Optimizing Industry-Standard XML Schemas with Oracle XML DB

Sam Idicula - Oracle
Sriram Krishnamurthy - Oracle
Steve Levine - UBmatrix
The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle’s products remains at the sole discretion of Oracle.
Agenda

• Industry Schema and Storages
• XBRL and Regulators
• XBRL Storage and XML DB
• XBRL Processing
• Oracle XMLDB and UBMatrix Demo
Industry Schema and Storages
Outline

• XML DB Schema Capabilities
• Complexity of industry XML Schemas
• 11gR2 Schema Enhancements
• Use Cases & Storage Guidelines
• Storage examples: FpML and OOXML
Why XML?

- Open, vendor-neutral standards, driven by W3C
  - XML, XMLSchema, XQuery, XSLT, DOM etc
  - Standard well-understood APIs available for most common development environments
- Flexible and verifiable data model
  - XML Schema allows data validation
  - Simplifies data exchange between loosely connected applications
- Applicable to a wide variety of applications
XML Schema in Oracle XML DB

- Validation of instance documents
- Object Relational storage model derived from XML Schema
  - SQL Object Types generated from type model defined by the XML Schema.
  - XML data persisted in object-relational tables.
- Binary XML Storage: Serialized post-parse representation
  - Uses XML Schema to improve storage/query efficiency
  - Allows schema-less storage as well
- XML Index use XML Schema to improve query optimization
Widespread usage of XML & Schema

- XBRL: Financial and Regulatory reporting
- FPML, FixML, Acord: Financial Services
- NIEM: Law Enforcement and Public Safety
- HL7: Healthcare
- OpenXML, ODF: Desktop Applications
- DICOM, EXIF: for Digital Imaging
- RSS: publishing / syndication of content
- DITA, DocBook: Technical Publishing
- OpenGIS, KML: Spatial applications
Under the hood…

• **NIEM: National Information Exchange Model**
  - 115 base schemas and several extensions
  - Over 1600 substitution group elements
  - Over 2000 subtypes
  - Deep type hierarchy
  - Complex interdependencies between schemas

• **HL7 CDA: Healthcare - Clinical Doc Architecture**
  - Over 100 schemas
  - Large number of substitution group elements
  - Large number of subtypes
  - Mutually recursive dependencies between schemas
Schema Registration Tips

• Factors that indicate complex schema sets
  – Recursive dependencies between multiple schemas
  – Deep type hierarchy and/or large types
  – Large number of subtypes for a particular complex type
  – Large number of elements in a single substitution group

• Tips for registration
  – Set aside sufficient shared_pool memory
    • Examples: NIEM needs about 1024M
  – Register different schemas or sets of schemas in different PL/SQL blocks if possible
  – If generating tables, set xdb:defaultTable=“” as appropriate
  – For O-R: break up large types using xdb:SQLInline=“false”
11gR2 Schema Enhancements

- Major improvements in both memory & time
- Much faster Registration and loading of Schemas
  - Speedup is higher for more complex schema sets
  - Upto 200x in some cases
- 2-3x reduction in shared memory usage
- Much less PGA usage during schema registration
- Streaming Schema Validator Cache for Binary XML
  - Validation
  - DML: Insert & partial update
  - Significant improvements for small documents
11gR2 Performance

- Registration: 10.5x improvement
- Validation: 6.4x improvement
- Insert/Load: 4.1x and 4x improvement
- Update: 7.9x improvement
Types of use cases: Data Model

**Structured**
- “Data Centric”
- Static XML Schema
- Limited Variability
- No “any” or “mixed”

**Semi Structured**
- Complex XML Schema Collections
- Volatile XML Schemas
- Islands of “any”
- Islands of Structure

**Unstructured**
- “Document Centric”
- No XML Schema
- Very flexible XML Schema
- Repeating Choice, “any” and “mixed”
Types of use cases: Application characteristics

• **Query & DML Characteristics:**
  - **Structured**
    - Typically simple XPaths and relational-like access
    - Updates of values typically
  - **Semi-structured & Document-centric:**
    - Complex XQuery constructs, highly variable XPaths
    - Updates could be large, involving mixed text

• **Document Retrieval**
  - **Structured:** Less common
  - **Semi-structured:** Fragment retrieval is quite common
  - **Document-Centric:** Both full-document and fragment retrieval are very common
## Rough Storage Guideline

<table>
<thead>
<tr>
<th>Use Case Type</th>
<th>Storage</th>
<th>Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>Structured</td>
<td>Object Relational Storage</td>
<td>BTree Indexes on columns</td>
</tr>
<tr>
<td></td>
<td>(e.g. employee record in XML)</td>
<td></td>
</tr>
<tr>
<td>Semi-structured</td>
<td>Binary XML Storage</td>
<td>Structured/hybrid XMLIndex</td>
</tr>
<tr>
<td></td>
<td>(e.g. HL7 CDA Patient Record)</td>
<td></td>
</tr>
<tr>
<td>Document-Centric</td>
<td>Binary XML Storage</td>
<td>Unstructured/hybrid XMLIndex + Text Index</td>
</tr>
<tr>
<td></td>
<td>(e.g. Functional Specification with author, date, title fields)</td>
<td></td>
</tr>
</tbody>
</table>
Storage Example: FpML

- **Standard for financial derivatives information exchange & trading**
- **Medium complexity**
  - 20 schemas, around 240 subtypes
- **Highly structured, low variability**
  - Very few `<any>`’s
  - Small number of substitution groups
  - Very little mixed text
- **Queries are typically specific & relational-like**
  - Can be rewritten well to O-R
- **O-R storage with indexes on nested table columns**
Storage Example: Office Open XML

• Developed by Microsoft as a successor to binary MS Office formats
• Standardization driven by Ecma International
• Uses Open Packaging Convention-based file package
  – Individual files contain text, images, charts etc
  – Relationships between these expressed in XML format in separate file
• XML Schemas for different types of docs
  – Primarily WordprocessingML, SpreadsheetML, PresentationML, DrawingML
• Medium complexity: 80+ schemas
OOXML (contd)

- High variability in content
  - Usage of `<any>`, `<choice>` etc
  - User-embedded XML represented using `<customXML>`
- Document-oriented content
  - Mixed Text
- Queries
  - Highly variable, complex XPaths
- Binary XML Storage with unstructured XMLIndex and Oracle Text Index
- Demo:
  - Exploiting XML structure of MS Office docs for better information retrieval and management
  - Demo booth: Moscone West W-015
XBRL and Regulators
What is XBRL?

- eXtensible Business Reporting Language
- Open standard based on XML
- Supported by XBRL.org and over 550 organizations worldwide
- Standardizes how to exchange financial information
  - Semantics
  - Validation
  - Extensibility
  - Business rules

“…will make it easier to generate, validate, aggregate, and analyze business and financial information which in turn will improve the quality, timeliness, completeness, and comparability of the information that companies use to make decisions”

2007 Breakthrough Ideas
• The XBRL taxonomy defines:
  – **Concepts**: what we can report about (XSD schema)
  – **Presentation semantics**: to aid the data modeler and to some extent guide report layout
  – **Calculation semantics**: what summations should hold in the reported data
  – **Reference semantics**: ‘provenance’ of the concept definition
  – **Labels**: support for multi-lingual labels of many different roles
  – **Dimensional structures**: multidimensional cubes, hierarchies
  – **Formulas**: to compute values or use computations in validations

• The XBRL instance document includes
  – **Concept**: Which item
  – **Context**: e.g. Time: What time period? Instant? Or Budget vs. Actual
  – **Unit**: Dollars? Euros? Shares?
• XBRL Taxonomy
  - Dictionary of Concepts, Labels, Calculations, and Instructions
  - Instructions are encoded as rules and can be validated

**Concept = Cash, Cash Equivalents & Short Term investments**
XBRL Primer

Key Concept – Instance Document

- XBRL Instance Document
  - Contains Concepts, Contexts & Fact Values only
  - Easy to exchange and consume

Concept = Cash, Cash Equivalents & Short Term investments

Fact Value = 21574000000

Context = FY08Q1e
Regulators Across the Globe Use XBRL

**EUROPE**
- Belgium - CBFA & National Bank
- EU - CESR, CEBS
- France - Bank de France, AMF
- Netherlands - 4 major ministries & Water Boards
- Norway - Exchange
- Spain - Bank of Spain, & CNMV, Sweden - Companies House
- UK - HMRC (Companies House)
- Italy - Stock Exchange and Banking

**MIDDLE EAST/AFRICA**
- Abu Dhabi - Exchange
- Israel - Securities Authority
- South Africa - Exchange

**ASIA/PACIFIC**
- Australia - Government Wide
- China - Shanghai & Shenzhen Exchanges, CSRC, SFC
- India - Mumbai & India National Exchange, Bank of India
- Japan - Tokyo Exchange & Bank of Japan
- Korea - KOSDAQ
- Singapore - ACRA
- Thailand - Thailand Exchange

**NORTH AMERICA**
- Canada - CSA, Toronto Exchange
- Cayman Islands - CIMA
- US - FDIC, SEC, IRS, Dept Parks
- World Bank - Micro Lending

**SOUTH AMERICA**
- Argentina - Bank of Argentina
- Bolivia - Government-wide
- Brazil - Bank of Brazil
- Chile - Bank of Chile
- Columbia - Bank of Columbia
- Peru - Bank of Peru

SEC Mandate Dec 15, 2008
XBRL Use case 1: **Regulators / Hubs**
(Analytics, Publishing over Reports/Taxonomy Submissions)

- Taxonomy Design
- Taxonomy Authoring
- Internal Publishing
- XBRL Storage
- Scalable Repository
- Taxonomy Publishing
- Validation & Acceptance
- External Publishing
- Scalable XBRL Engine
- Regulated Entity
- BI Engine
- Publishing Engine

Reports, Tx Extensions for all entities
XBRL Use case 2: Regulated / Enterprises
(Aggregation, Submission, Publishing of Subsidiary Instance/Taxonomies)
XBRL Storage On XML DB
XBRL Technical Challenges

- **XBRL has added challenges**
  - **XBRL Processing Engine**
    - *Process content based on business rules, XML engine not enough*
  - **Native XML Storage with security, ILM**
    - *File system based solutions inadequate with growing volumes*
  - **Maintaining Taxonomy Document Integrity**
    - *Managed individually by XBRL applications today*
  - **Queryability over large volumes of documents**
    - *Need efficient Queryability, Analytics over both instances and taxonomies*
  - **Dynamic reconstruction of hierarchical relationships**
    - *Hierarchical relationships not explicit and not static, need to be reconstructed from linkbases prior to any useful operations*
  - **Scalable Lookups of Schemas and Linkbases**
    - *Taxonomies large, numerous, extensible. Need scalable lookups when operating on large volumes*
    - *USGAAP 2009 contains around 211 schema and 460 linkbase files with 14000 concepts.*
Oracle XML DB with XBRL support

Submitted Reports

- Store reports as submitted in XML
- Maintain consistency and integrity

XBRL Repository

- Queryability and Analytics over content
- View Reports as filed
- Scalable XBRL Services

Taxonomies

Maintain consistency and integrity

View Reports as filed

Scalable XBRL Services

Queryability and Analytics over content

Store reports as submitted in XML
Oracle-UBMatrix End to End Solution

Technical Highlights

Full XBRL Processing Capabilities
Integrated XBRL Processing Engine

Scalable Query and Services
Minimize Loading for Taxonomies & Instances into Memory.

XBRL Storage
Provides storage and management of XBRL content including doc integrity

XBRL Repository

Protocols, SQL, XQuery

XBRL Services

XBRL Query

Instance Views
Taxonomy Views

Instance Documents
Taxonomies

Validation
Discovery
Rendering

Desktop Tools
XBRL Enhanced Services with XDB

• **XBRL Storage**
  - Binary XML and Structured XMLIndex solution for fast upload of XBRL documents.
  - Database native XML Persistence for XBRL content
  - Taxonomy Integrity enforcement

• **XBRL Query**
  - XBRL Dictionary
    - Taxonomy Catalog views.
    - APIs in PL/SQL and XQuery
    - No need to load XML Schema, linkbase in memory
  - Queryability
    - Queries over instance and taxonomy documents
    - As-Filed, Ad-hoc, Partial-document
  - Analytics
    - Live relational views over instance documents

• **XBRL Services**
  - Services to generate hierarchical relationships dynamically
  - Services to render XBRL reports, diff documents
  - All services dynamic and scalable.
XBRL Processing
Key Features of an XBRL Processor

• Reads and Writes Taxonomy and Instance Files
  – Discovers and Reads All References
  – Builds an internal memory model
  – Provides High level API to expose XBRL semantics

• Taxonomy and Instance Validation
  – Validation XBRL – Spec 2.1 plus Errata
  – Instance Document Calculations and Calculation Trace
  – Instance Document Dimension Validation
  – Instance Document Formula Execution

• XML Integration
  – Supports XML DOM Level II API(s)
  – Support XSLT and XQuery Interfaces
An XBRL Processor Must Support a Wide Range of Taxonomies
XBRL Processor Makes Development and Maintenance Easier

- XML Processor
- Hand Coded
- Syntax Processing
- Semantic Processing
- Complexity / Semantics

- US-GAAP
- FDIC Call Reports
- COREP / FINREP
- CRAS
- IFRS

2.1 Compliant XBRL Processor

Number of / Rate of Change
How Regulators Leverage XBRL

- Does the filing pass automated review?
- Is the filing consistent with domain best practices?
- Is the filing consistent with XBRL best practices?
- Does the filing conform to XBRL 2.1 specifications?
- Are the extensions valid?
- Is there a valid taxonomy?
- Is the grammar correct?
- Is there a valid schema?
- Is it well formed XML?

What is the quality of information provided?

Is the correct information provided?

Is the information formatted properly?

XML Processing

XBRL Validation & Calculation trace

XBRL Formulas
UBmatrix XBRL Processing Engine

- Designed for high volume, large scale reporting systems
- Extensible object oriented architecture
- Complete SDK (XPE API)
- XBRL and FRTA validation
  - XBRL 2.1 spec compliant
  - Calculation validation
  - Business rules validation
  - Fully supports XBRL Dimensions 1.0 and Aggregations working draft
- Transformation of taxonomies and instance documents using XSLT and XPath 2.0
- Supports “extended validation” via business rules
- Streaming document support
- Versions for both .Net and Java
Example XBRL Enabled Forms Collection Service Using Oracle DB – Currently in Implementation

- **Company A**
  - Downloads
  - E-Forms
  - XBRL Document Capture
  - Document Viewer
  - Validation Report

- **Company B**
  - XBRL Generator
  - XBRL Submitter

- **BI BACK OFFICE**
  - BI Tool
  - Application Database

- **Query & Viewing**
  - Document Viewer
  - XBRL Query Gateway

- **XBRL Document Database**
  - XBRL Document Archive
  - Taxonomies and Templates
  - XBRL-DB
  - XBRL-DB (BI) BACK OFFICE
  - BI Tool
  - Application Database

- **Mapping Tool**
  - XBRL Translator

- **Document Processing**
  - Reporting Manager
  - Analytics & Exception Reporting
  - Management Dashboard
UBmatrix

• Technology
  – First to support key XBRL technologies
  – Contributor to key taxonomies

• Products
  • Taxonomy Designer
  • Report Builder
  • XBRL Processing Engine
  • Enterprise Application Suite
  • Packaged solutions for COREP/FINREP & SEC VFP
  • Report Builder
  • XBRL Processing Engine
  • Enterprise Application Suite
  • Packaged solutions for COREP/FINREP & SEC VFP

• Customers
  – The largest XBRL systems use UBmatrix

“UBmatrix is at the heart of the XBRL market”
Cool Vendors in Finance and HCM, 2007
Putting it All Together-
The Demo
Questions?

For More Information

Sam – sam.idicula@oracle.com

Sriram – sriram.x.krishnamurthy@oracle.com

Steve – steve.levine@ubmatrix.com
ORACLE IS THE INFORMATION COMPANY