_1_seller.h2 (Common and Wood)
offer_b_project_1_seller.h3 (Common and Wood)
offer_b_project_1_seller.h4 (Common and Wood)
offer_b_project_1_seller.j
offer_b_project_1_supplier.h1 (Common and Wood)
offer_b_project_1_supplier.h2 (Common and Wood)
offer_b_project_1_supplier.h3 (Common and Wood)
offer_b_project_1_supplier.h4 (Common and Wood)
offer_b_project_1_supplier.j
offer_b_project_2.h1
offer_b_project_2.h2
offer_b_project_2.h3
offer_b_project_2.h4
offer_b_project_2.j
offer_b_project_2_long.h1
offer_b_project_2_long.h2
offer_b_project_2_long.h3
offer_b_project_2_long.h4
offer_b_project_2_long.j
offer_b_project_2_long_subtot.h1
offer_b_project_2_long_subtot.h2
offer_b_project_2_long_subtot.h3
offer_b_project_2_long_subtot.h4
offer_b_project_2_long_subtot.j
offer_b_project_2_short.h1
offer_b_project_2_short.h2
offer_b_project_2_short.h3
offer_b_project_2_short.h4
offer_b_project_2_short.j
offer_b_project_2_short_subtot.j
offer_b_project_3.h1
offer_b_project_3.h2
offer_b_project_3.h3

offer_b_project_3.h4
offer_b_project_3.j
offer_b_project_4.h1
offer_b_project_4.h2
offer_b_project_4.h3
offer_b_project_4.h4
offer_b_project_4.j
offer_b_project_5.h1
offer_b_project_5.h2
offer_b_project_5.h3
offer_b_project_5.h4
offer_b_project_5.j
offer_b_project_6.h1
offer_b_project_6.h2
offer_b_project_6.h3
offer_b_project_6.h4
offer_b_project_6.j
offer_b_project_6_architect.h1
offer_b_project_6_architect.h2
offer_b_project_6_architect.h3
offer_b_project_6_architect.h4
offer_b_project_6_architect.j
offer_b_project_6_contractor.h1
offer_b_project_6_contractor.h2
offer_b_project_6_contractor.h3
offer_b_project_6_contractor.h4
offer_b_project_6_contractor.j
offer_b_project_6_private.h1
offer_b_project_6_private.h2
offer_b_project_6_private.h3
offer_b_project_6_private.h4
offer_b_project_6_private.j
offer_b_project_6_seller.h1
offer_b_project_6_seller.h2
offer_b_project_6_seller.h3
offer_b_project_6_seller.h4
offer_b_project_6_seller.j
offer_b_project_6_supplier.h1
offer_b_project_6_supplier.h2
offer_b_project_6_supplier.h3
offer_b_project_6_supplier.h4
offer_b_project_6_supplier.j
offer_e_project.j
offer_e_project.qrf
JoPPS.Qif
Offer.Qif
Text.Qif

Specific parameters

%OFFER_… (HTML)		
…NOCOLORS%	Must background be used or not	0 = Yes, 1 = no
…OUTLINES%	Show outline or not	0 = no, 1 = yes
…FORMAT%	Type of offer	0 = price per group, 1 = in detail
…WIDTH%	Width of outline	30mm
…HEIGHT%	Height of outline	30mm
…WORDCOMPAT%	Word compatible or not (Word has problems with certain aspects of out HTML documents)	0 = no, 1 = yes
%OFFER_…		
…NOCOLORS%	Must background be used or not	0 = Yes, 1 = no
…FORMAT%	Type of offer	0 = price per group, 1 = in detail
…WIDTH%	Width of outline	30mm
…HEIGHT%	Height of outline	30mm
…TEMPLATEFN%	Template to be used to create Word document	Offer

9.19. Offer sketches

Files

offer_outlines_b_assembly.h1
offer_outlines_b_assembly.h2
offer_outlines_b_assembly.h3
offer_outlines_b_assembly.h4
offer_outlines_b_assembly.qrf
offer_outlines_b_project.h1
offer_outlines_b_project.h2
offer_outlines_b_project.h3
offer_outlines_b_project.h4
offer_outlines_b_project.qrf
offer_outlines_e_project.h1
offer_outlines_e_project.h2
offer_outlines_e_project.h3
offer_outlines_e_project.h4
offer_outlines_e_project.qrf
Jopps.Qif
Offer.Qif
Text.Qif
header_project.h1
header_project.h2
header_project.h3
header_project.h4

Specific parameters

%OFFER_OUTLINES_…		
…NOCOLORS%	Must background be used or not	0 = Yes, 1 = no
…COLUMNS%	No of columns in report	1 to 3
…WIDTH%	Width of outline	30mm
…HEIGHT%	Height of outline	30mm
…STYLE%	Sketch scaled or absolute	0 = scale (1/20) 1 = absolute(70x70mm)
…SCALE%	Scale value for sketches	1/20

9.20. Invoice

Files

invoice_b_project.qrf
invoice_b_project_1.h1
invoice_b_project_1.h2
invoice_b_project_1.h3
invoice_b_project_1.h4
invoice_b_project_2.h1
invoice_b_project_2.h2
invoice_b_project_2.h3
invoice_b_project_2.h4
invoice_b_project_3.h1
invoice_b_project_3.h2
invoice_b_project_3.h3
invoice_b_project_3.h4
Jopps.Qif
Offer.Qif
Text.Qif

Specific parameters

%INVOICE_…		
…NOCOLORS%	Must background be used or not	0 = Yes, 1 = no
…WORDCOMPAT%	Word compatible or not (Word has problems with certain aspects of out HTML documents)	0 = no, 1 = yes

9.21. Confirmation

Files

confirmation_b_project.qrf
confirmation_b_project_1.h1
confirmation_b_project_1.h2
confirmation_b_project_1.h3
confirmation_b_project_1.h4
confirmation_b_project_2.h1
confirmation_b_project_2.h2
confirmation_b_project_2.h3
confirmation_b_project_2.h4
confirmation_b_project_3.h1
confirmation_b_project_3.h2
confirmation_b_project_3.h3
confirmation_b_project_3.h4
confirmation_b_project_4.h1
confirmation_b_project_4.h2
confirmation_b_project_4.h3
confirmation_b_project_4.h4
Jopps.Qif
Offer.Qif
Text.Qif

Specific parameters

%CONFIRMATION_…		
…NOCOLORS%	Must background be used or not	0 = Yes, 1 = no
…OUTLINES%	Show outline or not	0 = no, 1 = yes
…WIDTH%	Width of outline	30mm
…HEIGHT%	Height of outline	30mm
…WORDCOMPAT%	Word compatible or not (Word has problems with certain aspects of out HTML documents)	0 = no, 1 = yes

9.22. Delivery note

Files

delivery_note_b_assembly.h1
delivery_note_b_assembly.h2
delivery_note_b_assembly.h3
delivery_note_b_assembly.h4
delivery_note_b_assembly.qrf
delivery_note_b_project.h1
delivery_note_b_project.h2
delivery_note_b_project.h3
delivery_note_b_project.h4
delivery_note_b_project.qrf
delivery_note_e_project.h1
delivery_note_e_project.h2
delivery_note_e_project.h3
delivery_note_e_project.h4
delivery_note_e_project.qrf
Jopps.Qif
Offer.Qif
Text.Qif
header_project.h1
header_project.h2
header_project.h3
header_project.h4

Specific parameters

%DELIVERY_NOTE_…		
…NOCOLORS%	Must background be used or not	0 = Yes, 1 = no
…COLUMNS%	No of columns in report	1 to 3
…WIDTH%	Width of outline	30mm
…HEIGHT%	Height of outline	30mm
…STYLE%	Sketch scaled or absolute	0 = scale (1/20) 1 = absolute(70x70mm)
…SCALE%	Scale value for sketches	1/20
…CODEINFO%	Show codes in code or not	0 = no codes, 1 = show codes

9.23. Statistics

Files

statistics_b_batch.qrf
statistics_b_project.h1
statistics_b_project.h2
statistics_b_project.h3
statistics_b_project.h4
statistics_b_project.qrf
statistics_e_assembly.h1
statistics_e_assembly.h2
statistics_e_assembly.h3
statistics_e_assembly.h4
statistics_e_assembly.qrf
statistics_b_batch.qrf
statistics_e_project.h1
statistics_e_project.h2
statistics_e_project.h3
statistics_e_project.h4
statistics_e_project.qrf
Jopps.Qif
header_project.h1
header_project.h2
header_project.h3
header_project.h4

Specific parameters

%STATISTICS_…		
…COLUMNS%	No of columns in report	1 to 3
…WIDTH%	Width of outline	30mm
…HEIGHT%	Height of outline	30mm

9.24. Operations

9.25. Glazing order

9.26. Cutting list (labels)

Files

lbl_cutting_list.l1 (Common, PVC)
lbl_cutting_list.l2 (Common, PVC)
lbl_cutting_list.l3 (Common, PVC)
lbl_cutting_list.l4 (Common, PVC)
lbl_cutting_list_b_batch.qrf (Common)
lbl_cutting_list_b_project.qrf (Alu, PVC, Wood)
JoPPS.Qif
Piece.Qif
Profile.Qif

Specific parameters

%LBL_CUTTING_LIST_…		
…LENGTH%	How must profile length be represented	0 = sawing length, including extra length for PVC 1 = length 2 = inside length
…FILTER%	Filter on departments numbers	(0,999)
…BATCHREF%	How must the reference be shown when running in batch	0 = project code, 1 = batch code

9.27. Optimization (labels)

Files

lbl_optimization_list.l1 (Common, PVC)
lbl_optimization_list.l2 (Common, PVC)
lbl_optimization_list.l3 (Common, PVC)
lbl_optimization_list.l4 (Common, PVC)
lbl_optimization_list_b_batch.qrf (Common)
lbl_optimization_list_b_project.qrf (Alu, PVC, Wood)
JoPPS.Qif
Piece.Qif
Profile.Qif

Specific parameters

%LBL_OPTIMIZATION_…		
…LENGTH%	How must profile length be represented	0 = sawing length, including extra length for PVC 1 = length 2 = inside length
…FILTER%	Filter on departments numbers	(0,999)
…BATCHREF%	How must the reference be shown when running in batch	0 = project code, 1 = batch code

9.28. Own report 1 thru 24

Editing jopps.ini can configure these.

The following lines have to be added for each report:

```
[REPORT_38]                ; for user_2 (see slot numbers next page)
Tag=abc                    ; Name of the report in the menu
OutputType=x               ; 0 = HTML, 1 = Label, 2 = Text, 3 = JoPPS script,
                             4 = Machine control
```

The following lines could be added:

```
CanBatch=                  ; 0 = cannot be run in batch, 1 = can be run in batch
Extension=                 ; 3 characters maximum
```

9.29. Slot constants

0 SLOT_OUTLINES
1 SLOT_BILL_OF_MATERIAL
2 SLOT_CUTTING_LIST
3 SLOT_ACCESSORIES_LIST
5 SLOT_FILLING_LIST
6 SLOT_FINISHING_LIST
7 SLOT_FRAME_PER_PAGE
8 SLOT_ASSEMBLY_PER_PAGE
9 SLOT_ORDERLIST_PER_SUPPLIER
10 SLOT_ORDERLIST_PER_PRODUCT
11 SLOT_OPTIMIZATION
12 SLOT_ESTIMATION_DETAIL
13 SLOT_ESTIMATION_SUMMARY
14 SLOT_CALCULATION_DETAIL
15 SLOT_CALCULATION_SUMMARY
16 SLOT_SECTIONS_HTML
17 SLOT_OFFER_HTML
18 SLOT_OFFER_OUTLINES
19 SLOT_INVOICE
20 SLOT_CONFIRMATION
21 SLOT_DELIVERY_NOTE
22 SLOT_STATISTICS
23 SLOT_OUTLINES_LBL, not used
24 SLOT_FINISHING_LBL, not used
25 SLOT_CUTTING_LIST_LBL
26 SLOT_OPTIMIZATION_LBL
27 SLOT_ORDERLIST_PER_SUPPLIER_TEXT
28 SLOT_ORDERLIST_PER_PRODUCT_TEXT
29 SLOT_OFFER_SCRIPT
30 SLOT_ORDERLIST_PER_SUPPLIER_SCRIPT
31 SLOT_ORDERLIST_PER_PRODUCT_SCRIPT
32 SLOT_SECTIONS_SCRIPT
33 SLOT_BILL_OF_MATERIAL_TEXT
34 SLOT_CUTTING_LIST_TEXT, not used
35 SLOT_ACCESSORIES_LIST_TEXT, not used
36 SLOT_FINISHING_LIST_TEXT, not used
37 SLOT_FRAME_LBL, not used
38 SLOT_USER1
39 SLOT_USER2
40 SLOT_USER3
41 SLOT_USER4
42 SLOT_USER5
43 SLOT_OPERATIONS
44 SLOT_OPERATIONS_TEXT
45 SLOT_GLAZING_LIST
50 SLOT_USER6
51 SLOT_USER7
52 SLOT_USER8
53 SLOT_USER9
54 SLOT_USER10
55 SLOT_USER11
56 SLOT_USER12
64 SLOT_USER13
65 SLOT_USER14
66 SLOT_USER15
67 SLOT_USER16
68 SLOT_USER17
69 SLOT_USER18

70 SLOT_USER19
71 SLOT_USER20
72 SLOT_USER21
73 SLOT_USER22
74 SLOT_USER23
75 SLOT_USER24

10. SQL select statement

In order for data to be represented in reports, the fields have to be selected first. This is done in SQL **SELECT** statements of a *.QRF file. This is only a short description of the use of the SELECT statement. For more in depth knowledge, we expect you to get hold of a good book regarding SQL.

Example from finishing_list_b_project.QRF:

```
select   min(e.%DB_EXTRA_BATCH%) as %DB_EXTRA_BATCH%,
           sum(e.%DB_EXTRA_NO%) as %DB_EXTRA_NO%,
           e.%DB_EXTRA_DESC%,
           e.%DB_EXTRA_WIDTH%,
           e.%DB_EXTRA_HEIGHT%,
           sum(e.%DB_EXTRA_PRICE% * %M_DB_EXTRA_REBATE%) as %DB_EXTRA_PRICE%,
           e.%DB_EXTRA_SUPPLIER%,
           e.%DB_EXTRA_PROJECT%,
           e.%DB_EXTRA_ASSEMBLY%,
           e.%DB_EXTRA_FRAME%,
           min(t.%DB_TEXT_CLIENT%) as %DB_TEXT_CLIENT%,
           min(t.%DB_TEXT_STREET%) as %DB_TEXT_STREET%,
           min(t.%DB_TEXT_ZIP%) as %DB_TEXT_ZIP%,
           min(t.%DB_TEXT_PLACE%) as %DB_TEXT_PLACE%,
           min(t.%DB_TEXT_PHONE%) as %DB_TEXT_PHONE%,
           min(t.%DB_TEXT_TELEFAX%) as %DB_TEXT_TELEFAX%,
           min(t.%DB_TEXT_NAME%) as %DB_TEXT_NAME%,
           e.%DB_EXTRA_LABEL%,
           min(t.%DB_TEXT_BEGIN%) as %DB_TEXT_BEGIN%
  from     %TBL_EXTRA% e, %TBL_TEXT% t
  where    e.%DB_EXTRA_SUPPLIER% = t.%DB_TEXT_CODE%
  and      e.%DB_EXTRA_DEPNO% not in %FINISHING_LIST_FILTER%
  group by e.%DB_EXTRA_SUPPLIER%, e.%DB_EXTRA_PROJECT%, e.%DB_EXTRA_ASSEMBLY%,
           e.%DB_EXTRA_FRAME%, e.%DB_EXTRA_DESC%, e.%DB_EXTRA_WIDTH%,
           e.%DB_EXTRA_HEIGHT%, e.%DB_EXTRA_LABEL%
  having   count(*) > 0
```

10.1. SELECT-clause

This clause specifies which columns from the table are to be represented in the result, as such, it selects the columns., and generates a new (in memory) table.

In case the column is not used in the GROUP BY-clause, we will specify which value has to be retained for later use.

In this example, we will use the lowest batch code, the sum of the number of clauses, …..

10.2. FROM-clause

This clause indicates from which table fields columns will be used. In above example there are two tables, EXTRA (in which all extra clauses of a project are put together) and the TEXT table (in which general information is put together).

Note that behind the table name an alias (usually 1 character) is given for the table. When multiple tables are being used, the alias is necessary to tell the SQL interpreter from which table to get the field. The other lines refer to the table through this alias (followed by a dot).

10.3. WHERE-clause

Specifies the condition which rows of the selected tables will be withheld for further use. Only these rows will create the content of the resulting table. Can be regarded as a loop in which the field in front of the equal sign, for every row in the column, is compared to the field behind the equal sign, for every row in the related column.

In above example, the condition is that only rows where the supplier code in the EXTRA table is equal to the customer/supplier code in the TEXT-table (indicating that the information of the said clauses in the EXTRA table regarding the supplier, exists in the TEXT-table).

Another condition is that the department number of the clause is not included in the filter list (by default, department 0 and 999 are filtered out)

10.4. GROUP BY-clause

This clause will organise the rows in the temporary table. The grouping is done using the columns specified. After this, a column can have more then 1 value, depending on the number of rows that have been grouped together.

In the SELECT clause, an action will be defined on the columns that are not grouped, to indicate the value to be used in the report.

In this example, rows will be grouped per supplier, then per project, assembly, frame, description of the clause, width and height of the clause, and last a description.

10.5. HAVING-clause

This clause is similar to the WHERE-clause. The difference is that WHERE acts on the rows of tables in the FROM-clause and HAVING acts on each group generated by the GROUP BY-clause.

(In this example all groups , have to be generated from at least 1 row). This because of a BDE bug.

10.6. ORDER BY-clause

This clause re-orders the resulting table, which can be necessary for certain break definitions.

11. HTML

This section describes the available HTML tags and their options as supported by the internal HTML Viewer in J*o*PPS and Data Reporter. For a full description, consult other sources of information, like books, the internet, etc. This document will list the supported tags and options, and describe the more important, commonly used tags in the current reports.

Although the internal HTML Viewer is fairly compatible with existing HTML standards, you should keep in mind that, if you plan to import the generated HTML output into other software packages (especially non-browser applications) such as MS Word and MS Excel, lots of tags and options may not or may be interpreted incomplete or even different. This is entirely due to these programs' limitations. A few reports have a compatibility setting, which should ease import into these packages.

11.1. Supported tags and options

(Source: David Baldwin's HTMLViewer)

Document Tags

```
<HTML>...</HTML>
<HEAD>...</HEAD>
<BASE>
   HREF=base
<TITLE>...</TITLE>
<BODY>...</BODY>
   BACKGROUND=bitmap
   TEXT=color
   BGCOLOR=color
   LINK=color
```

Physical Phrase Mark-up Tags

```
<PAGE>
<B>...</B>
<I>...</I>
<U>...</U>
<TT>...</TT>
<S>...</S>
<FONT>...</FONT>
   SIZE=n  (n=1..7) or SIZE=+-n (incremental change)
   COLOR=color
   FACE=facename
<SUB>...</SUB>
<SUP>...</SUP>
<BIG>...</BIG>
<SMALL>...</SMALL>
<STRIKE>...</STRIKE>
```

Logical Phrase Mark-up Tags

```
<EM>...</EM>
<STRONG>...</STRONG>
<CITE>...</CITE>
<VAR>...</VAR>
<CODE>...</CODE>
<KBD>...</KBD>
<SAMP>...</SAMP>
```

Block Tags

```
<P>...(</P>)
    ALIGN=left|center|right
<ADDRESS>...</ADDRESS>
<BLOCKQUOTE>...</BLOCKQUOTE>
<PRE>...</PRE>
<CENTER>...</CENTER>
<DIV>...</DIV>
    ALIGN=left|center|right
```

Lists

```
<OL>...</OL>
   START=value
   TYPE=1|a|A|i|I
<UL>...</UL>
   PLAIN
<DIR>...</DIR>
<MENU>...</MENU>
  <LI>...(</LI>)
```

Description List

```
<DL>...</DL>
  <DD>...</DD>
  <DT>...</DT>
```

Tables

```
<TABLE>...<TABLE>
   BORDER or BORDER=0
   ALIGN=left|center|right
   CELLSPACING=value
   CELLPADDING=value
   WIDTH=n|n%
   BGCOLOR=color
   BORDERCOLOR=color
<CAPTION>...</CAPTION>
   ALIGN=top|bottom
<TR>...</TR>
   VALIGN=top|middle|bottom
   ALIGN=left|center|right
   BGCOLOR=color
<TH>...</TH>, <TD>...</TD>
   ROWSPAN=n
   COLSPAN=n
   WIDTH=n|n%
   HEIGHT=n
   VALIGN=top|middle|bottom
   ALIGN=left|center|right
   BGCOLOR=color
```

Forms

```
<FORM>...</FORM>
   METHOD=post|get
   ACTION=url
<INPUT>...</INPUT>
   TYPE=text|password|submit|reset|radio|checkbox|hidden|image|button
   NAME=string
   VALUE=string
   SIZE=chars       (text, password)
   MAXLENGTH=chars  (text, password)
   CHECKED          (radio, checkbox)
   ONCLICK=string   (button, radio, checkbox)
<SELECT>...</SELECT>
   NAME=string
   MULTIPLE
   SIZE=lines
   <OPTION>...(</OPTION>)
      SELECTED
      VALUE=string
<TEXTAREA>...</TEXTAREA>
   NAME=string
   ROWS=n
   COLS=n
<OPTION>...</OPTION>
```

Character Level Elements

```
<BR>
   CLEAR=left|right
<IMG>
   SRC=image
   ALT=string
   ALIGN=top|middle|bottom|left|right
```

```
   BORDER=0
   WIDTH=n,n%
   HEIGHT=n
   TRANSP   (non-standard, Lower Left pixel defines transparent colour
   IMAGEMAP
   USEMAP=url (local URL only)
```

Anchors

```
<A>...</A>
   HREF=url
   NAME=string
```

Client Side Image Maps

```
<MAP>...</MAP>
   NAME=string
<AREA>
   SHAPE=RECT|CIRC|CIRCLE|POLY|POLYGON
   COORDS="n1, n2, ...., nn"
   HREF=url
   NOHREF
```

Miscellaneous

```
<!--...-->
<Hn>...</Hn>   n = 1..6
   ALIGN=left|center|right
<HR>
   SIZE=n
   WIDTH=n|n%
   COLOR=color
<BASEFONT>
   SIZE=n  (n=1..7) or SIZE=+-n (incremental change)
<BGSOUND>
   SRC=filename
   LOOP=n
```

Colour Syntax

Colour information may consist of a hexadecimal red, green, and blue value such as `FF8000` or be one of the following identifiers:

```
   Black, Maroon, Green, Olive, Navy, Purple, Teal, Grey,
   Silver, Red, Lime, Yellow, Blue, Fuchsia, Aqua, White
```

11.2. Table tags

The following schematic displays the normal usage of table related tags:

```
<table>
    |---------------|----------------|
<tr>|<th>           |<th>            |
    |          </th>|           </th>|</tr>
    |---------------|----------------|
<tr>|<td>           |<td>            |
    |          </td>|           </td>|</tr>
    |---------------|----------------|
</table>
```

<TABLE> options

Option	*Description*
BORDER *or* BORDER=0|1	Whether or not a 3D-shape grey border is drawn around all cells and the table. BORDER is equivalent to BORDER=1.
ALIGN=left|center|right	Where the whole table is put horizontally related to the current paragraph of the HTML document.
CELLSPACING=value	The width in pixels (120 dpi) of the border area, being the outer border and the space between cells (all directions).
CELLPADDING=value	The width in pixels (120 dpi) of the unused area (margin) inside a cell (all directions) next to the border area.
WIDTH=n|n%	The total width in pixels (120 dpi) or percentage of the client area where the table comes.
BGCOLOR=color	The default background colour of all cells in this table.
BORDERCOLOR=color	The fill colour of the border area, which is the outer border, the space between cells (all directions) and the unused space due to undefined or empty cells. When BORDER or BORDER=1 is specified, the 3D border is drawn on top of the outer pixels of the border area.

<TR> options

Option	*Description*
VALIGN=top|middle|bottom	The default vertical alignment inside cells of this row.
ALIGN=left|center|right	The default horizontal alignment inside cells of this row.
BGCOLOR=color	The default background colour inside cells of this row if different from the <TABLE> BGCOLOR colour.

<TH> and <TD> options

Option	*Description*
ROWSPAN=n	If specified and bigger then 1, this cell is actually *n* rows high. On the extra rows, do not define a cell for this column.
COLSPAN=n	If specified and bigger then 1, this cell is actually *n* columns wide. Do not define cells for the extra columns on this row.
WIDTH=n|n%	The cell's width in pixels (120 dpi) or the percentage of the table's width. If the cell is spanned over more then 1 column, it is the width of the span (means the total width of all cells spanned).
HEIGHT=n	The cell's height in pixels (120 dpi) or the height of all cells spanned.
VALIGN=top|middle|bottom	The cell's vertical alignment if different from the default (middle) or the VALIGN value of the <TR> tag.
ALIGN=left|center|right	The cell's horizontal alignment if different from the default (centre for <TH>, left for <TD>) or the <TR> ALIGN value.
BGCOLOR=color	The background colour inside the cell if different from the <TABLE> and <TR> BGCOLOR colour.

11.3. Other important tags

Tag and options	*Description*
<PAGE>	Forces a page break when printing document.
<B>...</B>	Text between the tags is bold.
<I>...</I>	Text between the tags is italic.
<U>...</U>	Text between the tags is underlined.
<FONT>...</FONT> SIZE=*n* (*n* = 1..7) *or* SIZE=+-n (incremental change) COLOR=color	Text between the tags is altered in the following ways: • SIZE sets the font to size *n*, or increments/decrements the size by *+-n* • COLOR specifies the new colour for the text
**_{**...}	Text between the tags is in subscript.
**^{**...}	Text between the tags is in superscript.
<P>...(</P>) ALIGN=left|center|right	Specifies a paragraph. Text inside the paragraph can be horizontally aligned. A line break and some extra space between 2 paragraphs are automatically respected when using multiple paragraphs. If the </P> tag is immediately followed by a <P> tag, </P> may be omitted.
** **	Forces a line break.
<H*n*>...</H*n*> (*n* = 1..6)	Text between the tags becomes a header paragraph with font size *n*.
<HR> SIZE=n WIDTH=n|n% COLOR=color	Draws a centred horizontal row (line) with the specified width in pixels (120 dpi) or percentage of the available client area where the horizontal row comes. SIZE specifies the height in pixels (120 dpi) [1..20] of the row. If COLOR is specified, a coloured rectangle is drawn, otherwise a hollow 3D row is drawn.
<!--...-->	Text between <!-- and --> is handled as a comment.

11.4. HTML characters

Special characters can be used to allow characters that would otherwise be considered syntax elements. There are 2 possible syntaxes:

1) **&#*n*;** where *n* is the ANSI code for the character
2) **&*char*;** where *char* is one of the following (case sensitive):

&*char*;	*Meaning*	*ANSI code*
"	double quote character	34
&	ampersand	38
<	less then symbol	60
>	greater then symbol	62
	no-break space	160
¡	inverted exclamation mark	161
¢	cent sign	162
£	pound sterling sign	163
¤	general currency sign	164
¥	yen sign	165
¦	broken (vertical) bar	166
§	section sign	167
¨	umlaut (dieresis)	168
©	copyright sign	169
ª	ordinal indicator', feminine	170
«	angle quotation mark', left	171
¬	not sign	172
	soft hyphen	173
®	registered sign	174
¯	macron	175
°	degree sign	176
±	plus-or-minus sign	177
²	superscript two	178
³	superscript three	179
´	acute accent	180
µ	micro sign	181
¶	pilcrow (paragraph sign)	182
·	middle dot	183
¸	cedilla	184
¹	superscript one	185
º	ordinal indicator', masculine	186
»	angle quotation mark', right	187
¼	fraction one-quarter	188
½	fraction one-half	189
¾	fraction three-quarters	190
¿	inverted question mark	191
À	Capital A, grave accent	192
Á	Capital A, acute accent	193
Â	Capital A, circumflex accent	194
Ã	Capital A, tilde	195
Ä	Capital A, dieresis or umlaut mark	196
Å	Capital A, ring	197
Æ	Capital AE diphthong (ligature)	198
Ç	Capital C, cedilla	199
È	Capital E, grave accent	200
É	Capital E, acute accent	201
Ê	Capital E, circumflex accent	202
Ë	Capital E, dieresis or umlaut mark	203
Ì	Capital I, grave accent	204
Í	Capital I, acute accent	205
Î	Capital I, circumflex accent	206
Ï	Capital I, dieresis or umlaut mark	207
Ð	Capital Eth, Icelandic	208
Ñ	Capital N, tilde	209
Ò	Capital O, grave accent	210
Ó	Capital O, acute accent	211
Ô	Capital O, circumflex accent	212
Õ	Capital O, tilde	213
Ö	Capital O, dieresis or umlaut mark	214
×	Multiply sign	215
Ø	Capital O, slash	216
Ù	Capital U, grave accent	217
Ú	Capital U, acute accent	218
Û	Capital U, circumflex accent	219
Ü	Capital U, dieresis or umlaut mark	220
Ý	Capital Y, acute accent	221
Þ	Capital THORN, Icelandic	222
ß	Small sharp s, German (sz ligature)	223
à	Small a, grave accent	224
á	Small a, acute accent	225
â	Small a, circumflex accent	226
ã	Small a, tilde	227
ä	Small a, dieresis or umlaut mark	228
å	Small a, ring	229
æ	Small ae diphthong (ligature)	230
ç	Small c, cedilla	231
è	Small e, grave accent	232
é	Small e, acute accent	233
ê	Small e, circumflex accent	234
ë	Small e, dieresis or umlaut mark	235
ì	Small i, grave accent	236
í	Small i, acute accent	237

î	Small i, circumflex accent	238
ï	Small i, dieresis or umlaut mark	239
ð	Small eth, Icelandic	240
ñ	Small n, tilde	241
ò	Small o, grave accent	242
ó	Small o, acute accent	243
ô	Small o, circumflex accent	244
õ	Small o, tilde	245
ö	Small o, dieresis or umlaut mark	246
÷	Divide sign	247

ø	Small o, slash	248
ù	Small u, grave accent	249
ú	Small u, acute accent	250
û	Small u, circumflex accent	251
ü	Small u, dieresis or umlaut mark	252
ý	Small y, acute accent	253
þ	Small thorn, Icelandic	254
ÿ	Small y, dieresis or umlaut mark	255

12. Data Reporter

The data reporter is mostly used to create a copy on paper of the content of tables. It is possible to do more, but this is beyond the scope of this tutorial.

The data reporter is triggered through the Tools item in the menu. Once started you get a screen with a menu and different tabs:

File
- New, create new QRF file
- Open, open an existing QRF file
- Save, save the current QRF file
- Save as, save current QRF file under different name

- Open Layout, open QIF file
- Save Layout, save QIF file

- Print settings
- Print

- End

Report
- Run
- Dump into file

Parameters
- Setup
- Delete

Help
- About

Tabs are:
- Query, shows contents of QRF file
- Layout, shows contents of QIF file
- Result, shows result after execution

When you open an QRF file, the corresponding layout file is also opened.

You can then execute the query. Depending on the query, the possibility exists to specify filters, which will take or skip records accordingly.

This reporter is best only used with the existing files. For the filters, look at related field names in text regarding QIF files.

13. Using language files

Two special command-line tools have been created for handling the translation for HTML (and HTML label) *layout files*: LanToRpt.exe and RptToLan.exe, which can be found on the CD-ROM under the Tools directory. They depend on the usage of the JoPPS HTML tags. LanToRpt.exe uses a language file and applies it to a set of HTML layout templates; RptToLan.exe generates a language file. Whenever a JoPPS tag is encountered such as `<JoPPS id=1001>Overview</JoPPS>` in the example below, the text between `<JoPPS id=1001>` and `</JoPPS>` will be replaced by a corresponding text from the language file respectively put into the language file. More information is available by executing these files with a `-?` parameter.

A language file generated by RptToLan.exe will always be sorted on the text's id. A line beginning with ### contains the filename where the text was found. A line beginning with ## is a comment to LanToRpt.exe. A line beginning with # followed by an id, is followed by 1 or more lines containing the text (an empty line is needed if a text should be empty).

This is the beginning of language file for the English reports in the common directory:

```
JoPPS language-file   TechWIN SOFTWARE 1998(c)
Generated by RptToLan on 6/04/99 17:04:14

###intro.h4
#1
Phone :
#2
Fax :
#3
E-mail:
###intro.h4

###STOP.h4
#4
Creation HTML code interrupted !
###STOP.h4

###outlines_b_project.h4
#1001
Overview
###outlines_b_project.h4

###outlines_b_assembly.h4
#1002
Group
###outlines_b_assembly.h4

###outlines_b_frame.h4
#1003
Group
#1004
Part
###outlines_b_frame.h4

…

…

…
```

14. NEW

14.1. Additional execution levels vent and vent part

`slotname_B_VENT` query performed at start of vent level
`Slotname_E_VENT` query performed on end of vent level
`slotname_B_PART` query performed at start of vent part level
`slotname_E_PART` query performed on end of vent part level

Parameter	***Description***	***Example value***
Vent and Part level related parameters (only available for vent and part level)		
%VENT_TEXT%	code of current vent	
%PART_TEXT%	code of current vent part	

Example:

Query `'ASSEMBLY_PER_PAGE_B_PART'` to report the sketches of the vent parts:

```
defines
;needed by JoPPS.QIF
  NO_COLORS     =0

include JoPPS.QIF
include Piece.QIF
include Attrib.QIF
include Glass.QIF
include Extra.QIF
include Annex.QIF

fields
  DSP_DRAWSASH ="BMPSASH:%PROJECT_TEXT%;%ASSEMBLY_TEXT%;%FRAME_TEXT%;%VENT_TEXT%;%PART_TEXT%;#
                 %ASSEMBLY_PER_PAGE_WIDTH%;%ASSEMBLY_PER_PAGE_HEIGHT%;#
                 %ASSEMBLY_PER_PAGE_MEASURES%;%ASSEMBLY_PER_PAGE_OUTLINECOLOR%;#
                 %ASSEMBLY_PER_PAGE_SCALE%;;;;;;#
                 %ASSEMBLY_PER_PAGE_VIEWPOINT%;%GLOBAL_OUTLINERESOL%;;#
                 %ASSEMBLY_PER_PAGE_SCENARIO%;;%BATCHTYPE%;#
                 %ASSEMBLY_PER_PAGE_REFFRAME%;%ASSEMBLY_PER_PAGE_REFVENT%" DISPLAY AS IMGREF(0,0,0)
  DSP_SCALE    ="SCALE:%PROJECT_TEXT%;%ASSEMBLY_TEXT%;;#
                 %ASSEMBLY_PER_PAGE_WIDTH%;%ASSEMBLY_PER_PAGE_HEIGHT%;#
                 %ASSEMBLY_PER_PAGE_MEASURES%;%ASSEMBLY_PER_PAGE_SCALE%;#
                 %GLOBAL_OUTLINERESOL%;%ASSEMBLY_PER_PAGE_SCENARIO%" DISPLAY AS EXTERNAL
```

The corresponding layout file `'ASSEMBLY_PER_PAGE_B_PART_DRW'` is as follows:

```
<!--assembly_per_page_b_part_drw-->
<!—Group per page: schets------------------------------------------------------------>
<br>
%IF{%ASSEMBLY_PER_PAGE_OUTLINES%,<table width=100% border=0>
<tr valign="top">
 <td>
  <a href="EDITOR:%ASSEMBLYATOM%">%DSP_DRAWSASH%%IF{%ASSEMBLY_PER_PAGE_SCALE%,<br>%DSP_SCALE%,}</a>
 </td>
</tr>
</table>
```

Important: These additional execution levels are available from JoPPS version **3.35 P3** and not provided in the standard JoPPS reporting.