Experiences with XML Publisher

Frank Rossiter
Senior Consultant – Xelocity Limited

Abstract

XML Publisher is relatively new part of the Applications middleware that allows all existing reports to be transformed into high fidelity reports with some set-up but with no programming required.

The business issue that Oracle has solved was the increasing expense of maintaining reports to meet diverse customer requirements. This was a large cost to both the customer who had to maintain customised reports and Oracle who had to maintain multiple formats of reports in multiple languages.

Oracle have reduced cost and increased usability by separating the source of the data from the formatting and the language. This is an example of Oracle using industry standard tools to great benefit for both themselves and their customers.

This paper will introduce XML Publisher, describe experiences using it to produce a reformatted Packing Slip and discuss other uses.

Setup required

Configure reports to produce XML in Define Concurrent Program

All existing reports in Oracle Applications can be configured to produce their output in XML. This is a native feature of the underlying Oracle Reports 6i technology. Selecting XML output format turns the report into a source of structured data without the complication of formatting.

The first step is producing the report in text format and then changing the report configuration to produce XML format output. Comparing the two outputs identifies the XML tag names used for the required part of the report.

One point worth mentioning is that often the tag names chosen by the developers are rather obscure and that the XML structure often have redundant hierarchies. Another point is that tags appear in the XML output that are hidden in the text output.

Create a template

Microsoft Word is used to create the template. Tables, shading, borders, free text and images are used to define the layout of the document.

The XML field names are then inserted to show where the report information should be placed. Finally control codes are inserted to control looping around certain sections of the report and control sorting or other special requirements.

When the original XML packing slip was created the codes had to be typed in manually. At the end of August 2005 Oracle released the XML Publisher Desktop (patch 4561835). This contains an add-in for Microsoft Word. As well it also installs a copy of the XML publisher engine that will run locally on the PC for testing purposes. This word add-in greatly simplifies the addition of the field and control codes and is strongly recommended.

Define a Data definition / Template

The XML Publisher Administrator responsibility is first used to define a data definition. This defines a report as a data source in XML Publisher and is the link between the standard report in applications and the XML template.

The template defines the various formats of the report that are available for each report. For example different territories could have different format invoices.

Setting the default Template

The setup of the concurrent program is changed to use the new template by default. The Update Concurrent Program window (available from the System Administration Responsibility) is used to set the default template in the Onsite Setting tab. Now by default every time the report is run the XML publisher version will be produced not the text version. The Template field is not available from the Forms-based Concurrent Programs window.

Producing a Packing Slip

Business Requirements

As part of the shipping process when goods are despatched to customers a report called the packing slip is normally produced. The packing slip goes with the shipment and is given to the customer with the delivery. The customer's inwards goods staff uses the packing slip to reconcile to the customers order and to check that the delivery is correct.

The packing slip lists the customer's details, goods ordered and goods delivered as well as other details. The standard packing slip as supplied by Oracle prints six lines per delivery line. It caters for all the possible delivery options including packaged items, lots, and item with secondary units of measure.

As part of the original implementation the packing slip was customized to remove superfluous information and only print one line per delivery line. As business requirements have changed further customisations have been done in one case reversing early changes.

Based on feedback from their customers two further changes were required. Some customers who regularly receive large orders had complained that it was difficult to match the packing slip back to their original purchase order. This difficulty was causing delays to receipt the goods because:

- The packing slip is sorted by Inventory Item Number and is not sorted the same as the customer's purchase order. Hence with orders of dozens of lines reconciliation is difficult.

- If there is no stock of an item available to ship these items are recorded at the end of the packing slip not in the main body of the report

Using Oracle Reports 6i

Oracle wrote the standard packing slip in Oracle 6i. Customising a complex report such as the packing slip is often not a task developer's relish. Changing the sort order was expected to be relatively straightforward. However combing the two sections of the delivered goods and the backordered goods was expected to be quite complex.

Using XML Publisher

Creating the template in MS Word proved to be even simpler that originally thought. The sort order was set by simply adding a sort token.

Combining the two sections was straight forward as well. This was done with the "for-each-group" command and the pipe command to join the G_C_INV_ITEM_ID and G_BO_UNSHIPPED_QTY nodes group-by line numbers. They were also grouped by sales order lines number.

Nested tables were found to be the easiest way of producing the desired layout. Hence there was a master table with the header in one cell and the body in the other. The body itself was made up of three other nested tables. One for lotted items, one for non lotted items and one for lines that were not delivered at all.

Leanings and Limitations

A few limitations and leanings were discovered during this work.

One limitation is that only information that is in the original report can be used. For example the backordered lines only have an item number and no item description. This is because there was no description in the original report. In this implementation the description was report was modified further. It was easy to add more information without having to worry about the grouping or layout. Alternatives that may have been used if the report had not be modified would include combining the output of two reports from separate concurrent requests (the packing skip and the order report). While XML supports combining documents a practical method of implementing this with XML Publisher is not known. The packing slip could have been combined with an extract of the item details to a fixed location.

Another issue is that while all reports can be output in XML format, the Oracle developers probably did not expect that XML would be produced. Hence many of the tag names are not particularly helpful and often the grouping is more complex than desirable.

If an incorrect code is used in the template the error message is not very useful. Creating debugging information is somewhat tedious. This has been improved slightly in the XML publisher desktop but could still be more helpful.

As well changing the parameters of the report changed the structure of the XML output.

Other Current Uses

Producing the packing slip only illustrates some of the capabilities of XML Publisher. By underlying industry standards produce a rich and powerful set of tools. Soe of the uses are listed below.

XML publisher can be used to produce bank Electronic Transfer Files (EFT) or EDI files. To do this a special type of template called eText templates are used.

XML publisher can be used to format Financial Statement Generator reports to automatically create a financial pack for monthly or yearly reporting. XML Publisher can automatically generate graphs as part of the pack. Alternatively XML Publisher can also produce a formatted Excel document to speed up financial reporting.

Government forms are often supplied as PDF files. XML Publisher will accept a PDF template instead of a RTF template. The advantage is that if that as XML Publisher maps to the named PDF fields. Hence if the layout changes in the following years the mapping will still be valid.

Some of the most recent Family patchsets have include XML Publisher as standard. Modules include Oracle Incentive Compensation, Oracle Service Contracts, Oracle Advanced Collections, About Oracle Product Lifecycle Management, About Oracle General Ledger, About Oracle Time and Labour, About Oracle iPayment, About Interaction Center Intelligence, Oracle Purchasing (Note:295914.1) and Oracle payables.

Some of there modules (eg Oracle Advanced Collections) are using the Delivery Manager functionality to automatically Email, Fax or FTP the complete report to the intended recipient after it is produced.

Future

XML Publisher will form the basis for future reporting in Oracle applications. By separating sourcing the data from formatting the data much of the need for customisations will be removed. As mentioned above Oracle developers are already finding it easier to use XML Publisher for producing custom reports. The ability to use the delivery manager to eMail or fax the result increases the usability of the reports.

Oracle have already announced that the next versions of Peoplesoft Enterprise and JDE EnterpriseOne will have XML Publisher available as one of the first steps in Project Fusion. Both these products already can produce XML outputs. Oracle will just need to integrate with the different report managers used in those products.

Glossary

eXtensible Markup Language. (XML)

This is a more structured version of HTML. This is the source of the data for the reports.

XSL Transformations (XSLT)

a language for transforming XML. These commands can be used to rearrange the structure of the data.

XSL Formatting Objects (XSL-FO)

An XML vocabulary for specifying formatting semantics. This defines the layout of the reports.

XML Path Language (XPath)

an expression language used by XSLT to access or refer to parts of an XML document.

Pasta

Printing technology used by Oracle to view and print a variety of formats of files including PDF.

Node

Data in XML documents are organised into a tree structure. Each branch of the tree is refered to as a node. Each node (apart from the root node) has a parent and may have children.