ODTUG Kscope11

LONG BEACH, CALIFORNIA * JUNE 26-30

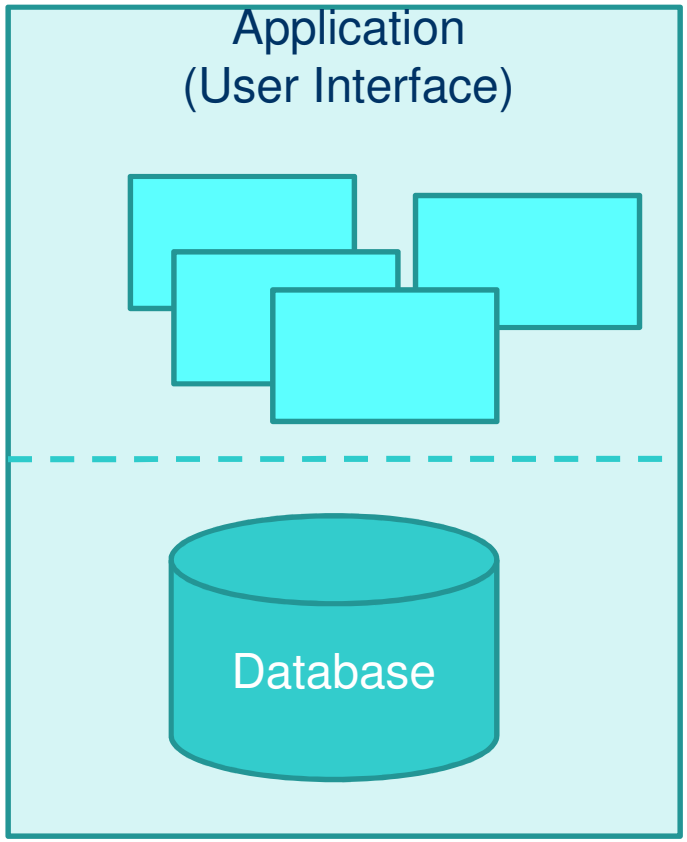
Bridging the Gap between SOA and the Database

Peter Ebell
AMIS

Agenda

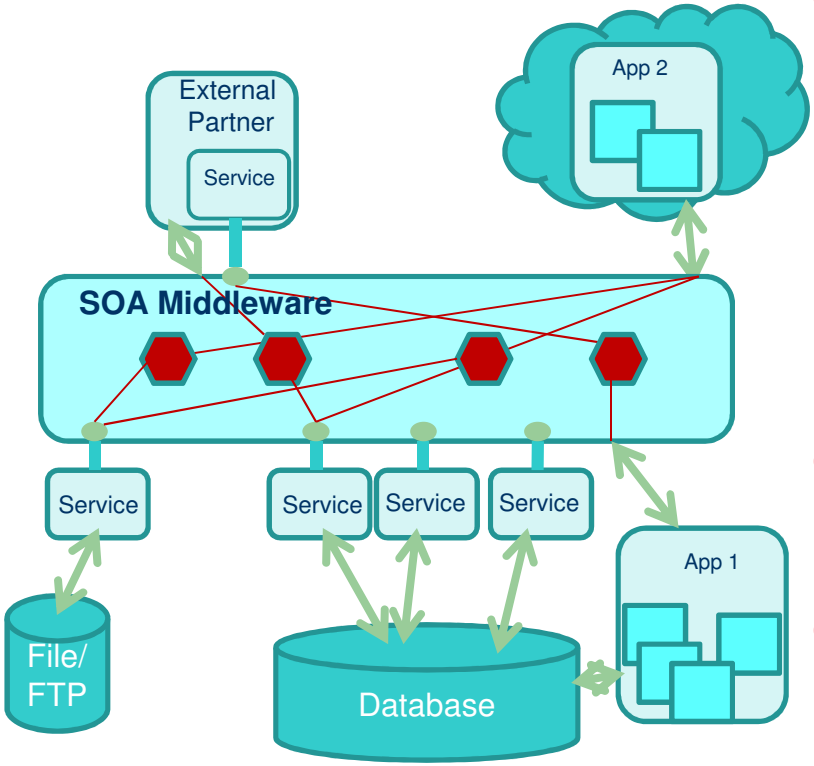
- Two different worlds: Database and SOA?
- Bridging the Gap
  - How the Database reaches out to SOA Middleware
  - How SOA Middleware reaches out to the Database
- Where the Two Worlds Meet

Introducing: The Oracle Database



- The “traditional” database
  - Contains Data
  - Contains Business Logic
  - Speaks SQL and PL/SQL
  - Connects via SQLNet/JDBC/ODCB
- Applications “talk” to it directly
- Has proven to be the most central and consistent component in the infrastructure

Introducting: The SOA Middleware



- SOA Middleware
  - Is All About (Web) Services
  - Contains Process and Business Logic
  - Speaks XML
  - Connects via HTML/FTP/….
- Everything is very much “decoupled”
- Is a very different world indeed.

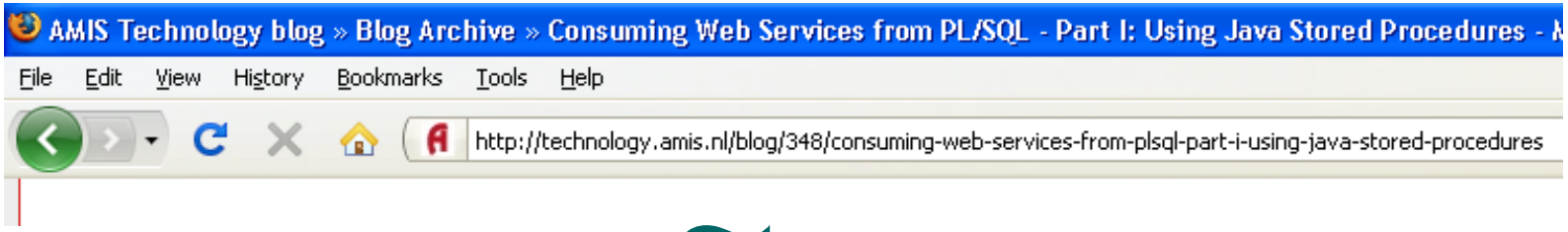
ODTUGKscope11

Bridging The Gap – In Two Directions!

SOA from the Database

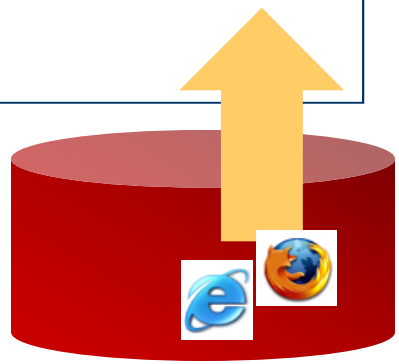
- The Oracle database is equiped with essential tools to interact with the SOA environment:
  - It can do HTTP (calling and publishing)
  - It can do XML (creating and interpreting)
  - It can do Events & Queuing
  - It can do Files
  - It can do Email
  - ….

The Database is a Browser too...



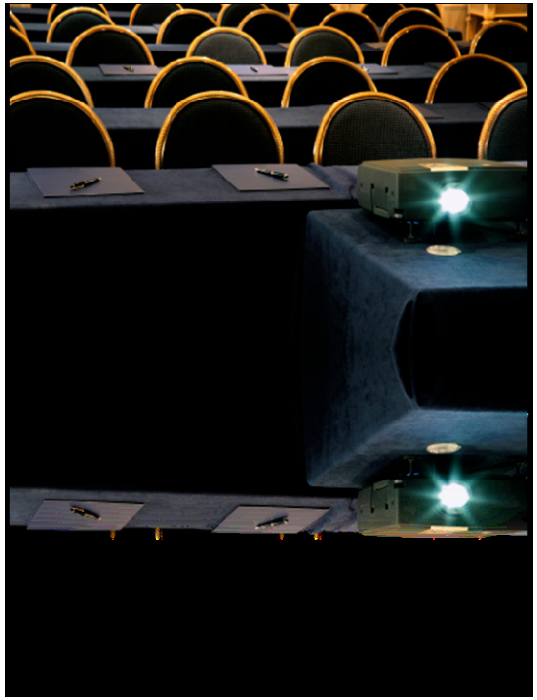
≈

```
select  utl_http.request
        ( ' http://technology.amis.nl/blog/348/consuming-'
        ||'web-services-from-plsql-part-i-using-java-'
        ||'stored-procedures '
        , null
        )
from    dual
```



The Database is a Browser too…

Demo: UTL_HTTP



Database acting as browser using utl_http

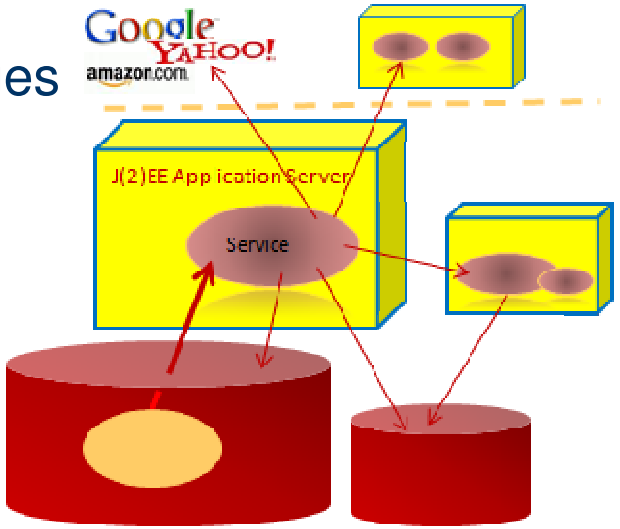
DB can “reach out” out through UTL_HTTP

- Application Server
  - Servlet
  - (intranet) WebService
  - BPEL Process
  - ESB Service
- Internet (there are some “decoupling” issues here!)
  - RSS feed
  - WebService
  - Web Site

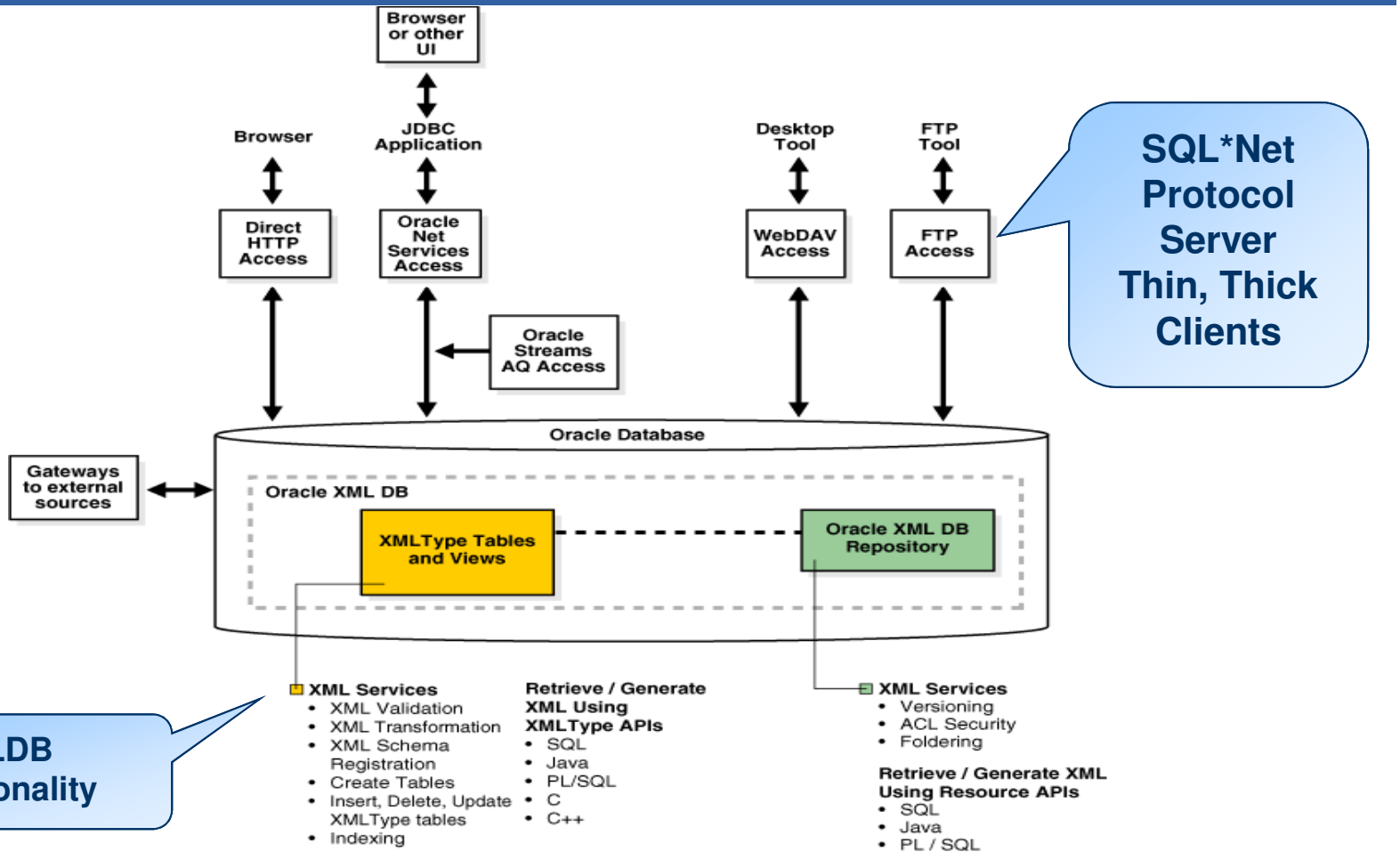
Note: HTTP Get requests can also be performed by:
- dbms_xmlparser.parse(url)
- HTTPURITYPE(url)/DBURITYPE
- Oracle Text (index external docs)
- utl_tcp, stored Java
- utl_dbws

Think Services, Services, Services!

- The database can ask the middle tier to:
  - Get information from (or to) services on the intranet or internet ('please get us the price of a barrel of oil')
  - Publish/Send information to the internet (RSS, email, chat)
- Can tell the middle tier
  - About interesting events like data changes
    - Trigger cache refresh!
  - Alert about (im)pending issues, attempted rule violations, …
- Typically, these services talk "XML"
  - Luckily, the database can do that too!



Oracle XML DB



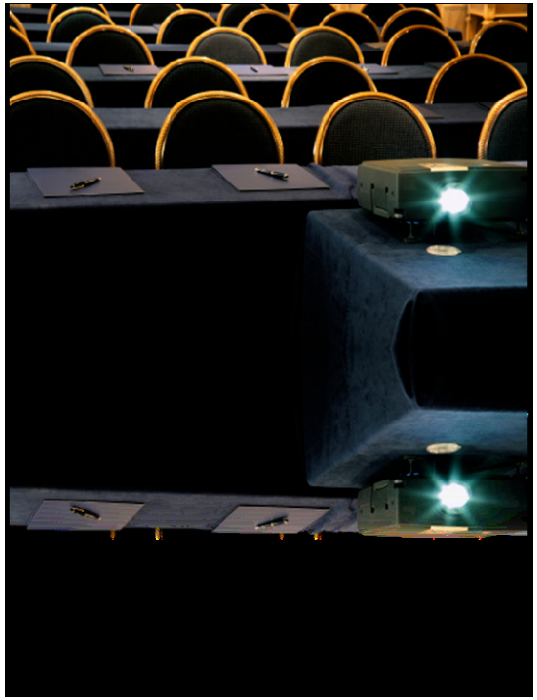
XML DB: Mayor Features

- Storage based on the **XMLType** datatype
  - XMLType Column, XMLType Table
- **Retrieval** of data via XML/SQL, XPath, XQuery
- **Security** based on ACL, Oracle Roles
- **The Protocol Server**
  - HTTP(s), FTP, WebDAV, Native Databas WebServices (NDWS)
- The XMLDB **Repository**
  - XMLSchema Support and Evolution
  - Versioning, CMS Features

Publish RSS as "View" inside database

```
create view AMIS_WEBLOG_FEED as
select title
,      link
,      author
,      to_date(substr(publication_date,6,20)
       ,'dd mon yyyy hh24:mi:ss') timest
from   xmltable
       ( XMLNAMESPACES('http://purl.org/dc/elements/1.1/' AS "dc")
       , 'for $i in //item
          return <Article>{$i/title}{$i/dc:creator}
                          {$i/pubDate}{$i/link}
                 </Article> '
       passing httpuritype('http://technology.amis.nl/blog/?feed=rss2').getXML()
       COLUMNS
         title            varchar2(100) path 'title'
       , link             varchar2(100) path 'link'
       , author           varchar2(100) path 'dc:creator'
       , publication_date varchar2(100) path 'pubDate'
       )
```

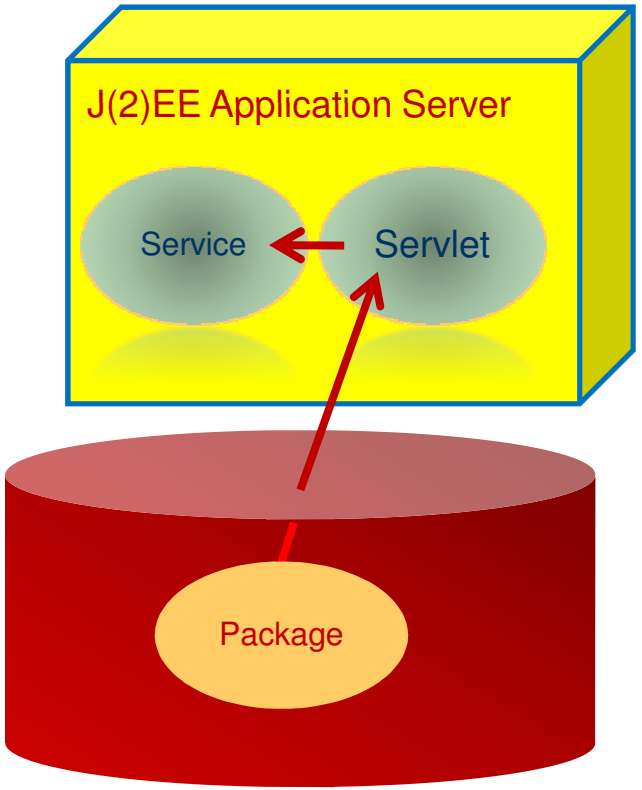
Demo: RSS Feed View



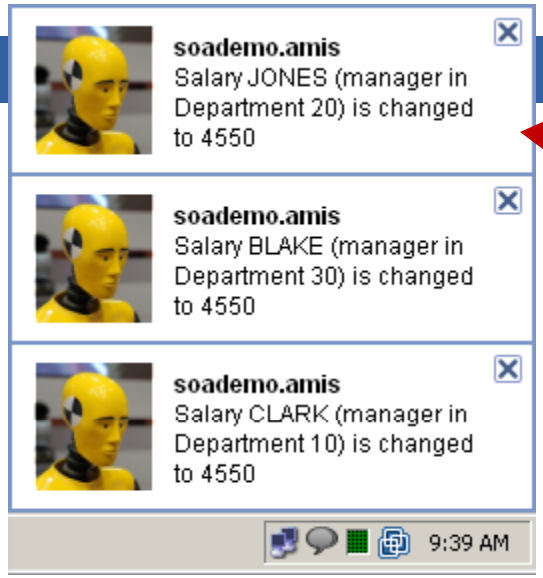
Database View based on RSS Feed

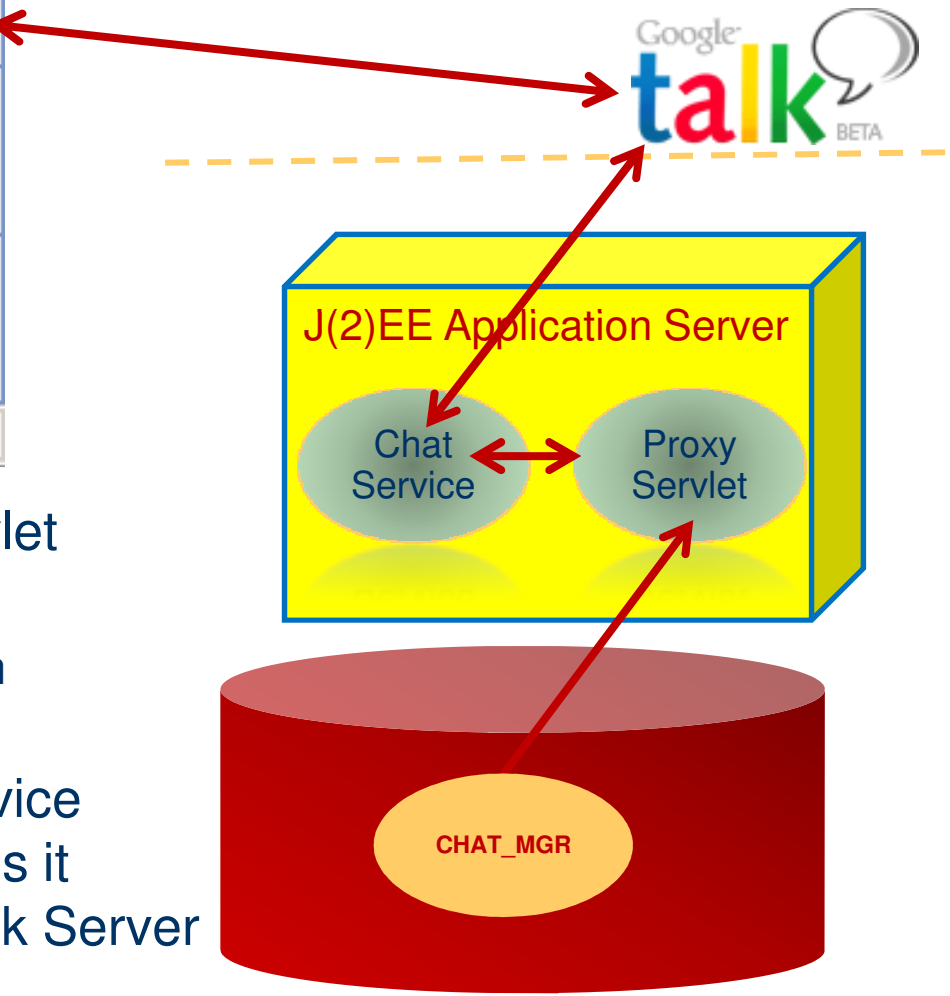
Design Pattern: Servlet "Proxy"

- Java has many libraries and facilities for calling Services (and other middle tier resources)
- We can leverage this from the database using UTL_HTTP
  - Create a simple servlet
  - That can be called through a simple URL
  - And does very complex stuff
  - Call with UTL_HTTP and receive results from complex middle tier and beyond actions



Example: IM from the Database

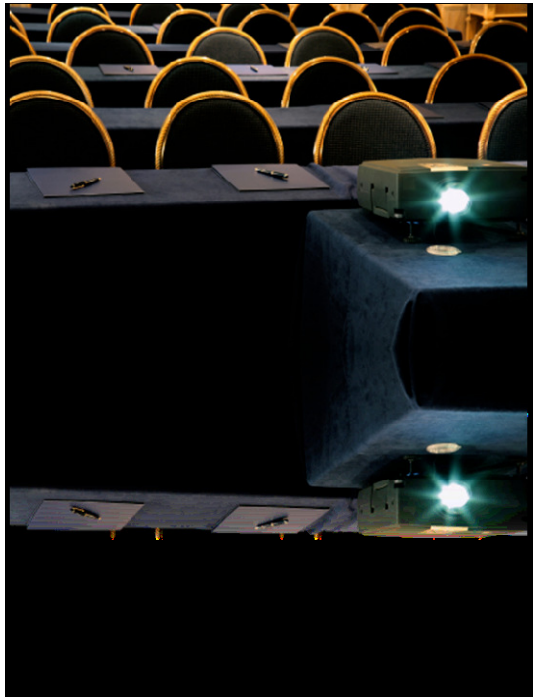




- Database calls Proxy Servlet using UTL_HTTP
  - Passing in the destination and the message content
- This servlet invokes a Service which communicates sends it onwards to the Google Talk Server

ODTUGKscope11

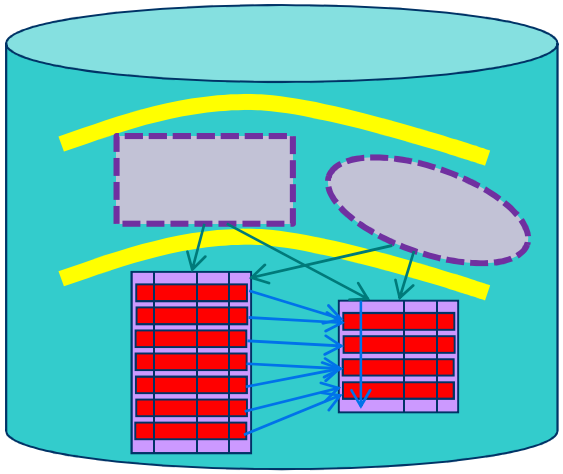
Demo: IM from the Database



Notifying data changes via Instant Messaging

Offering "SOA-ready" Services

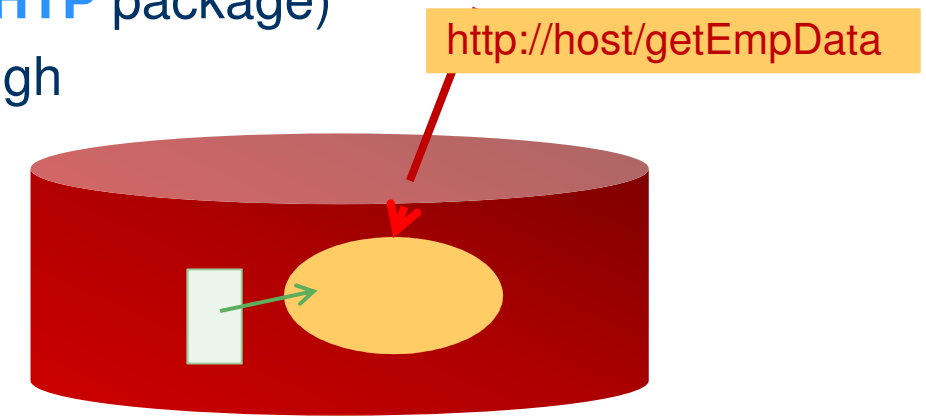
- Build APIs from Views (with IOT) and PL/SQL Packages
  - Design from Contract, Hiding the data model, centralizing the SQL => managing dependencies
- Publish URL (HTTP) based services (instead of SQL or PL/SQL)
  - HTTP Get and Post Requests
  - XML answers



- Publish full blown WebServices
  - "Manually" using dbms_ws
  - Publish through 11g Native Web Services

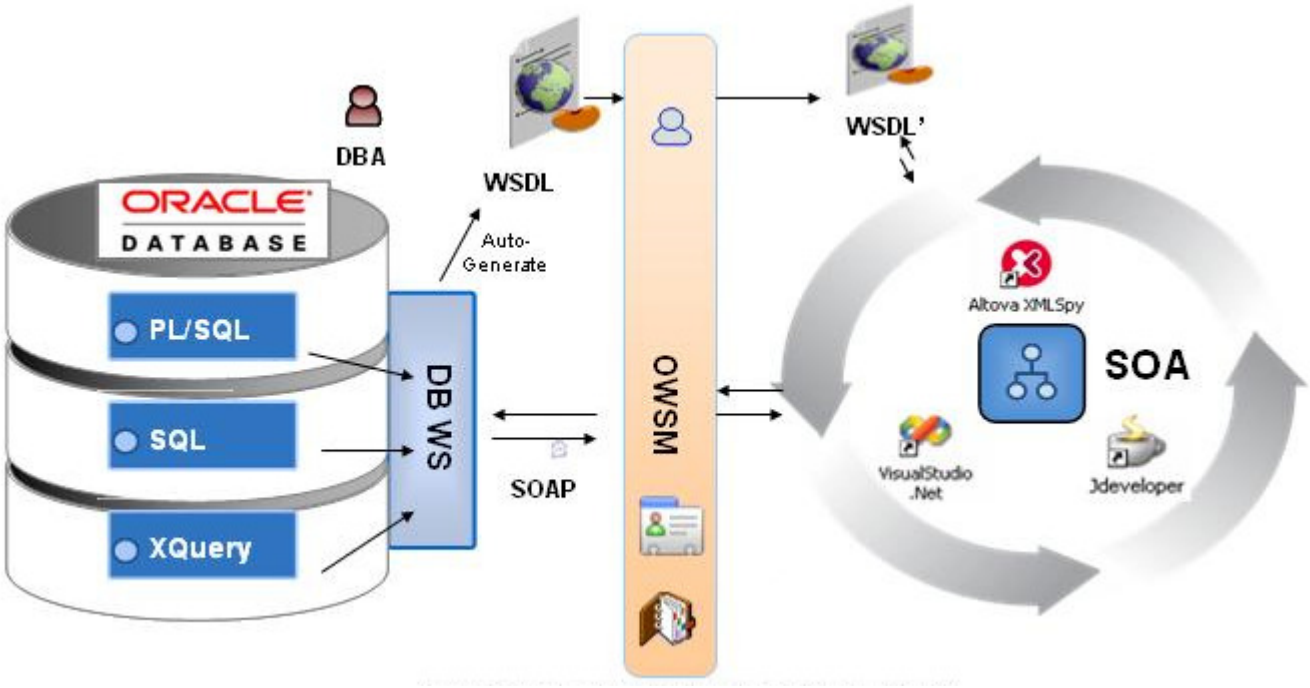
Publish Http Data Service

- The steps to create an HTTP Data Service that returns XML based data in response to HTTP requests
  - create the SQLXML queries that create the desired XML document
  - transform the raw XML result to as proper format
  - create a package that returns the XML data through the PL/SQL Web Toolkit (the **HTP** package)
  - Publish the package through the embedded PL/SQL gateway (**DBMS_EPG**)



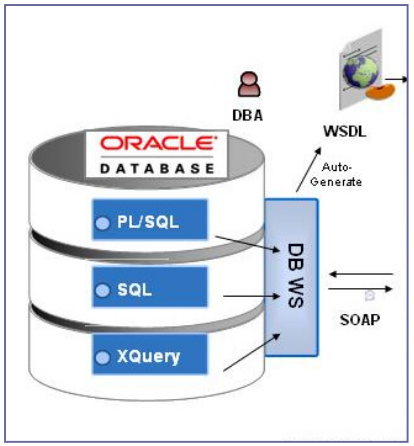
Oracle 11g Native WebServices

- Publish a PL/SQL Package as externally available WebService
  - WSDL is auto-generated
  - SOAP calls over http and https are supported

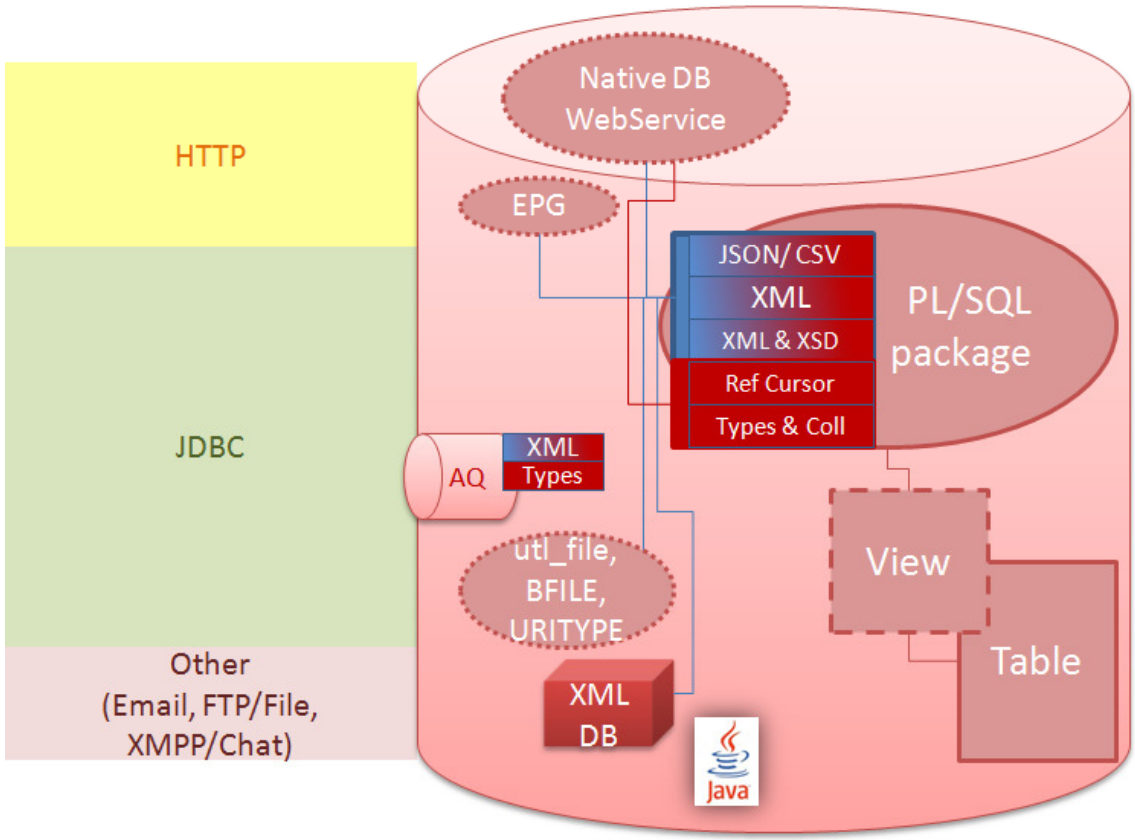


11g Native Web Services

- An arbitrary SQL query request can be sent to the main 'oawsv' endpoint, and the results are returned as XML in the SOAP response
  - Provided the request is authenticated with user name and password
- All PL/SQL program units are automatically exposed as SOAP WebServices
  - Dynamically generated WSDL and XSD
  - SOAP calls handled by the database



Interaction Overview - Database



ODTUGKscope11

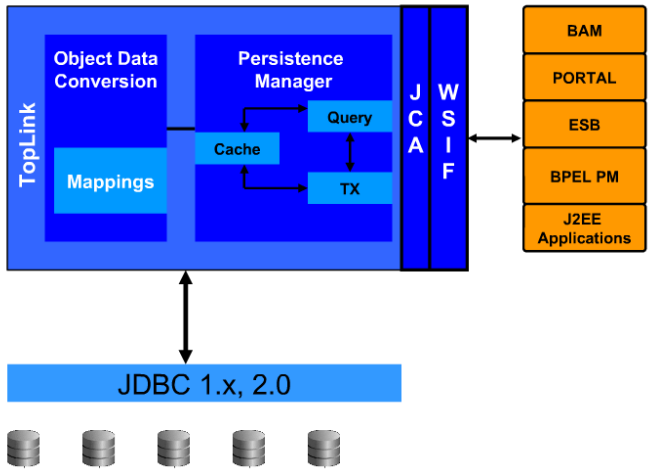
Bridging The Gap – In Two Directions!

SOA To The Database

- Oracle Fusion Middleware is packed with features to facilitate communicating with Databases:
  - Database Adapter
    - Insert/update/delete
    - Retrieve/Query By Example
    - Detect data changes
  - AQ Adapter
  - ADF BC Web Services
  - SDO Data Objects

Database Adapter

- Available as (JCA) service to all Fusion Middleware products (OSB, SOA Suite, ..)
- Leverages proven TopLink technology
- Caching for High Performance
- Supports JTA Transactions



Database Adapter - Features

- Declarative, wizard-driven development
  - No coding of any kind required!
- Performs all “XML-to-Relational” mappings
  - Query Results
  - ADT Object Types
  - PL/SQL native data structures
  - Works two ways!

DB Adapter - Operations (Outbound)

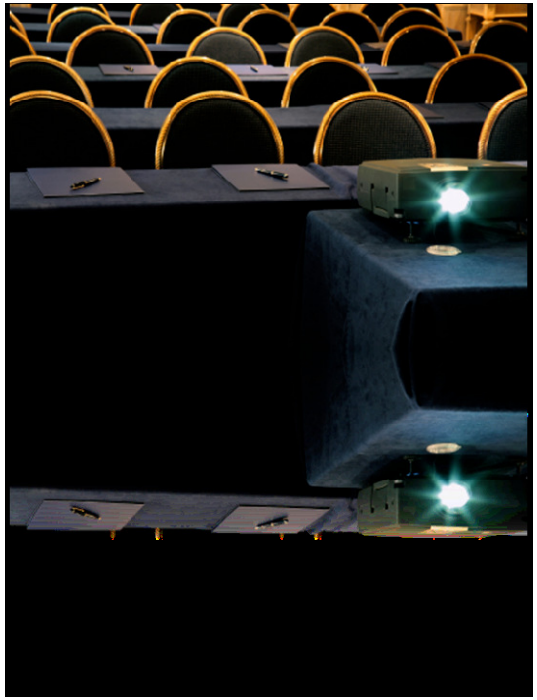
- Call Stored Procedure Or Function
- Perform DML operation on a Table
  - Insert
  - Update
  - Merge
  - Delete
- Execute Pure SQL

DB Adapter - Operations (Outbound)

- Querying
  - Retrieve object using Unique/Primary key
  - Retrieve information from many tables in one operation (both “lookup” and “detail” tables)
  - Parametrize queries
  - Declaratively create joins and where clauses using Query Builder
  - Alternatively, use “Execute Pure SQL”
- Query By Example

ODTUGKscope11

Demo: Database Adapter



Queries and Transactions

DB Adapter - Operations (Inbound)

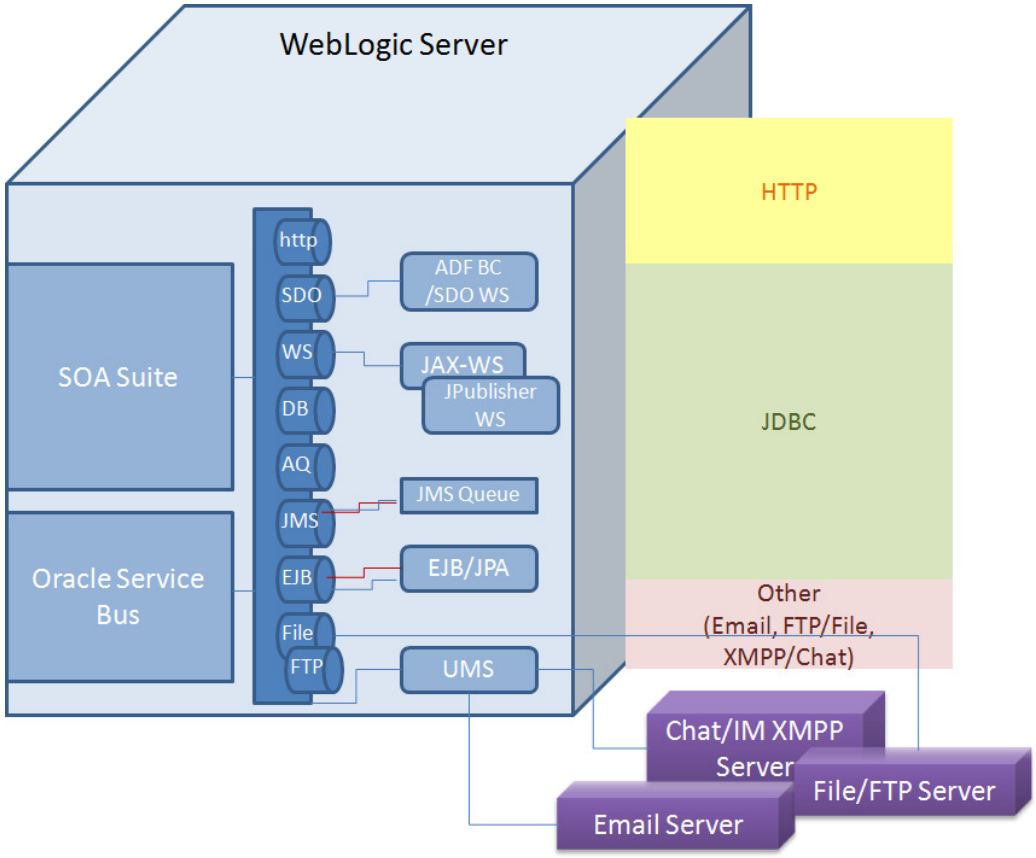
- Poll for new or changes records in a Table
- After read:
  - Delete the rows that were read
  - Update a field in the row (logical delete)
  - Update a sequencing Table or File
- Like with Query, data from lookup and detail tables can be retrieved automatically

ODTUGKscope11

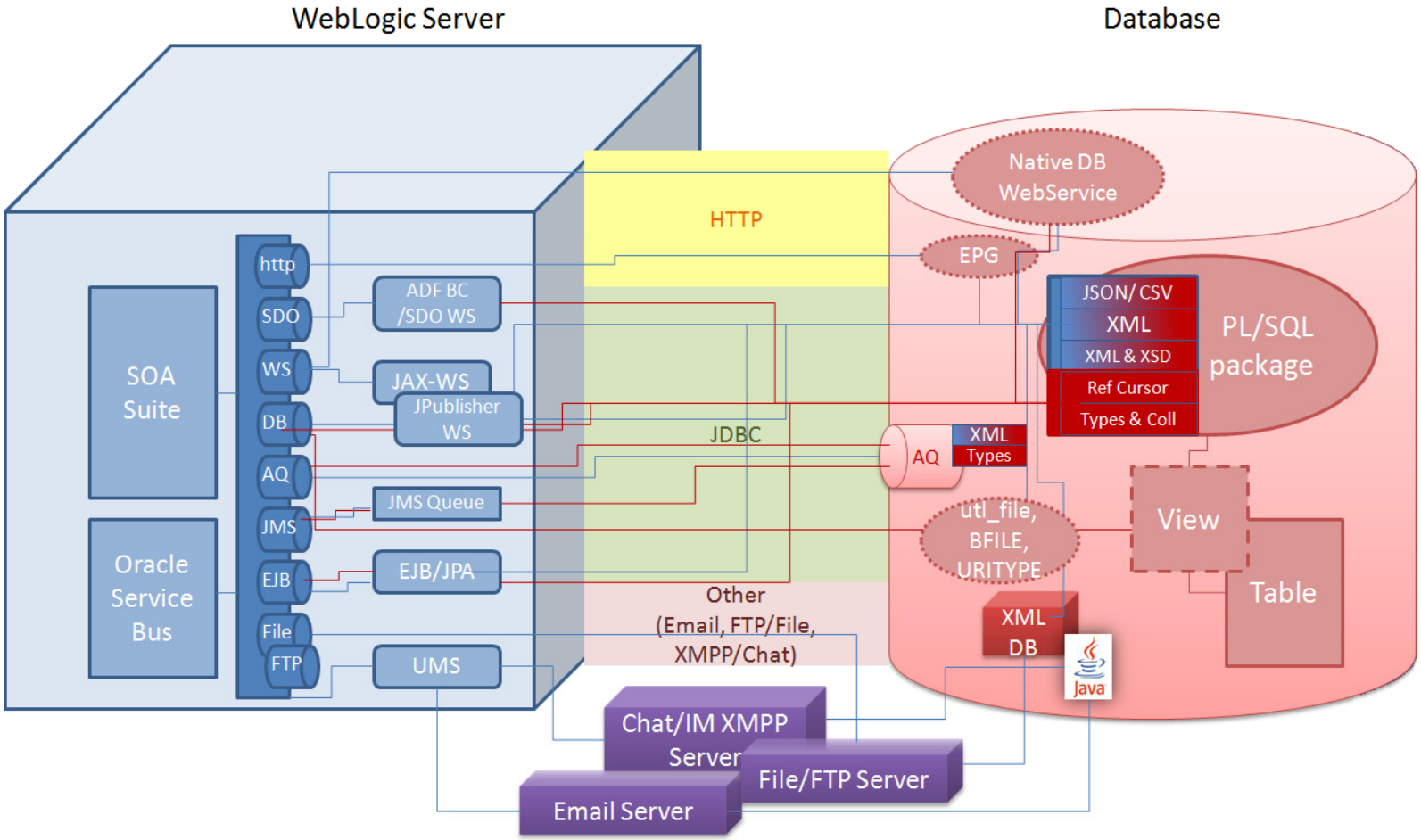
AQ Adapter

What Is SDO

- SDO architecture is based on disconnected Data Graphs. Data graphs are tree structures of Data Objects retrieved from data sources, which are then accessed and modified by the client as part of its data processing logic. These changes are then persisted to the underlying data source in a consistent manner, ensuring that data in the data source that is being updated is not stale. These changes are detected and automatically applied by the SD without the client having to invoke and update operation explicitely.


WebLogic Server
HTTP
JDBC
Other
(Email, FTP/File,
XMPP/Chat)
SOA Suite
Oracle Service
Bus
http
SDO
WS
DB
AQ
JMS
EJB
File
FTP
ADF BC
/SDO WS
JAX-WS
JPublisher
WS
JMS Queue
EJB/JPA
UMS
Chat/IM XMPP
Server
File/FTP Server
Email Server

Where The Two World Meet



ODTUGKscope11

Common Scenarios & Best Practices

ODTUG Kscope11

LONG BEACH, CALIFORNIA * JUNE 26-30

Bridging the Gap between SOA and the Database

Peter Ebell
AMIS