
ORACLE®
SOFTWARE POWERS THE INTERNET™

Geoff Lee
Senior Product Manager
Oracle Corporation

ORACLE

XML in Oracle9*i*

A Technical Overview

ORACLE

Agenda

- Survey
- Technical Overview
- Summary
- Q & A

Agenda

- Survey
- Technical Overview
- Summary
- Q & A

XML in Oracle9i - Overview

- XML and its family of standards are vital to the future of e-Business
- Oracle9*i* XML Developer's Kits support the family of XML standards to provide a complete XML-enabled Internet application platform
- Oracle9*i* Database Native XML support enables fast, flexible, and scalable storage and retrieval of XML data and documents
- XML Messaging and Transformation support in AQ provide a centralized, easy to manage, secure infrastructure for global messaging

Agenda

- Survey
- Technical Overview
  - XML
  - XDK
  - Database Native XML
  - XML Messaging
- Summary
- Q & A

Application Requirements

1 **Internet Content Management**

Consolidate Internet content
Build dynamic web sites/portals

2 **Internet Application Development**

Make web site transactional, secure, scalable and available

5 **Business Intelligence**

Capture, analyze, and share business intelligence

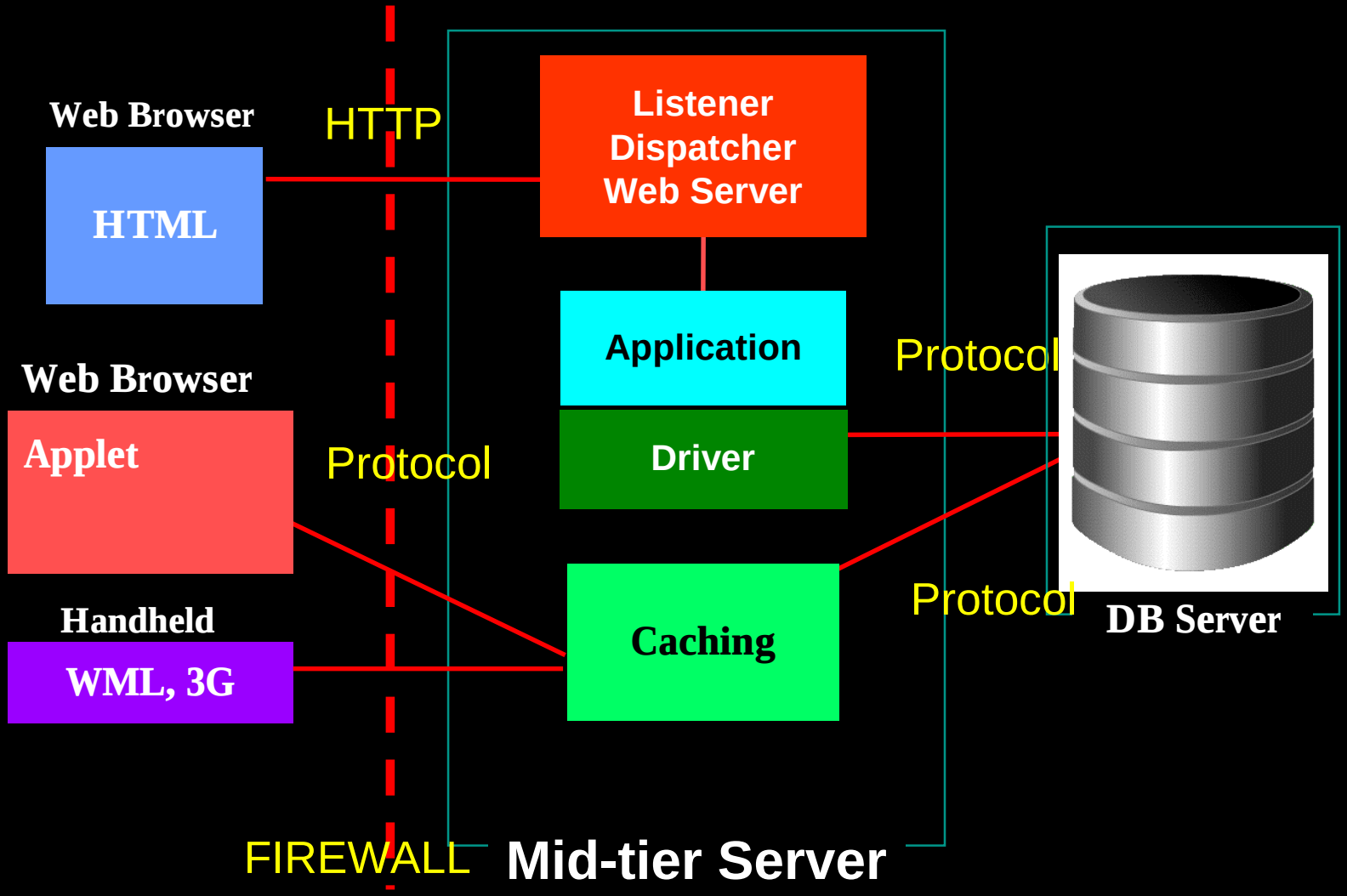
3 **Enterprise Application Integration**

Integrate web sites, ERP, legacy systems, suppliers

4 **Mobile Information Access**

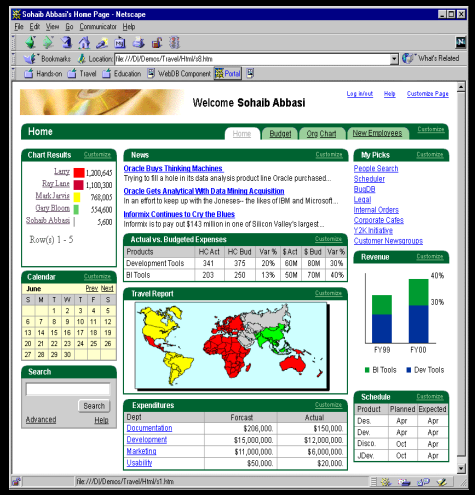
Make web sites accessible from any mobile device

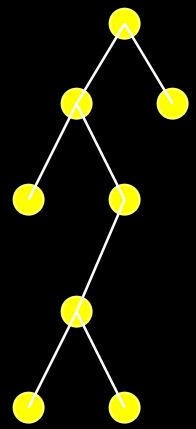
Internet N-Tier Architecture

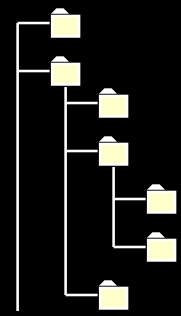


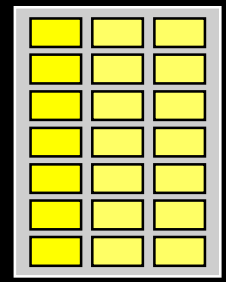
Challenge: Handling Different Models?

Web Content Objects Documents Tables









Interoperability across APIs and component models

Solution: A Universal Language – XML

- The Gartner Group refers to XML over Web protocols as the “Digital Dial Tone”
- Forrester describes XML and HTTP as central parts of “Internet Middleware”
- The W3C (World Wide Web Consortium) states that “The Extensible Markup Language (XML) is the universal format for structured documents and data on the Web.”

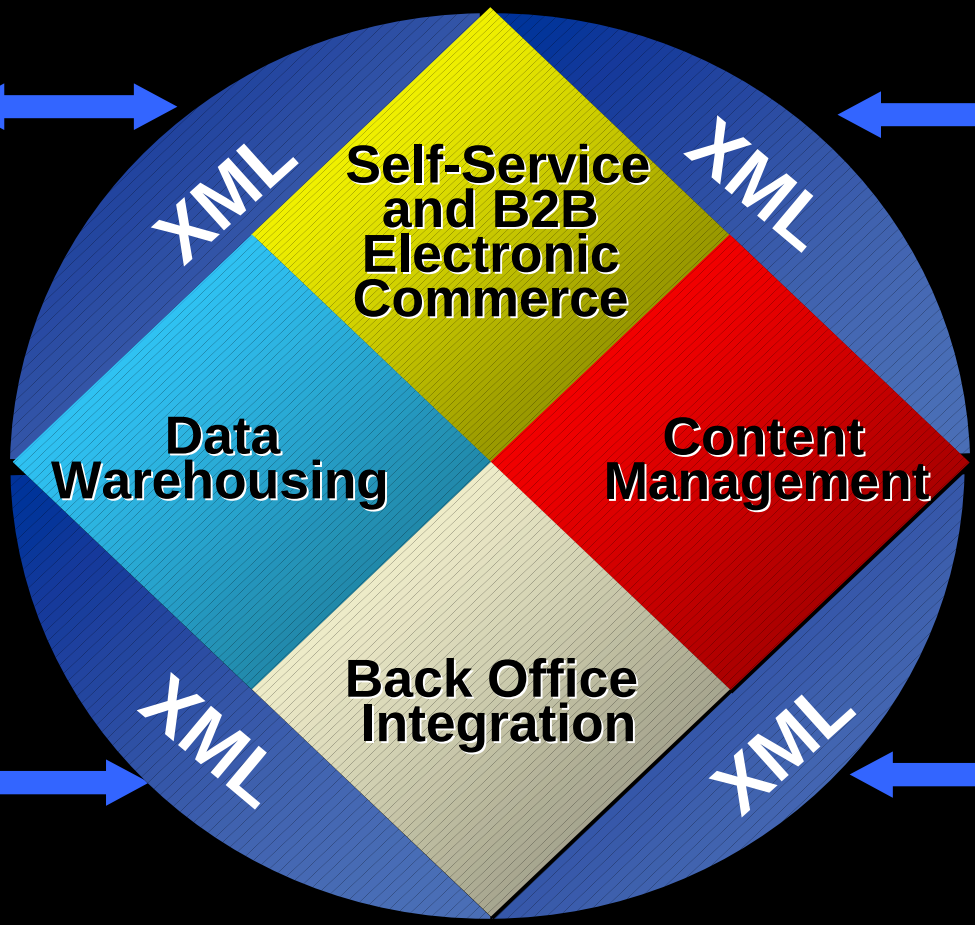
XML is Driving e-Business

- XML Makes Existing Data More Valuable
- Business Application Data interchange through XML Schema
- Dynamic Customized Presentation of Data with XML and XSL
- Databases as XML Information Sources

XML-Powered Applications

Trading Partners

Consumers & Businesses



Service Providers

Content Providers

The XML Family of W3C Standards

Current

- XML 1.0 - Well-formed and valid, self-describing
- XML Namespaces 1.0 - URIs for uniquely qualifying tags and attributes
- DOM 1.0 - Document Object Model (tree-based access)
- SAX 1.0 - Simple API for XML (event-driven access)
- XSLT 1.0 - Extensible Stylesheet Language Transformation
- XPath 1.0 - Syntax for Navigating XML documents

New

- XML Schema 1.0 - simple and complex datatypes
- DOM 2.0 and SAX 2.0

XMLSchema

- W3C Recommendation (May 2, 2001)
  - Beyond DTD (Document Type Definition)
    - Written in XML for documents and data
    - Based on other W3C TR (XPath, Namespace, etc.)
    - Datatypes (primitive, derived, and user-derived)
      - Simple Types: elements with primitives
      - Complex Types: elements with subelements/ attributes
- Similar to SQL99 type system
- Oracle has implemented the standard

XMLSchema & O-R system

XMLSchema Components	Object-Relational System

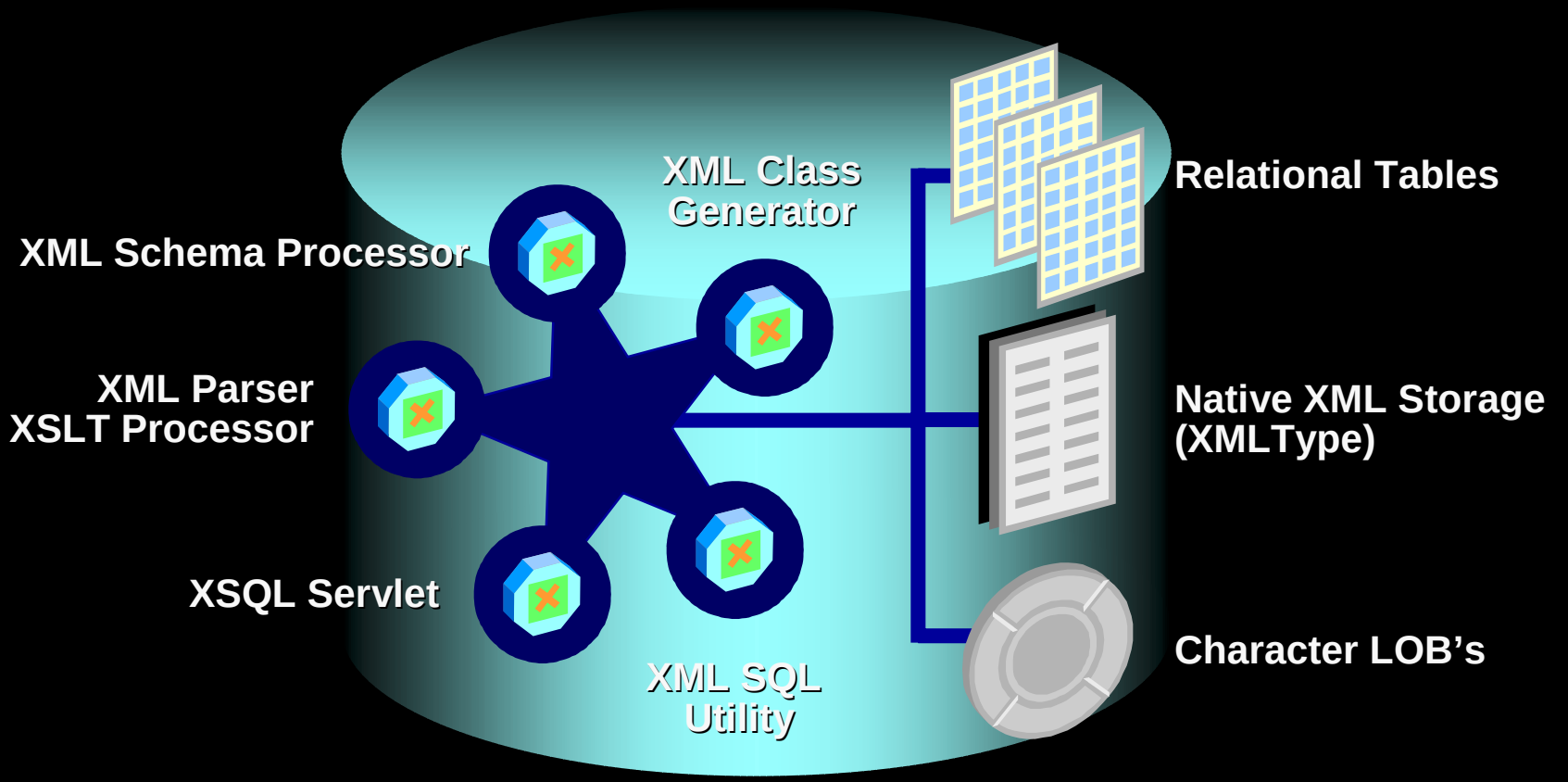
• Complex types, groups
• Derived datatypes
• Lists
• Typing constructs
• XPath navigation
  – /po/pono
• Open content
• Type, schema and element Constraints
	• Object types
	• Type inheritance
	• Collections (Arrays)
	• SQL type constructs
	• O-R traversal
	  – select a.po.pono from..
	• LOBs
	• SQL type and table constraints

- Can define XMLSchema for SQL types

Agenda

- Survey
- Technical Overview
  - XML
  - XDK
  - Database Native XML
  - XML Messaging
- Summary
- Q & A

Built-in XML Developer's Kits (XDK) and Native XML Database Support

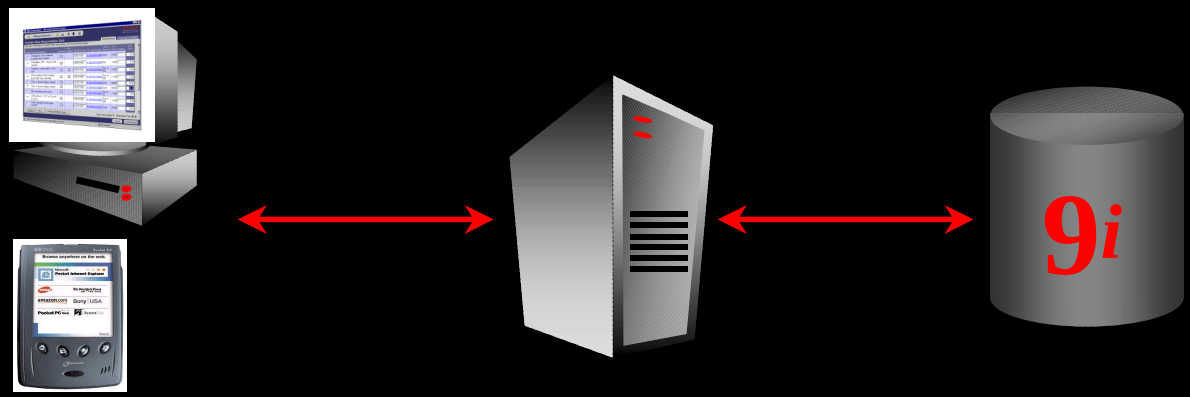


Oracle9i XML Developer's Kits

- XDKs for Java, JavaBeans, C, C++ and PL/SQL
  - XML Parsers now support DOM 2 and SAX 2 providing increased web functionality
  - XSL Processors have improved performance and reduced resource usage
  - New XML Schema Processors add support for simple and complex datatypes
  - XML Class Generators
  - XML Java Beans with Database Access

Deployment Architectural Flexibility

- XML can be used anywhere as it is Transport Protocol Independent
  - Deploy XDK Components in the Database
    - Java XDK Runs in Oracle9i JVM
    - C/C++/PL-SQL XDK linked into the Kernel
  - Deploy XDK Components in a Web/Application Server
  - Deploy XDK Components on a Thin or Thick Client



Application and Server Support

- The Oracle XDKs are ready to deploy or redistribute in applications
  - XDK technical support is included under Oracle Server Support Agreements
  - Stand-alone Support Available will be available through the Oracle Store
  - Support is Backed by DDR and Development Groups
- Consulting Services are XML-trained
- XDKs include a flexible re-distribution and deployment license

XML Schema Processors

- XML Schema 1.0 Compliant
- XML Schema Processors for Java, C/C++
- Simple and Complex Structured Datatypes
- Validation of XML documents and messages for data interchange

XML Class Generators

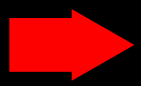
- Versions for Java and C++
- Accepts full range of DTDs and XML Schemas
- Generates Class files for XML Document and Message Creation
- Support for Full Set of Character Encodings
- Optional Validation Mode aids debugging

XDK for Java Beans

- XML DOM Builder Bean with asynchronous parsing support
- XSL Transformer Bean with Stylesheet caching
- XML Source and Tree Viewer Beans for GUI Applications
- New DBView and DBAcess Beans for Database Connectivity

Agenda

- Survey
- Technical Overview
  - XML
  - XDK
  - Database Native XML
  - XML Messaging
- Summary
- Q & A



High Performance Native XML Support

- A new native XMLType datatype
  - SQL query with XPath support
  - Essential Member functions
- Native XML Generation & Storage
  - New SQL functions and packages
- New native URI-Ref datatypes
  - Universal references to database and XML document fragments

New Native XMLType Datatype

- A new native datatype (like InterMedia types)
  - create columns of this type
  - use it in PL/SQL functions and procedures
  - pass it in arguments, etc.

```
CREATE TABLE po_tab (Id NUMBER, total NUMBER(7,2),
po_body XMLType)
```

- Methods operate on the XML content
  - ExtractNode(...) extracts a specific node
- Abstraction for storing XML documents

XMLType Methods

XMLType Methods	Description
getClobVal	Return the contents of the XMLType as a CLOB value
isFragment	Is the document really a fragment?
getStringVal	Gets a string value from a XML Node
getNumVal	Gets a numeric value from a XML Node
extract	Extracts a portion of the document using a XPath-like syntax, returning a XMLType
existsNode	Checks if there are any resultant Nodes in the XPath expression

Indexing XMLTypes

- ExistsNode, ExtractNode can be speeded up by building functional indexes

```
CREATE INDEX idx on
  po_tab(po_body.extract('/PO/@PONO').getNumVal());
```

- Text indexes can also be built

```
CREATE INDEX idx1 on po_tab(po_body).getNumVal());
```

```
SELECT extract(p.po_body, '/po/cust/custname')
.getStringVal() FROM po_tab p WHERE
CONTAINS(p.po_body/Remarks, 'URGENT') >=1 AND
existsNode(p.po_body, ''//po/cust'') >= 1;
```

Benefits of XMLType

- Brings the XML and SQL worlds together
  - SQL operations on XML content
  - XML operations on SQL content
- Convenience
  - built in functions, indexing, navigation etc.
- Abstraction over different XML storage models

Native XML Generation and Storage

- New SQL operators
  - SYS_XMLGEN and SYS_XMLAGG
  - Generate XML documents from tables and columns
- New DBMS_XMLGEN package
  - Creates an XML document from any SQL query and gets the document as a CLOB
- New Table function
  - Decomposes and stores XML documents

SYS_XMLGEN

- Converts a scalar value, UDT instance, or XMLType instance to an XML document
- An optional XMLGenFormatType object specifies formatting options for the result
- Returns an XMLType
- Is used to create and query XML instances *within* SQL queries.

```
SELECT SYS_XMLGEN(empno*2) FROM emp where ename
                    LIKE 'Scott%';
```

```
<?xml version=''1.0''?>
   <EMPNO>60</EMPNO>
```

SYS_XMLAGG

- Aggregates inputs into a single XML document
  - aggregate (group) related XML data
  - aggregate (concatenate) fragments

```
SELECT SYS_XMLAGG(SYS_XMLGEN(ename),
SYS.XMLGENFORMATTYPE.createFormat('EmployeeGroup')).getC
lobVal() FROM emp GROUP BY deptno;
```

```
<EmployeeGroup>
        <ENAME>Scott</ENAME>
        <ENAME>Mary<ENAME>
</EmployeeGroup >
<EmployeeGroup >
        <ENAME>Jack</ENAME>
        <ENAME>John>/ENAME>
</EmployeeGroup >
```

DBMS_XMLGEN

- Creates an XML document from any SQL query gets the document as a CLOB
- Provides a ‘fetch’ interface with maximum rows and rows to skip – useful for pagination in web applications
- Provides options for changing tag names for ROW, ROWSET etc.

DBMS_XMLGEN Example

```
CREATE TABLE temp_clob_tab(result CLOB);
DECLARE
       qryCtx DBMS_XMLGEN.ctxHandle;
      result CLOB;
BEGIN
       qryCtx := dbms_xmlgen.newContext('SELECT * from
EMP;');
       DBMS_XMLGEN.setRowTag(qryCtx, 'EMPLOYEE');
       DBMS_XMLGEN.setMaxRows(qryCtx, 5);
       LOOP
              result := DBMS_XMLGEN.getXML(qryCtx);
       EXIT WHEN
       DBMS_XMLGEN.getNumRowsProcessed(qryCtx)=0);
       INSERT INTO temp_clob_tab VALUES(result);
       END LOOP;
END;
```

Generated XML

```
<?xml version=‘’1.0’’?>
<ROWSET>
       <EMPLOYEE>
              <EMPNO>30</EMPNO>
              <ENAME>Scott</ENAME>
              <SALARY>20000<SALARY>
       </EMPLOYEE>
       <EMPLOYEE>
              <EMPNO>31</EMPNO>
              <ENAME>Mary</ENAME>
              <AGE>25</AGE>
       </EMPLOYEE>
</ROWSET>
```

TABLE Functions

- Model any arbitrary data (internal to the database or from an external source) as a collection of SQL rows
- Executed in parallel pipeline for performance
- Decomposes XML into SQL rows which can be consumed by regular SQL queries and inserted into regular relational or object-relational tables

New Native URI-Ref Datatype

- URI Reference: Universal Resource Identifier for XML fragment
- URIType: An abstract type with subtypes to reference local or remote data
  - DBURIType: access local data in a database
  - HTTPURIType: access remote data via HTTP
- Used in XPath syntax for XML navigation

URIType Methods

URIType Method	Description
getClob	Returns the value pointed to by the URL as a character LOB value.
getURL	Returns the URL stored in the URIType.
getExternalURL	Similar to getURL, but calls escaping mechanism to escape characters according to the URL specification

Using URITypes

```
CREATE TABLE tax_deductible_po_tab ( po_uri UriType,
   max_deduction NUMBER(7,2), poName VARCHAR2);

INSERT INTO tax_deductible_po_tab VALUES (UriFactory.getURL
('/scott/po_tab/row[pono=10]'), 2500.00, 'Scott's DB PO)';

INSERT INTO tax_deductible_po_tab VALUES (UriFactory.getURL
('http://proxy.oracle.com/webaccts/pos/scott/po1), 1000.00,
   'Scott's Web PO');

SELECT e.pouri.getClob()
   FROM tax_deductible_po_tab e);
```

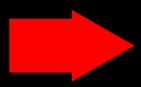
- The UriFactory package contains methods to generate relevant URIs

HTTP Access for DBURIRefs

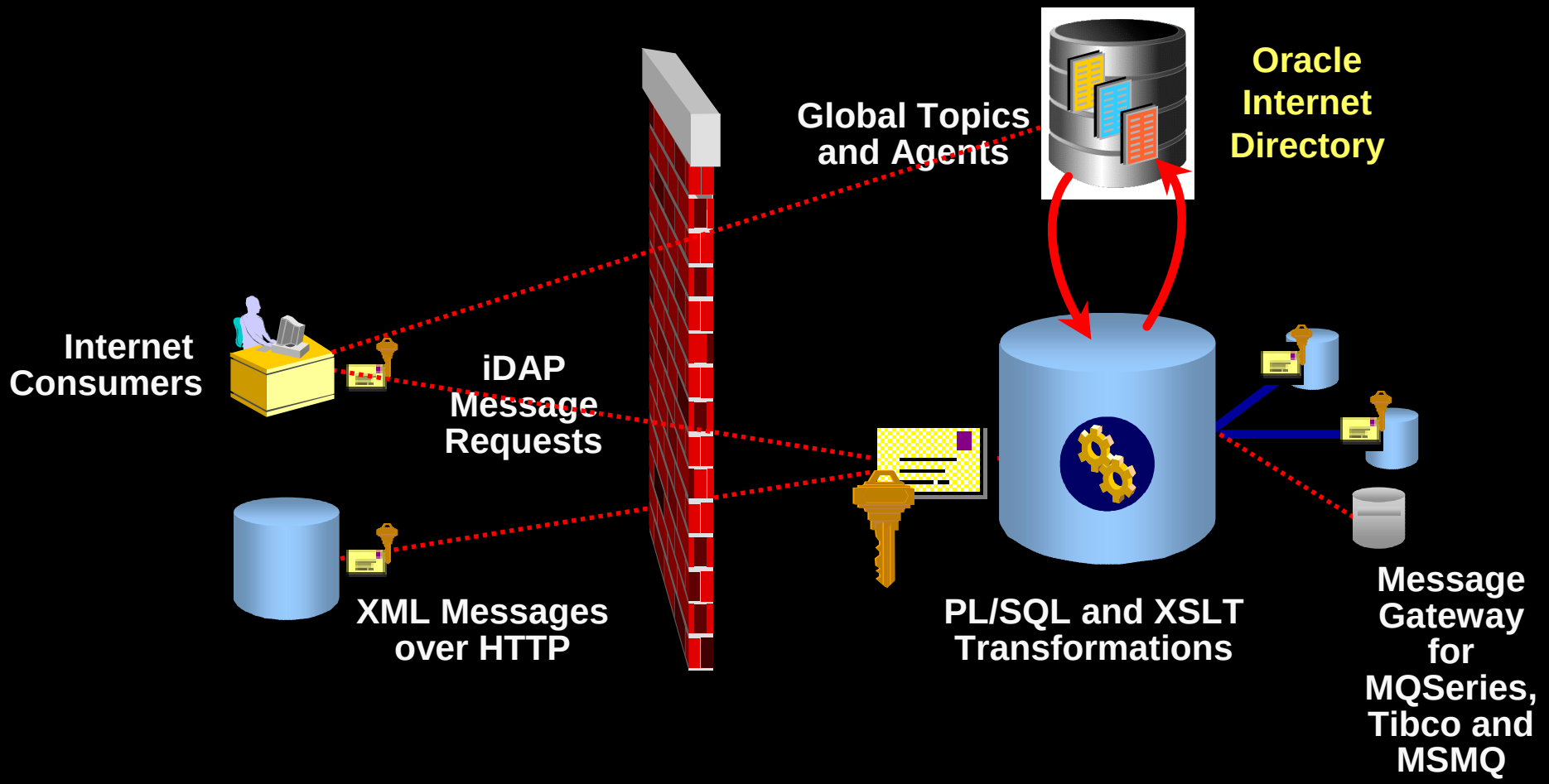
- Access DB references from a Web Browser
- Servlet support for DBURIRefs
  - A default servlet provided with 9i runs in the Oracle Servlet Engine
  - oracle.xml.dburi.OraDbUriServlet()
    - http://machine.oracle.com:8080/oradb/scott/emp/row[empno=7369]/ename
    - generates xml content in output stream
    - Runs in SYS/DBUser realm
- Custom servlet can be implemented for other Servlet Engine to execute DBUriRef calls

Agenda

- Survey
- Technical Overview
  - XML
  - XDK
  - Database Native XML
  - XML Messaging
- Summary
- Q & A



Global XML Messaging

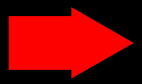


AQ Enhanced XML Messaging

- Messages can be transformed from one type to another
  - Object to XML, XML to Object, XML to XML
- Users define attributes in the destination message as expressions involving attributes in the source message
  - e.g., Subscription based on existNode()
- Expressions can be in either SQL, PL/SQL, Java or XSLT (for XML)
- Transformations can be called when messages are enqueued, dequeued or propagated

Agenda

- Survey
- Technical Overview
- Summary
- Q & A



Summary

- Full XML Standards support for XML-enabled Internet application development on all Oracle 9i platforms
- High performance database-native XML support for storage and retrieval of XML data and documents
- XML Messaging and Transformation support in AQ provide a centralized, easy to manage, secure infrastructure for global messaging

For Further Information . . .

- Oracle Technology Network
  - otn.oracle.com/tech/xml

Oracle Technology Network

- Complete technical resource for Internet developers
- XDK and utility downloads
- XML online technical discussion forum
- Sample downloads & live demos
- Whitepapers, technical information, documentation

http://otn.oracle.com

Related Sessions and Demos

- XML in Oracle9i in Oracle9i DEMOgrounds
- UML Design Patterns
  - Wed., 8:30 - 9:30, Hall 2.1
- Oracle C++ Call Interface
  - Wed., 14:30 - 15:30, Hall 2.3


QUESTIONS
&
ANSWERS


ORACLE®
SOFTWARE POWERS THE INTERNET™