Inside Oracle Database 11g Release 2 XML DB

Nipun Agarwal  Vikas Arora  Mark Drake
Director       Senior Manager  Product Manager
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.
Presentation Outline

- XML DB Overview
- XML Query
- XML Index
- XML Storage
- Industry Schemas
- Demos
XML DB Overview
Why XML in the database?

• XML contains mission critical information
  – Interchange with external organizations
  – Web Services

• Need to manage XML effectively and efficiently
  – Reliability, Scalability, Availability
  – Accurate and fast information location and retrieval

• XML DB provides
  – Processing XML close to data for high scalability and performance
  – Unified management with other types of data
  – Relational interoperability
    • Evolve your applications to leverage XML
Advanced XML Capabilities

- Native storage for schema-based and schema-less XML
- Native XQuery Engine
- XML and Full-Text indexing
- Native storage for schema-based and schema-less XML
- XML views of relational Content
- Document ad data Centric Access
11gR2 Objectives

• Improve
  - Overall Performance and Scalability
  - Operational Completeness
  - Standard Conformance & Simplification
  - Semi-structured Data Management

• Focus on Key Industry Schemas
XML Query
XML Application Paradigms

- **Midtier XML Processing**: XDK
  - JDBC, OCI
  - XML STORAGE

- **XML Generation**: SQL/PLSQL, XQuery
  - RELATIONAL STORAGE
  - XML STORAGE

- **Database XML Processing**: SQL/PLSQL, XQuery
  - XML STORAGE

- **Relational Access over XML**: SQL/PLSQL, XQuery
  - XML STORAGE
XDK C and Java Libraries

- XML Parser
  - DOM & SAX Parser
- XML Schema Validator
  - DOM based validator
  - Stream based Validator
- XMLDiff and XMLPatch
- XPath, XSLT, XQuery processors
- Libraries Available in both C and Java version
XDK
11gr2 Enhancements

• XDK Java
  - End to End binary xml enhancements
  - Streaming execution over Scalable DOM
    • Improved performance by 8x for very large input.
    • Streaming execution of common XSLT elements
    • Constant memory usage with large input when streaming. Tested up to 2GB input with 10GB output

• XDK C
  - XML Virtual Machine
  - End to End binary xml enhancements
SQL/XML

• Defines an **XMLType** data type and operators
  - Generation functions
    • `XMLElement()`, `XMLAgg()`, `XMLAttributes()`, `XMLForest()`
  - XQuery functions
    • `XMLQuery()` : Fragment Extraction
    • `XMLTable()` : Projection
    • `XMLElements()` : Filtering
    • `XMLCast()` : Conversion to SQL type system
  - Ancillary functions
    • `XMLTransform()` : XSLT Transformations
    • `XMLNamespaces()` : Namespace management
XML Generation
11gr2 Enhancements

- **XQuery and SQL/XML operators**
  - Execution optimizations

- **Handling Large XQueries**
  - Re-write enabled for large size XQuery
  - 8x increase in size and complexity of supported XQuery operations
  - Upto 60x improvement for XML generation from relational data
XML Generation
Comparison of 11gr2 with 11gr1

Test case

- **Oracle 11gr2**
- **Oracle 11.1.0.6**
XMLType

• Makes database XML aware
• Abstraction for Storing XML in the database
• Application logic independent of physical storage
  – Flexible, native storage and indexing
  – Optimized for Schema-based and Schema-less XML
  – Object-relational, Binary and Text storage models

• Use as Type for
  – Table, Column, Variable, Argument or Return Value
• XMLType methods and SQL operators for
  – Query, DML, Transformations, Schema validation
• PL/SQL, C and Java APIs
XQuery

• W3C standard for generating, querying XML
  – Natural query language for XML content
  – Evolved from XPath and XSLT
• Basic construct is the FLWOR clause
  – FOR, LET, WHERE, ORDER, RETURN...
• Analogous to SQL in the relational world

• Use to
  – Query and update XML content in the database
  – Generate XML from relational data
  – Transform XML content
  – Query and update XML in a mid-tier environment
XQuery and SQL/XML
11gr2 Storage Independent Enhancements

- XML virtual machine (XVM)
  - 20x + improvement for functional evaluation of XQuery
  - Provides Native Optimization and Execution for procedural logic
    - Virtual Machine Based Architecture
    - XPath, XSLT, XQuery are compiled into byte code
    - Stack based engine for function executions, parameters & local & global variables
    - Pushes down the query part to the Query Processor
    - Based on Common DOM APIs
    - Leveraging Oracle Core for datatypes & functions
      - Also available as part of XDK
  - Enhanced XQuery Optimization Algebra Rules
  - Multi-phased XQuery, SQL/XML Transformation Driver
XMark Benchmark
Comparison of 11gr2 with 11gr1

→ 11gr2 is 4.51x faster
(based on gmean query times, 10X10m doc)

Oracle 11gr2 Structured Storage
Oracle 11.1.0.6 Structured Storage
XMark Benchmark
Comparison of Oracle XML DB with another DB

Oracle is 5x faster

Log Elapsed Time (ms)

Queries

(based on Gmean query time, 100M doc., Q8-12* timed out for another DB capped at 1800s)

- Red: Oracle 11gR2 OR Storage
- Light Gray: Another DB SB with XIDX
XML Standards
11gr2 Product Conformance and Participation

• SQL/XML 2006
  – Support for standard operators
• XQuery Language 1.0
  – Database conformance upto 91%
  – Active in XQuery 1.1 definition
• XQuery Java API (JSR 225)
  – Released as a standard
  – Standard API for Java access to XQuery
  – Oracle and Datadirect spec leads
• Simplification
  – Deprecate Oracle proprietary operators e.g. extract, extractValue, existsNode, ora:instanceof, ..
XMLIndex Structured Component
New in 11gr2

• Usecase
  – Typical XML Queries based on structured attributes within XML
  – Example 1:
    Relational Views over XML content
  – Example 2:
    Document centric content frequently queried on metadata attributes.
    Publications with Title, Author, Date, ..

• Example query:

```sql
SELECT * FROM DOCUMENT_TAB doc
WHERE XMLEXISTS(
  '$doc//document [
    title = "indexing XML Techniques" and
    pubdate > xs:date("2007-03-01") and
    pubdate < xs:date("2007-12-31") and
    affiliation = "Oracle"
  ]'
)
PASSING VALUE(doc) AS "doc")
```
XMLIndex Structured Component

Index Structured Metadata in XML Content

- Project out commonly searched structured attributes

- Physical rewrite using XQuery/XPath expression matching

- Pivot each leaf item as a column in the table
  - All xpath matching is avoided at run time
  - All joins to ensure the structured leaf data from the same parent node is avoided
  - All structured leaf data in the same group (having the same parent node) are stored in one row

- Secondary Indexes can be created on Structured Index
  - Relational indexes on projected scalar attributes
  - Text Index on projected text attributes
  - Domain specific Index on domain attributes, e.g. image
Structured XMLIndex Layout

XML data

```
<Document>
<title>Indexing XML Techniques</title>
<affiliation>Oracle</affiliation>
<pubdate>2007-04-10</pubdate>
...
</Document>

<Document>
<title>Object relational storage</title>
<affiliation>Oracle</affiliation>
<pubdate>2003-03-15</pubdate>
...
</Document>
```

Structured XMLIndex

<table>
<thead>
<tr>
<th>Row ID</th>
<th>Title</th>
<th>Affil</th>
<th>Pubdate</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Indexing XML Techniques</td>
<td>Oracle</td>
<td>2007-04-10</td>
</tr>
<tr>
<td>20</td>
<td>Object relational storage</td>
<td>Oracle</td>
<td>2003-03-15</td>
</tr>
</tbody>
</table>
## Structured XMLIndex

<table>
<thead>
<tr>
<th>Benchmark Query (Schema based Binary XML Storage)</th>
<th>Elapsed Time Improvement Over No XMLIndex</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1</td>
<td>5582x</td>
</tr>
<tr>
<td>Q2</td>
<td>2735x</td>
</tr>
<tr>
<td>Q3</td>
<td>111x</td>
</tr>
<tr>
<td>Q4</td>
<td>6.8x</td>
</tr>
<tr>
<td>Q5</td>
<td>175x</td>
</tr>
<tr>
<td>Q6</td>
<td>1130x</td>
</tr>
<tr>
<td>Q7</td>
<td>584x</td>
</tr>
</tbody>
</table>
XML Storage & Indexing
XML Use Cases

Structured

“Data-Centric”
Static Schema with Occasional Variability
No ANY or MIXED content

Semi Structured

“Variable Data”
Dynamic & Complex Schema
Islands of ANY or MIXED content

Unstructured

“Doc-centric”
No Schema Variable & Flexible Schema
Repeating ANY & MIXED content
XML DB Customers

Structured

Relational to XML

XML Data Management

Bayer,
Arch

Insurance Group®

BMW

GM

ThyssenKrupp

Medtronic

NYFINX

8i/9i

Semi-Structured

XML Data Management

bp

Lamborghini

Waters

Moody’s

SAFRAN

Temenos

ENERGYSYS

UBMatrix

Intuit

Genentech

9i/10g

11g

Document

Oracle

12g
Object-Relational Storage

- XML stored as objects in relational tables
- Suitable for highly structured XML use-cases
- DOM fidelity
- Predicated on an XML Schema
  - Storage model automatically generated from the XML Schema
- Fragment and Leaf level updates
- Supports database features like partitioning
Object Relational Storage
11gr2 Enhancements

- Physical Rewrite enhancements
  - Inheritance
  - DML Operations
- Fast Path Insert
- Manageability enhancements
  - Choose intelligent defaults
  - Make repetitive tasks easier
Object Relational Improvements
Comparison of 11gr2 with 11gr1

![Bar chart comparing Log Elapsed Time (ms) for different test cases between Oracle 11gr2 and Oracle 11.1.0.6. The chart shows that Customer1 (inheritance) has the highest Log Elapsed Time for Oracle 11gr2, while Customer4* (Fastpath insert) has the highest for Oracle 11.1.0.6.]
Binary XML Storage

- XML stored in a post-parse representation in a LOB
- Schema-less and XML Schema aware versions
- Format is optimized for indexing and fragment extraction
- Single representation used on disc, in-memory and on-wire
- Reduced storage requirements
  - Tags are tokenized
  - Content stored in native representation
- Lower CPU and memory usage
  - Support streaming evaluation for queries
Binary XML Storage
Enhancements

- New search-based decoder
  - Very efficient for XPath Evaluation
- Schema-aware NFA
  - Use XML Schema to pre-calculate XPath and push transitions down to search-based decoder
- Document-level Summary
  - Allows fast streaming for forward axes over large documents
- XPath Evaluation Cache
  - Identify patterns of repeated XPath Evaluation
  - Cache results and build in-memory index for filters
Binary XML - Comparison with another DB

1/6th the size 3x faster

<table>
<thead>
<tr>
<th>Storage needed for XMark data</th>
<th>Mean XMark Query Response functional eval</th>
</tr>
</thead>
<tbody>
<tr>
<td>67 MB</td>
<td>451 msec</td>
</tr>
<tr>
<td>10 MB</td>
<td>161 msec</td>
</tr>
</tbody>
</table>

Another DB
Oracle 11.2
XMLIndex Path-based

- Primary use case in conjunction in Binary XML
- Available since 11gR1
- Organizes paths and values in single path table
- Allows easy indexing of interesting sub-trees
- Allows asynchronous maintenance
- Updates to document result in piece-wise index updates
- Ideal when xpath to be queried not known apriori
## XMLIndex Path-based Layout

<table>
<thead>
<tr>
<th>RID</th>
<th>path</th>
<th>Order key</th>
<th>locator</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>/Document</td>
<td>1</td>
<td>Locator to get binary content</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>/Document/Title</td>
<td>1.1</td>
<td>Locator to get binary content</td>
<td>Indexing XML Techniques</td>
</tr>
<tr>
<td>10</td>
<td>/Document/Affiliation</td>
<td>1.2</td>
<td>Locator to get binary content</td>
<td>Oracle</td>
</tr>
<tr>
<td>10</td>
<td>/Document/pubDate</td>
<td>1.3</td>
<td>Locator to get binary content</td>
<td>2007-04-10</td>
</tr>
<tr>
<td>20</td>
<td>/Document</td>
<td>1</td>
<td>Locator to get binary content</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>/Document/Title</td>
<td>1.1</td>
<td>Locator to get binary content</td>
<td>Object relational storage</td>
</tr>
</tbody>
</table>
XMLIndex Path-based
11gR2 Enhancements

- Partitioning, parallel index creation and parallel query supported
- Physical rewrite for path subsets
- Queries improve by 5x on average for XMark (10M)
- 5 XMark queries improve by 20x
- Asynchronous DML performance improves 2.5x
Oracle Binary XML
Client Access
End to End Binary XML

- Graph based on retrieval of a customer dataset
- Token caching on client-side for Binary XML support
- Improve performance of application using XDK/JAVA and OCI
XML Schema
XML Schema in Oracle XML DB

- Validation of instance documents
- Object Relational storage model derived from XML Schema
  - SQLTypes automatically generated from type model defined by the XML Schema.
  - Content persisted as SQLTypes in relational tables.
- Binary Storage uses XML Schema to improve storage efficiencies
  - Simple types mapped to native formats
  - Improved tokenization algorithms for elements and attributes
- XML Indexing uses XML Schema to improve query optimization
XML Schema Enhancements

• Schema registration performance
  - Eliminate internal and external memory fragmentation
  - Optimized schema loading
  - Time and memory improves by 50x for US-GAAP, HL7, NIEM

• Schema Validator Cache
  - Improves XML schema validation by around 5x, more for small docs

• Can handle complex industry schemas
  - GJ-XML, GML, US GAAP, NIEM, HL7, FixML, MPEG-7, KML
  - ACORD, SDMX, FPML, Reed, OAGIS, MPEG7: Binary & O-R
XML Schema
Performance Improvements over 11gR1

Avg. improvement for
ACL, SecurityClass, PO, NIEM, HL7, US GAAP, FPML
XML Database Market Trends

• Vertical Industry standards driving XML adoption
  – Financial: XBRL, FpML
  – Healthcare: HL7 CDA
  – Government: NIEM

• XML as the persistent standard for documents
  – MS Office: Open XML
  – OpenOffice: Open Document Format
  – Adobe Framemaker: DITA

• XML as the standard for extensible data and metadata
  – Oracle Apps (Clinical Healthcare)
  – Data Pump, Spatial, Multimedia (DICOM)
  – EM Repository
# Cross Functional Completeness

Verified in 11gR2

<table>
<thead>
<tr>
<th>Functional Area</th>
<th>Binary</th>
<th>OR</th>
</tr>
</thead>
<tbody>
<tr>
<td>RMAN</td>
<td></td>
<td>●</td>
</tr>
<tr>
<td>Physical Standby</td>
<td></td>
<td>●</td>
</tr>
<tr>
<td>RAC</td>
<td></td>
<td>●</td>
</tr>
<tr>
<td>PQ</td>
<td></td>
<td>●</td>
</tr>
<tr>
<td>JDBC</td>
<td></td>
<td>●</td>
</tr>
<tr>
<td>SQLLoader</td>
<td></td>
<td>●</td>
</tr>
<tr>
<td>DataPump</td>
<td></td>
<td>●</td>
</tr>
<tr>
<td>TTS</td>
<td></td>
<td>●</td>
</tr>
<tr>
<td>Partitioning (Range, Hash, List)</td>
<td></td>
<td>●</td>
</tr>
<tr>
<td>Text Index</td>
<td></td>
<td>●</td>
</tr>
</tbody>
</table>
XML DB Repository

• Organize content as Files in Folders rather than rows in tables
• Accessible using standard desktop Tools
  – HTTP, FTP and WebDAV protocols
• Enables document centric development paradigm
  – Path based access to content
  – Queries based on location
  – Supports URL centric standards like XLink and Xinclude
XML DB Repository

- **Access control**
  - Grant / Revoke permissions on a document by document basis
- **Versioning**
  - Simple linear versioning model with Check-In and Check Out
- **Event model**
  - Associate code with operations on files and folders
- **Standard and user defined Metadata**
  - Manage metadata independently from content
- **JCR Connector (JSR-170)**
  - Java content management system API
XML DB Repository
Enhancements

• Secure files for all repository content
• Create Operations
  • Improved caching of XDB system schemas in SGA
  • Optimized in-memory structures
  • Enabled “direct insert” for resources
• Retrieval
  • Direct path gets for resources
• Queries (equals_path)
  • Direct invocation of function
• Update
  • Direct Update of Contents
XML DB Repository
Performance Improvements over 11gR1
Conclusion – 11gR2

• Improves
  – Overall Performance and Scalability
  – Operational Completeness
  – Standard Conformance & Simplification
  – Semi-structured Data Management

• Focus on Key Industry Schemas
Future…

- **Document Centric XML**
  - XQuery Full-text support
  - XQuery update and modules
  - Fast & Scalable DOM Tree Traversal support
- **Provide complete solutions for Industry Standard XML**
  - XBRL
  - OpenXML (MS Office)
- **More Operational completeness for XML and Repository**
The preceding 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.
ORACLE IS THE INFORMATION COMPANY