Oracle XML DB

Types of XML Documents (in general)

There are two types of XML documents – the data- and document-centric XML documents.

Data-centric XML Documents

- for data collections with complex structures of usually numerical and non-text attribute-value data
- utilized usually for export/import of data from/to non-XML databases or applications
- the XML documents have good and regular struture
- the ordering of attribute-value data in the XML documents is not important

For example, the following breakfast menu was adopted from W3Schools [https://www.w3schools.com/xml/xml_xslt.asp]:

```
<?xml version="1.0" encoding="UTF-8"?>
<breakfast_menu>
  <food>
    <name>Belgian Waffles</name>
    <price>$5.95</price>
    <description>Two of our famous Belgian Waffles with plenty of
real maple syrup</description>
    <calories>650</calories>
  </food>
  <food>
    <name>Strawberry Belgian Waffles</name>
    <price>$7.95</price>
    <description>Light Belgian waffles covered with strawberries
and whipped cream</description>
    <calories>900</calories>
  </food>
  <food>
    <name>Berry-Berry Belgian Waffles</name>
    <price>$8.95</price>
    <description>Light Belgian waffles covered with an assortment of fresh berries and whipped cream</description>
    <calories>900</calories>
  </food>
  <food>
    <name>French Toast</name>
    <price>$4.50</price>
    <description>Thick slices made from our homemade sourdough bread</description>
    <calories>600</calories>
  </food>
  <food>
    <name>Homestyle Breakfast</name>
    <price>$6.95</price>
    <description>Two eggs, bacon or sausage, toast, and our ever-popular hash browns</description>
    <calories>950</calories>
  </food>
</breakfast_menu>
```

Document-centric XML Documents

- for text documents that are marked up as XML to capture document structure (paragraphs, sections, footnotes etc.).
- utilized usually does not make sense to store such XML documents into databases as structured data
- the XML documents do not have regular struture, a same element can be used in different context
- the ordering of text data in the XML documents is important

For example, the following XHTML web-page source code was adopted from W3Schools [https://www.w3schools.com/html/html_xhtml.asp]:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <title>Title of document</title>
  </head>
  <body>
    A break: <br />
    A horizontal rule: <hr />
    An image: <img src="happy.gif" alt="Happy face" />
  </body>
</html>
```

Types of XML Data Storage, Transfer, and Processing Software (in general)

XML can be utilized in various ways.

Databases with Support of XML Data

- the most of current relational database products support XML data
- XML data are stored as structured/object data and may be queried
- e.g., Oracle XML DB, PostregSQL XML type, MySQL XML functions, etc.

Native XML Databases

- data model is based on the Document Object Model (DOM) [https://en.wikipedia.org/wiki/Document_Object_Model]
- queries and operations on the XML data are fast
- e.g., BaseX [http://basex.org/], eXist-db [http://www.exist-db.org/], and Sedna [https://www.sedna.org/]

XML Wrappers/Mappers

- to access XML data in relational way, e.g., to query such data in SQL
- e.g., an XML wrapper as a component of the DB2 federated architecture [https://www.redbooks.ibm.com/abstracts/sg246994.html]

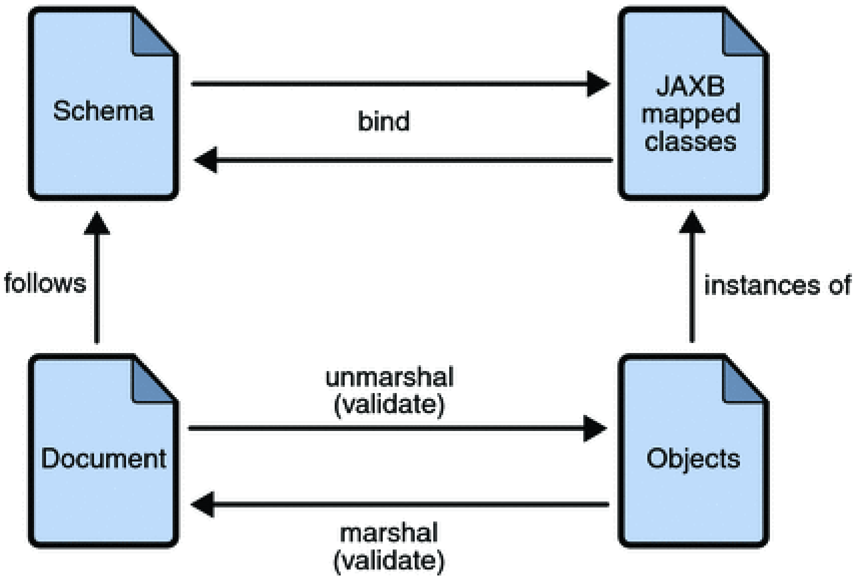
Middleware Utilizing XML Technolgies

- to communicate between application components in XML format

- the XML can be utilized as platform-independent data format with the ability to check validity, to do enrichment, transformations, and content-based routing of messages in XML
- e.g., SOAP/RDF/RSS-based Web-services [https://www.w3schools.com/xml/xml_services.asp], ESB [https://en.wikipedia.org/wiki/Enterprise_service_bus]

XML Data Binding

- the mapping of XML to object data and vice versa
- to provide a persistence of the objects (i.e., serialization and deserialization to/from XML data)
- e.g., Java Architecture for XML Binding [https://en.wikipedia.org/wiki/Java_Architecture_for_XML_Binding]



XML in Relational Databases (in general)

There are three approaches to storing and querying XML data in relational database:

1. the XML data are stored in binary or text data types (`CLOB` or `BLOB` SQL data types)
   - fast and easy to implement

  - strictly respects/keeps the content of the original XML document
  - difficult to query or update as a relational databse engine cannot access the XML elements inside the stored text/binary values

2. the XML data are parsed and individual elements-content and attribute-values are stored in a relation table as column-values
   - the extracted data are relational and can be queried/modified by common means
   - the original XML document is lost and it may be difficult to reconstruct from the relational data
3. the XML data are stored in SQL `XML` data type as object-relational data
   - XML objects have attributes read/write and methods to update/query the XML data
   - the content of the XML document can be integrated with relational data into SQL statements and constraints

XML and SQL

SQL:2003-14 [https://en.wikipedia.org/wiki/SQL:2003] defines

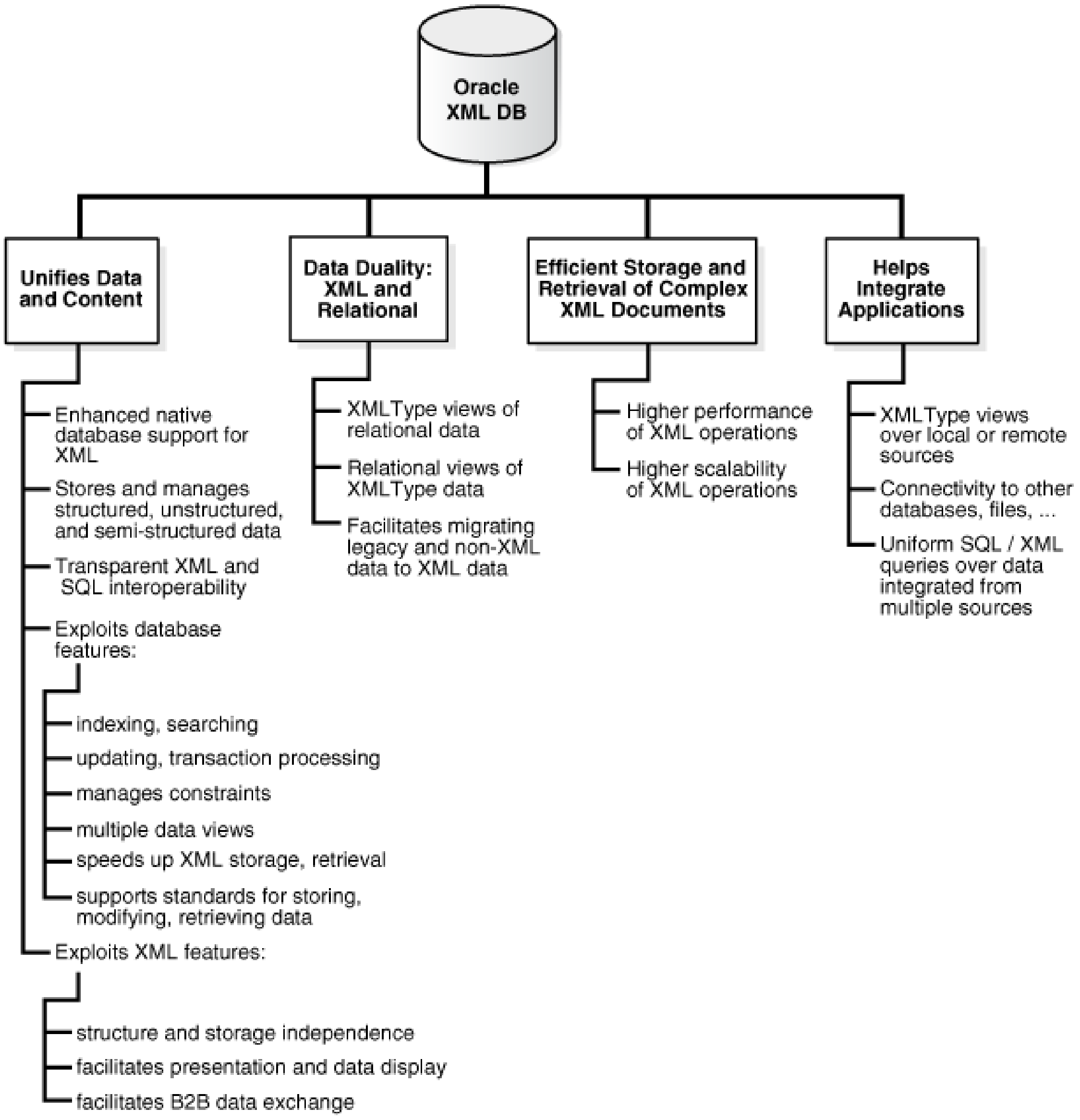
- the basic `XML` datatype,
- mappings of XML and SQL data types and meta-data,
- predicates to check XML content (`CONTENT`), match to a XQuery expression (`XMLEXISTS`), and validity (`VALID`),
- and functions such as `XMLQUERY` to extract values from XML fields.

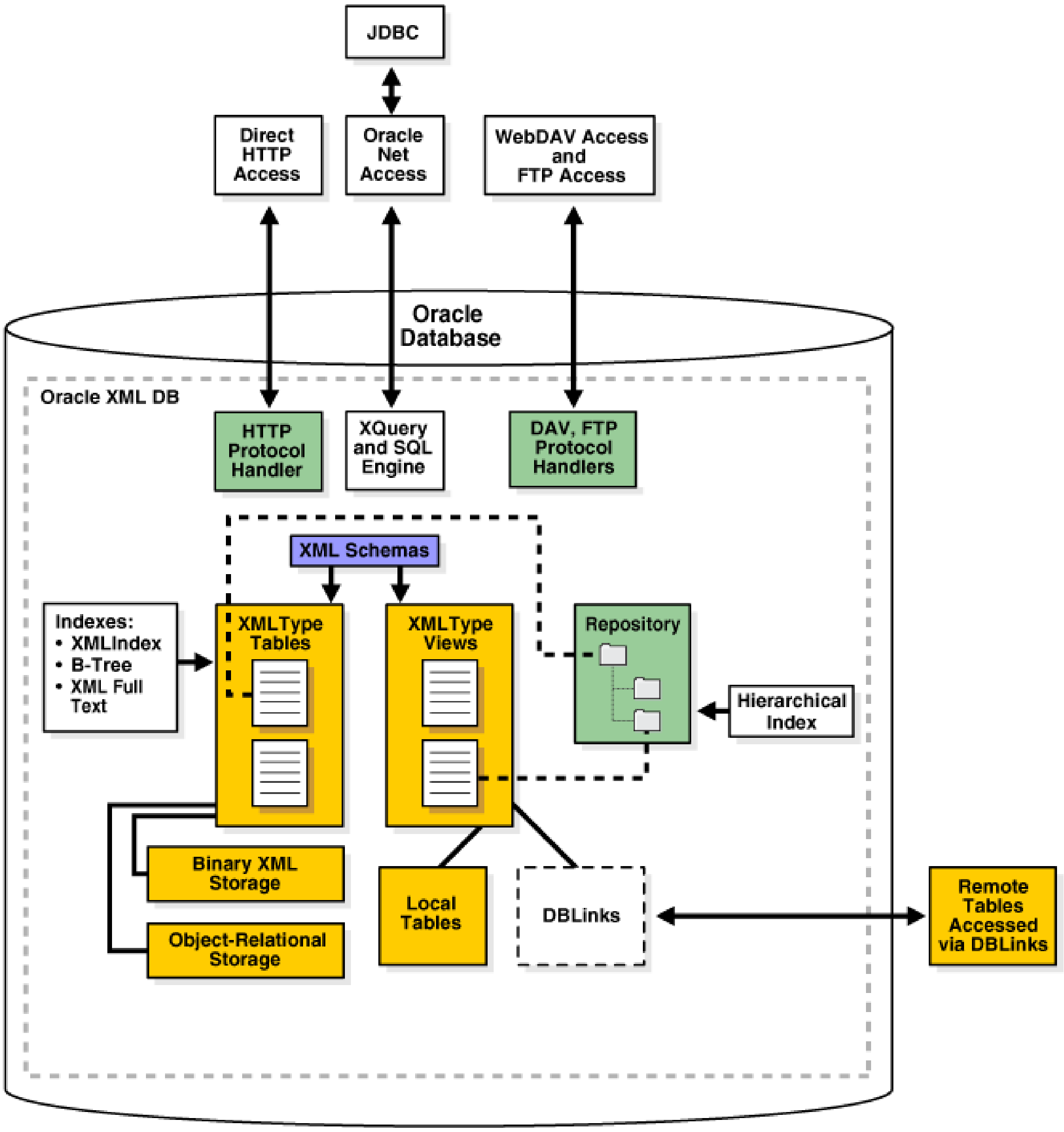
However, many database vendors do not support this standard (e.g., MySQL) or provide more-or-less different solution (e.g., Oracle, IBM DB2, or MS SQL Server). For Oracle, see Oracle Compliance with SQL/XML [https://docs.oracle.com/en/database/oracle/oracle-database/18/sqlrf/Oracle-Compliance-with-SQLXML2011.html#GUID-0D0F19C8-0FB7-4FDD-A55B-18839F340E17].

The XML in SQL were later updated mostly in SQL:2006-14 [https://en.wikipedia.org/wiki/SQL:2006], and partially also in SQL:2008-14 [https://en.wikipedia.org/wiki/SQL:2008] and SQL:2011-14 [https://en.wikipedia.org/wiki/SQL:2011].

XML in Oracle

- XML data supported in Oracle 9.2 and later (since 2002)
- there is XMLType [https://docs.oracle.com/en/database/oracle/oracle-database/18/adxdb/intro-to-XML-DB.html#GUID-02592188-AC38-4D00-A2FD-9E53604065C8] which partially corresponds to SQL:2003 XML data type
- XML data are stored and queried by means of Oracle XML DB component


Oracle XML DB
Unifies Data and Content
Enhanced native database support for XML
Stores and manages structured, unstructured, and semi-structured data
Transparent XML and SQL interoperability
Exploits database features:
indexing, searching
updating, transaction processing
manages constraints
multiple data views
speeds up XML storage, retrieval
supports standards for storing, modifying, retrieving data
Exploits XML features:
structure and storage independence
facilitates presentation and data display
facilitates B2B data exchange
Data Duality: XML and Relational
XMLType views of relational data
Relational views of XMLType data
Facilitates migrating legacy and non-XML data to XML data
Efficient Storage and Retrieval of Complex XML Documents
Higher performance of XML operations
Higher scalability of XML operations
Helps Integrate Applications
XMLType views over local or remote sources
Connectivity to other databases, files, ...
Uniform SQL / XML queries over data integrated from multiple sources



Querying XML Data in SQL

XPath and XSLT

- XPath [https://en.wikipedia.org/wiki/XPath] is a language for querying XML data/documents by a path to matching elements
- XSLT [https://en.wikipedia.org/wiki/XSLT] is a language for transforming XML documents into other XML documents

For example, the following SQL statement with an Xpath expression was adopted from Querying XML [https://www.oracle.com/technetwork/database/database-technologies/xmldb/oow-2004-querying-xml-132839.pdf]:

```
SELECT id AS id,
  EXTRACT(resolution, '/resolution[@public-
private="public"]/action') AS action,
  EXTRACTVALUE(resolution, '/resolution[congress="108"]/official-
title') AS title
FROM resolutions_xml
WHERE EXISRSNODE(resolution, '/resolution[legis-num = 558]') = 1;
```

XQuery

For example, the following XQuery expression was adopted from Querying XML [https://www.oracle.com/technetwork/database/database-technologies/xmldb/oow-2004-querying-xml-132839.pdf]:

```
FOR $r IN DOC("/public/oow04/resolution.xml")/resolution
  LET $a := $r/action
  WHERE $a/action-date="20040311"
  ORDER BY $r/legis-num ASCENDING
  RETURN
<all-sponsors>
  {$a/action-desc/sponsor}
  {$a/action-desc/cosponsor}
</all-sponsors>
```

An XQuery can be integrated into an SQL query in Oracle as:

```
SELECT XMLQUERY('
  FOR $v IN DOC("/public/xml2004/vote.xml")/rollcall-vote
  LET $d := $v/vote-data
  RETURN
<vote-count>
  <total>
    {count($d/recorded-vote[vote="Yea"])}
  </total>
  <republican>
    {count($d/recorded-vote[legislator/@party="R" and vote="Yea"])}
  </republican>
  <democrat>
    {count($d/recorded-vote[legislator/@party="D" and vote="Yea"])}
  </democrat>
</vote-count>' RETURNING CONTENT) AS result FROM DUAL;
```