Sage ERP X3

SETTING UP ZPL PRINTING

Sage ERP X3 V6 – November 2011

Contents

1. INTRODUCTION

Printing labels using the Crystal Reports runtime shipped with the Sage ERP X3 is relatively costly in terms of performance, and results in labels that are not necessarily perfectly readable when they include bar codes (15% readability in some cases for complex bar coding).

The setup of ZPL printing (Zebra printers) in X3 has been implemented in Sage ERP X3 V6 to cater for this issue.

This document describes the step-by-step set up of ZPL printing in X3. It does not cover Zebra programming as it is assumed that this is mastered by the consultants implementing this solution on customer sites.

2. PRINCIPLES

2.1. Pattern Files

Zebra printers are controlled using a language called ZPL (*Zebra Programming Language*). It is the original Zebra page description language that allows the definition of fields and their placement on a label, using a simple instructions or more complex masks.

A report written in ZPL is called a **pattern**.

When a pattern file or stream is processed, pattern fields (tags) are replaced with the actual values (data) before the report is sent to the printer.

2.2. ZPL Printing in X3

ZPL printing in X3 relies on the existing report dictionary function, and on printing destinations, a standard feature of X3.

A report may be flagged as ZPL in which case if the destination used is flagged as ZPL as well, it will process a **ZPL Report** instead of a regular Crystal Report.

Features of the ZPL Report (Common data > Products > Products):

- A ZPL Report is attached to the report dictionary (X3 report) via the **ZPL Printer** tick box.
- It shares the same parameter definition features as a regular X3 report (**Parameters** tab)
- It processes the **ZPL pattern file** stored on the server by computing variables and expressions using an X3 **Data Model** with selection criteria.
- It uses calculated expressions with prefixed names that may be listed in the report definition along with conditions

3. A QUICK GUIDE TO SETTING UP ZPL

3.1. Activating ZPL Printing

ZPL printing depends on the **AZPL** activity code.

- In your application folder, go to **Development > Development Setup > Activity Codes** and make sure the **AZPL** activity code is active.
- Go to Development > Utilities > Dictionary > Validations > Dictionary and validate all elements linked to the AZPL activity code.
- Log out and back into your application folder. Verify that the following menu exists: **Development > Processes > ZPL Reports**.
  If it does not, go to the Objects dictionary in the Development > Processes menu and revalidate object **ARZ**. Log out and back into your application folder.
- You should now be able to use ZPL features in X3 as follows.

3.2. The ZPL Pattern File

It is not the object of this document to guide you through ZPL programming. However we will provide some advice and we will present specific tags to be used for communicating with X3.

3.2.1. Constructing the pattern file

The pattern file is written in ZPL. You can use ZPL programming interfaces such as **Teklink's CODESOFT** package. Crystal Reports developers will be familiar with this drag and drop interface. **Zebra** also provide their own ZPL programming tools.

Simple parsed patterns contain header, data and footer sections. The data section is repeated for as many times as there are labels to print when the stream (or file) is sent to the printer:

```
^XA
^MCY
^XZ
^XA
^FWN^CFD,24^PW1535^LHO,0
^CIO^PR6^MNM^MTT^MMT^MDO^PON^PMN^LRN
^XZ
^XA
^DFR:TEMP_FMT.ZPL
^LRN
^XZ
^XA
^XFR:TEMP_FMT.ZPL
^AON,42,42^FO224,203^FVX3F_[F:ARM)CLEA3^FS
^AON,42,42^FO224,251^FVX3F_[F:ITM]ITMDES1^FS
^AON,42,42^FO224,300^FVX3F_[F:ARM]CLEA4^FS
^AON,42,42^FO224,157^FVX3E_1^FS
^BY2^FO49,46^BCB,124,N,N,N^FV>:X3E_1^FS
^AOB,37,32^FFO177,97^FVX3E_1^FS
^PQ1,0,1,Y
^XZ
^XA
IDR:TEMP_FMT.ZPL
^XZ
```

Fig. 1 – A sample ZPL pattern file with Sage ERP X3 tags

Masked patterns contain a mask definition followed by a section with variables to be placed in the mask. The difference is that masks can be stored in the printer memory, thus avoiding unnecessary network traffic.

Both pattern types are supported with Sage ERP X3.

3.2.2. Tags to use with Sage ERP X3

You may insert special tags in the pattern file that will be processed by the system and replaced with actual data from the X3 tables. More details will be given on the use of tags and their definition in Sage ERP X3 in the next section of this document.

Construct your ZPL pattern file (using masks or simple parsed patterns) and store it in the **PAT** directory of your X3 folder on the application server **Folders** directory, with a name and a **.pat** extension, for example **LABEL.PAT**.

Make sure the **PAT e**xtension is uppercase.

If the **PAT** directory does not exist, simply create it in your application folder's directory.

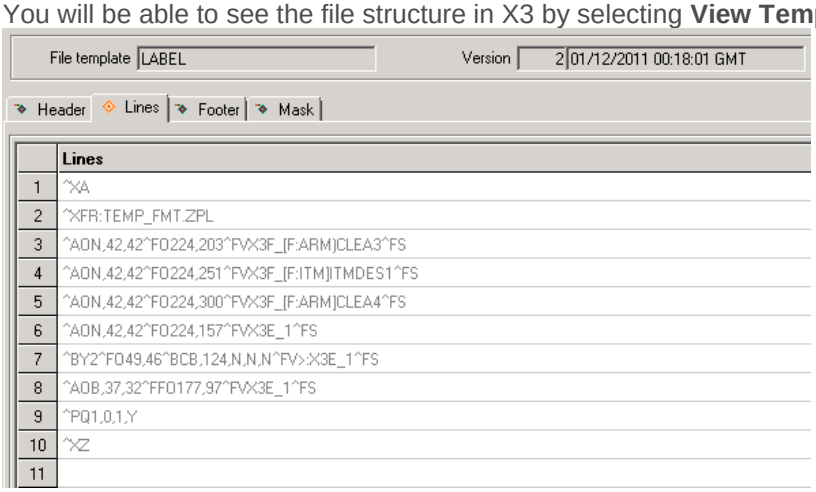
3.3. ZPL Report Definition

Development > Processes > ZPL Reports

3.3.1. Loading the pattern file

You will define your ZPL report in the **ZPL Reports** function: Press the **New** button, give your report a code (e.g. LABEL) and proceed to the **Template File** field.

- Select the file you placed in the **PAT** directory using the right-click **Selection** option on that field.
- You will be able to see the file structure in X3 by selecting **View Template** on the right-click menu:



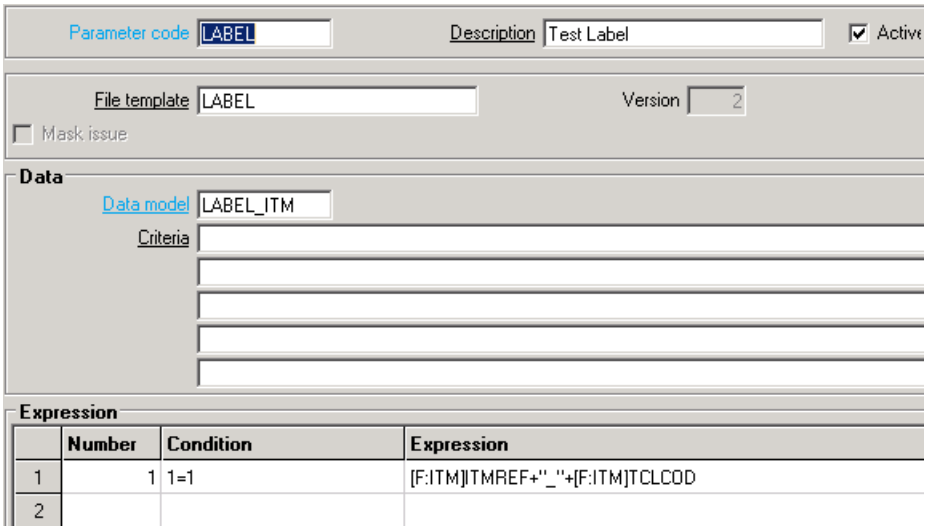
3.3.2. Defining calculated expressions

There are two types of special tags that may be used in the pattern file to instruct X3 to evaluate them and replace them with actual values at execution time:

Tag	Processed as…
X3E_**n**	Expression **n** in the table of expressions in the report definition (see below)
X3F_[F:XXX]YYY	Table field (XXX = Table abbreviation; YYY = Field code) – e.g. **X3F_[F:ITM]ITMREF**

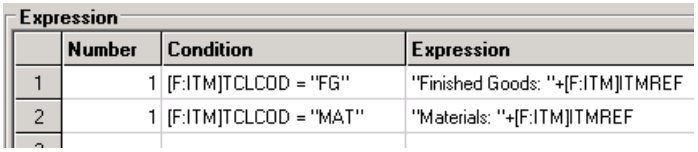
As you can see in the pattern file example in the previous section, the data section (**Lines**) in the pattern file contains four Sage ERP X3 tags:

- **X3F_[F:ARM]CLEA3**: Evaluated as the value of field CLEA3 from table AREPORTM ([F:ARM])
- **X3F_[F:ITM]ITMDES**: Evaluated as the product designation from table ITMMASTER ([F:ITM])
- **X3F_[F:ARM]CLEA4**: Evaluated as the value of field CLEA4 from table AREPORTM ([F:ARM])
- **X3E_1**: This is an expression that has to be defined in the report, as follows:



The number following the underscore character in the pattern file tag **X3E_1** points in this example to the expression numbered **1** on the report definition. **X3E_2** would point to the expression numbered **2**, and so on.

The **Condition** column allows the definition of condition criteria (expressions) that, if false, will prevent the line from being taken into account in the report. This allows the creation of alternatives, such as the one below:



In this case if the product category (TCLCOD) is "FG", the value replacing **X3E_1** will be "Finished Goods:" followed by the product code. Otherwise the **X3E_1** code will be prefixed with "Materials". Note that both lines are numbered **1**.

3.3.3. Data Models

Sage ERP X3 **Data Models** (**Development > Data and Parameters > Development Setup > Data Models**) can be used to make sure all tables are opened and read correctly.

The Data Model is populated in the **Data** block on the ZPL Report definition and may be associated with filtering criteria (expressions from any field in the linked tables from the Data Model)

3.3.4. Updating the pattern file

You can update the pattern file at any point in time by replacing the file on the server (**PAT** directory) and selecting **New template version** from the **Template file** field right click options.

When you are done constructing your ZPL report definition do not forget to validate the report using the **Validation** button.

3.4. The Report Dictionary: Running the ZPL Report

Development > Reports > Reports

You now need to link your ZPL report to a regular Sage ERP X3 report definition. Here is how:

- Construct your report definition as you would do a normal Crystal Reports. You may include report parameters as usual.
- Tick ☑ **ZPL Printer** in the **Output characteristics** block.
- Populate the **Parameter Code** with the ZPL report definition code you created from section 3.3.
- In the **Destination** field, you may enter a destination for which you have selected **ZPL Printer** as the printer type.

Note: It may still be interesting to add a Crystal Report file to the ZPL report definition. When the chosen printing destination is not ZPL, the Crystal Report file will be run instead of the ZPL pattern. You may use this feature for testing.