Rethinking the Database

Using XML databases to align with business processes and enhance capabilities

Presented by the Center for Technology in Government and the NYS Department of Taxation and Finance

Components of XML

- XML - encoding documents electronically
- XSL - transforming and rendering XML documents
- XPath - addressing the parts of an XML document
- XML Database – storing collections of XML data
- XQuery - querying collections of XML data
- SQL/XML - querying XML within SQL
- XForms - interfacing with XML data
- XRX - coming soon to a database near you

XML for Data Exchange

- Data exchange - XML enables platform-independent data exchange among applications.
- XML serves to "glue together" or mediate a common data layer between two separate and already existing programs.
- Typically, XML-based messaging services (such as SOAP) enable different applications to communicate.

XML for Content Management

- Web content management - usually implemented as a Web application, for creating and managing HTML content.
- Most systems use a database to store content. Content is frequently, but not universally, stored as XML, to facilitate reuse and enable flexible presentation options.
- A presentation layer displays the content to regular Web-site visitors based on a set of templates. The templates are sometimes XSL files.

XML for Syndication: RSS

- RSS is a simple XML format used to syndicate headlines.
- It is used by websites that publish new content regularly and provides a list of headlines with links to their latest content.
- Content such as news feeds, events listings, project updates, and most recently podcasting, video and image distribution can all be distributed by RSS.

Databases: Relational vs. XML

- Relational – no hierarchy or significant order; based on two-dimensional tables. Used for storing and querying data.
- XML – hierarchical and sequential; based on trees in which order matters. Used for exchanging and displaying data.

A Complicated Relationship

- Data exchange between XML, applications, and databases is not a simple, one-step event.
- It involves many processing steps and translations of the data into totally different formats.

Simplifying the Relationship

- Storing data as XML (in native XML or XML-enabled databases) eliminates the process of translating data back-and-forth into various formats.
- Data is received, stored, and processed as XML.
- Eliminates multiple translation steps (along with their development times and their possibilities for errors).

The Real Difference

"XML is by definition self-describing data. Build the database around that structure not the other way around. The implementation is far from being that simplistic. This basic concept however – leverage XML's self-describing and hierarchical nature to manage it – is the very foundation of an XML database." (from http://bigmenoncontent.com/2009/05/28/xdb-matters/)

Rethinking the Database

Using XML databases to align with business processes and enhance capabilities

Presented by the Center for Technology in Government and the NYS Department of Taxation and Finance

Two Immediate Benefits

By storing XML data as XML in XML database:

- Simplified storage
- New query capabilities

Resources

XML Databases - The Business Case
http://www.cfoster.net/articles/xmldb-business-case/
Ronald Bourret, Consulting, writing, and research in XML and databases
http://www.rpbourret.com/xml/
Introduction to Native XML Databases
http://www.xml.com/pub/a/2001/10/31/nativexmldb.html
A comparison of XML-enabled and native XML data management techniques
http://xml.sys-con.com/node/104980?page=0,0
Feature Comparison: EMC Documentum xDB vs. Oracle XML DB & IBM DB2 pureXML
https://community.emc.com/docs/DOC-2999

Contact

Jim Costello
Center for Technology in Government (CTG)
jcostello@ctg.albany.edu
518-442-3812

For more info and training opportunities on XML and XML databases as part of CTG's Advanced Web Technologies Initiative, visit http://www.ctg.albany.edu/awt/

Facilitating Business Alignment

- Sharing the same Business defined business object (XML)
- Eliminates Technical Abstraction. All processes, functions and reports "configured" in user's terms.
- The same core Business Object is leveraged by all of the system components .

Returns Processing

Operational XML – Next Generation DB Design

Relational Table by Form

- 3600 tables required
- Difficult to get filing context
- Made Rules engine, display difficult
- Much IO

Generalized Relational Table

- Needed DB to translate fields
- Sparsely populated
- Performance issues
- Rules engine limitatations

XML Solution

- Business object based (the audit folder)
- Keeps business context
- Robust rules processing
- Can leverage XML tooling

Transactional XML Layering

- Exceptions
  - Quality or Condition of Data
  - Auto Routing

  - Customer Service
- History
  - What, Why and Who of all Changes
  - Auditability
  - Exception Resolution
- All relevant data in one place
  - Less I/O
  - Data Integrity
  - Enable Transaction Processing

Development Enablement
Establishing Patterns
- External Channels → XML → Processing

Web Development challenges
- Develop quicker
- Reuse "segments" of web apps
- Consistent features (print, return to application)
- Consistently defined navigation patterns
- Track users usage of application

What is a Web Application?
A series of form based UI objects that create a transaction for processing.
- Then each of these "forms" can be designed exactly the same way….
- A Form . . . has a UI object . . . has an object enforcing form rules . . . has an XML segment in the transaction XML
- In this way pages can be coded separately and following the same pattern and integrated into a web application.

Web Navigation Pattern
Web Navigation highlights
- All pages coded exactly the same
- Single XML table allows restart
- Navigation patterns enforced (Wizard, conditional, etc.)
- Process server allows for externalization of navigation
- Common error handling
- Can use previous filing to start new transaction

Web Service Pattern
Web Service Process highlights
- Same pattern serves all web services
- Leverages same rules as web
- Once web is established no additional coding required

XML within CM
Receive XML and show the document in the form (integrated with our EDMS)
- user view is independent of channel!
- all data received is stored in one table
- form can be used as input and for correction
- PureXML solution uses advanced indexing

Paper Process Pattern
Paper Process highlights
- Same pattern serves all paper processes
- Leverages the same form Interfaces

- Leverages same XML database
- Leverages same rules as web
- Has built in capabilities to have different acceptance rules for paper

XML Indexing

XML indexing highlights

- Allows for a many to one indexing scheme
- Index fields can be added on fly
- Supports optional index

Legacy Integration

Legacy Integration highlights

- Allows for modernizing data capture while keeping legacy processing
- Single XML table stores all received documents and integrated with CM
- Many mapping services are re-usable

Process Automation Pattern

Process Automation highlights

- Can be integrated with UI for complete inline processing
- Single XML table stores all received documents and integrated with CM
- Process can be reported on through Monitor
- Operational data system becomes XML

NY State Tax XML/SOA Processing

- PIT (Personal Income Tax)
  - 11M returns processed
  - Peak in April: 390,000+ per day
  - Up to 14,500 different data elements for the filings (60% Electronic)
  - 6M Refunds ($4.9 B), direct deposit up 13.1%, checks down 6%
  - Electronic extensions up 160% (439,000)
- Corporate Tax
  - IRS ELF Program – 2007 - 32,317, 2008 -193,977
  - Peak month: 100,000 returns in April 2009
  - Peak day: 20,000 returns
- Sales Tax
  - 1Q2009 : 60,000 on the Web
  - 1Q2009 : 400,000 from partners
- Withholdings Tax
  - 50,000 web filings of XML
- STAR Property Tax Rebate Application (Tax Refund)
  - 3.5 web applications in a 3 month period

NYS Tax at Direction

- Convert other Subsystems (Domains) to XML
  - Simplify conversion
  - Map data structures closer to the business
  - Leverage the rules engine
- Expand the use of web navigation with integration into operational XML
- Incorporate more XML enabled tools to speed delivery and improve product
- Leverage the XML data in new ways (AJAX, REST, RIA)

Contact

Jim Lieb
NYS Department of Taxation and Finance
james_lieb@tax.state.ny.us