ORACLE

An Introduction to Oracle XML DB in Oracle Database 19c and 21c

Technical Overview

Agenda

- Introduction to XML DB
- XMLType
- SQL/XML
- XML Indexing
- XML Schema
- XML Generation
- XML Database Native Web Services
- XDK C and Java
- XBRL Extension to Oracle XML DB
- References

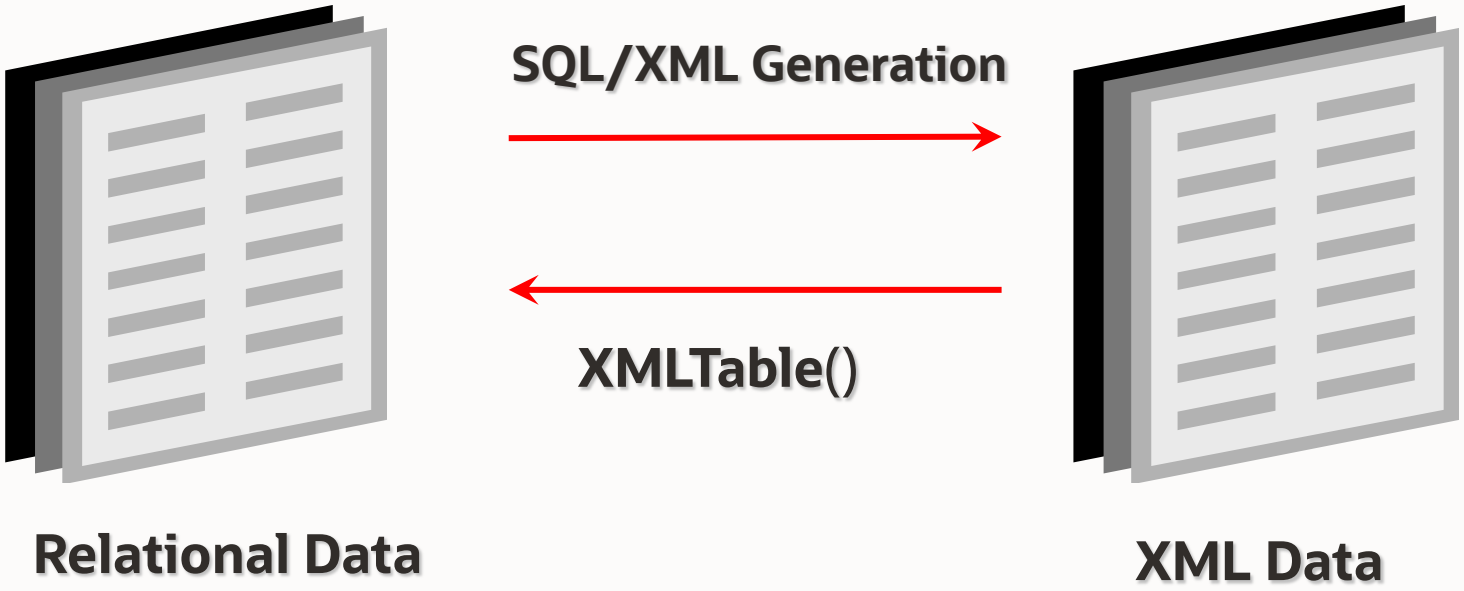
Oracle's XML Vision

- XML data model for OLTP
  - Fast document-centric CRUD operations
- XML data model for OLAP
  - Document-centric query, search, analytics and data integration
- Multi-Model interoperability
  - Support declarative multi-model transformation via SQL
  - Support bi-directional transformations between hierarchical data and relational data
- Enable a single source of truth for all your data: XML, relational, JSON, Text, Spatial and more.
- Deliver Oracle's commitment to Reliability, Security, Availability and Scalability

XML Data Model for OLAP and OLTP

	XML OLAP	XML OLTP
XML Storage	Binary	Binary
Query	SQL/XML with XQuery, XQFT, XSLT	SQL/XML with XQuery
XML Update	SQL/XML with XQuery Update (XQUF)	SQL/XML with XQuery Update (XQUF)
XML Indexing	XML Search (full-text) Index	Structured XML Index (SXI)
XML Schema	XML Schema Validation	XML Schema Validation

Multi-Model Interoperability



Agenda

- Introduction to XML DB
- XMLType
- SQL/XML
- XML Indexing
- XML Schema
- XML Generation
- XML Database Native Web Services
- XDK C and Java
- XBRL Extension to Oracle XML DB
- References

XMLType – Binary XML

```
CREATE TABLE purchaseorder (
  po_number   NUMBER,
  po_details.   XMLTYPE
)
XMLTYPE COLUMN po_details  STORE AS BINARY XML ;
```

- Standard data type, makes database XML aware
  - Use as Column, Variable, Argument or Return Value
- Store post-parse binary representation of XML
- Optimized for streaming, indexing and fragment extraction
- Load table using SQL, JDBC, OCI, PLSQL and SQL Loader

Agenda

- Introduction to XML DB
- XMLType
- SQL/XML
- XML Indexing
- XML Schema
- XML Generation
- XML Database Native Web Services
- XDK C and Java
- XBRL Extension to Oracle XML DB
- References

SQL/XML

- Standards compliant
  - Strict adherence and conformance
- XQuery operators
  - XMLQuery() : Fragment Extraction
  - XMLTable() : Projection
  - XMLExists() : Filtering
  - XMLCast() : Conversion to SQL type system
- Other operators
  - XMLTransfom(): XSL based transformation
  - XMLNamespaces(): Namespace management
  - XMLSerialize(): Serializing XML data into a string or LOB
  - XMLParse(): parsing XML Data into XMLType instance

XMLExists() XQuery Predicates

- Use in SQL where clause to filter rows based on an XQuery expression
- Bind variables are supplied via the “Passing” clause

```
SELECT p.po_details “XML”
    FROM purchaseorder p
     WHERE XMLExists (
                '$PO/PurchaseOrder[Reference=$REF]’
                PASSING p.po_details as "PO",
                'SKING-20021009123336131PDT' as "REF“
           );
```

XML

```
<PurchaseOrder >
  <Reference>SKING-20021009123336131PDT</Reference>
  …
</PurchaseOrder >
```

XMLQuery() Fragment access

- Use in SQL to extract a fragment from each document in a result set
- Bind variables are supplied via the “Passing” clause

```
SELECT XMLQuery (
                '$PO/PurchaseOrder/ShippingInstructions’
                PASSING p.po_details as "PO“
                returning content) XML
  FROM purchaserorder p
  WHERE XMLExists (
            '$PO/PurchaseOrder[Reference=$REF]‘
             PASSING p.po_details as "PO",  'SKING-20021009123336131PDT' as "REF");
```

XML

```
<ShippingInstructions>
  <name>Steven A. King</name>
...
</ShippingInstructions>
```

XMLTable() Relational Views of XML

- The “COLUMNS” clause extends XMLTable, allowing the creation of in-line relational views of XML content
- Enables SQL operations on XML content
  - Views allow Non-XML aware tools access to XML content
- Collection hierarchy managed using chained XMLTable operations
  - Repeating elements passed down the chain as XMLType fragments

XMLTable() Columns Clause

```
SELECT m.REFERENCE, i.LINENO, i.QUANTITY
    FROM purchaseorder p,
         XMLTable(
           '$PO/PurchaseOrder'     passing p.po_details as "PO"
           COLUMNS
             REFERENCE                  VARCHAR2(32) PATH 'Reference',‘,
             LINEITEM_FRAGMENT          XMLTYPE PATH 'LineItems/LineItem’
         ) m,
         XMLTable(
           '$LI/LineItem'              passing m.LINEITEM_FRAGMENT as "LI"
            COLUMNS
              LINENO                        NUMBER(4)   PATH '@ItemNumber‘,
              UPC                           NUMBER(14) PATH 'Part/text()‘,
              QUANTITY                      NUMBER(5)   PATH 'Quantity‘
         ) i
   WHERE i.UPC =  '24543000457’;
```

RERERENCE	LINENO	QUANTITY
AKHOO-20100418162507692PDT	2	2
PVARGAS-20101114171322653PST	1	7
JTAYLOR-20100518182653281PDT	5	4

XMLQuery() XQuery-Update support

- Standards-compliant update of XML content
  - http://www.w3.org/TR/xquery-update-10/
- Combine an XMLQuery operator containing an XQuery-Update expression with a SQL Update statement
  - The XQuery-Update supplies the new value for the XMLType

```
UPDATE table_name
     SET xml_column =  XMLQUERY(
                              'copy $NEWXML := $XML modify (
                                let $TARGET := $NEWXML/rootElement/targetElement
                                 return replace node $TARGET with $NEWCONTENT
                              )
                              return $NEWXML'
                              passing XML_COLUMN as "XML",  V_NEW_CONTENT as " NEWCONTENT "
                              returning content
                           )
     WHERE ...
```

Agenda

- Introduction to XML DB
- XMLType
- SQL/XML
- XML Indexing
- XML Schema
- XML Generation
- XML Database Native Web Services
- XDK C and Java
- XBRL Extension to Oracle XML DB
- References

XQuery Full Text Index

- Index everything for ad-hoc queries
  - Requires no knowledge of the XML being indexed
- Supports Full text and XML Queries
  - Queries use the index to optimize equality, range, text searches.

```
CREATE INDEX purchaseorder_xqft_idx
   ON purchaseorder (po_details)
      INDEXTYPE IS CTXSYS.CONTEXT
       PARAMETERS (
           'storage STORAGE_PREFS section group XQFT'
       );
```

XQuery Full Text Support

- Use with XMLExists() operator to perform Full-Text filtering
- XMLExists() is evaluated through XML search index

```
SELECT po_details
   FROM purchaseorder p
   WHERE XMLExists(
            '$P/PurchaseOrder/ShippingInstructions/Address/street[. contains text
            "Big" ftand "Street" ]'
            PASSING p.po_details as “P”
   );
```

Structured XML Index

```
CREATE INDEX purchaserorder_xml_idx
        ON purchaseorder (po_details)  INDEXTYPE IS XDB.XMLINDEX
            PARAMETERS ('PARAM PO_SXI_PARAMETERS');
```

- Indexes “Islands of Structure”
  - Requires some knowledge of the XML being indexed and the queries that will be performed
- Specific leaf-level nodes projected into relational tables
  - Table for each island, leaf node values stored as columns
  - Very fast extraction, aggregations over leaf nodes
- Data type aware
- Based on XMLTable syntax()
- Optimizes all SQL/XML operators
  - XMLQuery(), XMLTable() and XMLExists()

XML Table Index DDL

```
CALL DBMS_XMLINDEX.registerParameter(
       'PO_SXI_PARAMETERS',
       'GROUP   po_lineitem
            XMLTable po_index_master
              "/PurchaseOrder"
              COLUMNS
                 reference              VARCHAR2(30)          PATH "Reference",
                 lineitem               XMLType               PATH "LineItems/LineItem"
             VIRTUAL XMLTable           po_index_lineitem
              "/LineItem"  PASSING lineitem
              COLUMNS
                 itemno                 NUMBER(38)            PATH "@ItemNumber",
                 upc                    NUMBER(14)            PATH “Part/text()",
                 description            VARCHAR2(256)          PATH "Part/@Description"
      ');
```

Agenda

- Introduction to XML DB
- XMLType
- SQL/XML
- XML Indexing
- XML Schema
- XML Generation
- XML Database Native Web Services
- XDK C and Java
- XBRL Extension to Oracle XML DB
- References

XML Schema

- Use XML schema to validate an XML instance

```
DBMS_XMLSCHEMA.registerSchema  (
   SCHEMAURL       => 'http://www.example.com/xsd/purchaseOrder.xsd',
   SCHEMADOC       => xmlType(bfilename('XMLDIR','po.xsd'), nls_charset_id('AL32UTF8')),
   GENTYPES        => FALSE,
   GENTABLES       => FALSE,
   OPTIONS         => DBMS_XMLSCHEMA.REGISTER_BINARYXML )
```

```
SELECT p.po_details.isSchemaValid('http://www.example.com/xsd/purchaeOrder.xsd',
          'purchaseOrder')
FROM purchaseorder p;
```

Agenda

- Introduction to XML DB
- XMLType
- SQL/XML
- XML Indexing
- XML Schema
- XML Generation
- XML Database Native Web Services
- XDK C and Java
- XBRL Extension to Oracle XML DB
- References

Generating XML using SQL/XML

- SQL/XML makes it easy to generate XML from relational data
- XMLElement()
  - Generates an Element with simple or complex content
- XMLAttributes()
  - Adds attributes to an element
- XMLAgg()
  - Generates an XML Fragment
  - Aggregation operator used to process the results of a nested sub-query
- XMLForest, XMLConcat, XMLComment and etc

Example : XML Generation using SQL/XML

```
SELECT  xmlElement ( "Department",
              xmlAttributes ( d.DEPTNO as “Id"),
              xmlElement ("Name", d.DNAME),
              xmlElement ("Employees”, (
                   SELECT xmlAgg (
                        xmlElement ("Employee",
                             xmlForest (
                                  e.ENAME as "Name",
                                  e.HIREDATE as"StartDate”
                                            )
                                       )
                                  )
                    FROM emp e
                    WHERE e.deptno = d.deptno
                  ) )
            ) AS xml
FROM dept d;
```

XML

```
<Department Id="10">
 <Name>ACCOUNTING</Name>
 <Employees>
  <Employee employeeId="7782">
   <Name>CLARK</Name>
   <StartDate>1981-06-09</StartDate>
  </Employee>
  <Employee”>
   <Name>KING</Name>
   <StartDate>1981-11-17</StartDate>
  </Employee>
  <Employee>
   <Name>MILLER</Name>
   <StartDate>1982-01-23</StartDate>
  </Employee>
 </Employees>
</Department>
```

Agenda

- Introduction to XML DB
- XMLType
- SQL/XML
- XML Indexing
- XML Schema
- XML Generation
- XML Database Native Web Services
- XDK C and Java
- XBRL Extension to Oracle XML DB
- References

Native Oracle XML DB Web Services

- 'Zero-Development', 'Zero-Deployment' solution for publishing PL/SQL packages.
  - Any package method, function or procedure can be accessed as a SOAP end-point
- Leverages the Oracle XML DB HTTP Server
  - No additional infrastructure required
- Automatic generation of WSDL
  - URL to Package, Function or Procedure mapping scheme
- Uses XML DB infrastructure for processing request and generating response
- Includes 'SQL Query' and 'XQuery' Services

Agenda

- Introduction to XML DB
- XMLType
- SQL/XML
- XML Indexing
- XML Schema
- XML Generation
- XML Database Native Web Services
- XDK C and Java
- XBRL Extension to Oracle XML DB
- References

XDK C and Java

- XML Developer's Kit
- Standalone components in C and Java for XML applications
- XML Parser
  - SAX APIs
  - DOM APIs
- XSLT Processors
  - XML document transformation by applying an XSLT stylesheet
- XML Schema Processors
  - Validate XML against a DTD or XML schema

Agenda

- Introduction to XML DB
- XMLType
- SQL/XML
- XML Indexing
- XML Schema
- XML Generation
- XML Database Native Web Services
- XDK C and Java
- XBRL Extension to Oracle XML DB
- References

XBRL Extension to Oracle XML DB

- Native database storage of XBRL data.
- Database enforcement of integrity, based on XBRL rules.
- Ability to query XML data using XBRL semantics.
- Relational representation of XBRL content. Ability to expose XBRL content to relational applications and SQL queries.
- PL/SQL transforming procedures that generate derived XBRL views based on XBRL relational representations, network generation APIs, or dimensional information.
- Scalable XBRL services: reports, network generation, transformations.
- Online analysis based on XBRL dimensions, both explicit and typed.

Agenda

- Introduction to XML DB
- XMLType
- SQL/XML
- XML Indexing
- XML Schema
- XML Generation
- XML Database Native Web Services
- XDK C and Java
- XBRL Extension to Oracle XML DB
- References

Learn more..

Oracle.com

- Oracle XML DB - https://www.oracle.com/database/technologies/appdev/xmldb.html
- Oracle Autonomous Database - https://www.oracle.com/database/autonomous-database.html

Documentation

- XML DB Developer Guide - https://docs.oracle.com/en/database/oracle/oracle/database/21/adxdb/

Livesql

- XML DB Tutorial - https://livesql.oracle.com/apex/livesql/file/tutorial_HE5NRRMNBOHLLKRLZJU0VNRCB.html

Full suite of XML Features

Functionality	XQuery	1.0+
	XQuery update	1.0
	XQFT	1.0
	XSLT	1.0
	APIs	JDBC, PLSQL, SQL, OCI, ODP.net
	Stored Procedures, Triggers	■
	Schema Flexibility, Validation	■
	Multi-model	■
	Indexing	Search Index and Structured Index
	Enterprise Security	■
Infrastructure	Replication	■
	Scale-out (Sharding, Partitioning)	■
	ACID transactions	■
	Comprehensive end-to-end application life cycle support	■

■ Full support, □ No support, ◘ Partial support

ORACLE