



XML Database

Lesley Smith, Samuel Mann Matthew Hamilton

Otago Polytechnic
Dunedin, New Zealand

lsmith@tekotago.ac.nz

Proceedings of the 15th Annual NACCQ, Hamilton New Zealand July, 2002 www.naccq.ac.nz

As part of the requirements for two papers in the BIT, (SE205 - Software Engineering, and DB309 - Databases 3), students are required to read selected research papers, then to present the material to the class. It is useful to capture these presentations in a format which enables other students to follow up topics of interest.

In 2000 the students were required to submit their readings using an html template. The pages were added to the class website and could be accessed through a linked list. While useful, this required lecturer input to get the pages published quickly, and formatting varied widely among students.

Class discussions on the role of XML for data storage led to the development in 2001 of an 'XML database'. A combination of Cold Fusion web pages, XML documents and an Access2000 database captured the readings information and displayed it dynamically from the class webpage. This served two purposes. It freed the lecturer from the responsibility of updating the pages. It also exposed students to a view of the XML document structure, both as they entered their reading into an XML template within a form field, and as it displayed in the browser. ColdFusion tags were used to save the XML document into a single field in the database.

Unfortunately this solution led to frustration from the students. The XML document structure is very exacting, and errors in formatting could cause the data input to fail.

In 2002 the Database3 class was required to develop a web based database as a major assignment. One student had an interest in XML and suggested the redevelopment of the

readings database in a more robust and attractive format.

The requirements were to

- ◆ Provide a structure for document storage
- ◆ Demonstrate XML in use
- ◆ Create a suitable user entry interface
- ◆ Allow document browse and search.

This posed some interesting implementation issues as many XML technologies are still under development. Options for XML document storage currently include:

- ◆ XML document (append each new addition to a single XML document)
- ◆ Blob (stored in Access field)
- ◆ Native XML database
- ◆ Add-on to relational databases
- ◆ Object database.

We first explored the use of a single XML document, but realised it would quickly become unmanageable. As native XML databases are still an immature software, we decided to implement using a BLOB field in the database. To enable searching on the XML document keywords were extracted from the information within the ColdFusion page as it was submitted.

The final result is an attractive interface with a "wizard" input form which the students can use to input text and upload images or files. The readings or complete articles are displayed using XSLT style sheets or as unformatted XML using the Microsoft MSXML Parser. The database has been a valuable learning and teaching experience for the whole class.

