Universal Web Services for Providing Enterprise Data

Petr PALAS
PortSight Software Architect
petrp@portsight.com
www.PortSight.com

www.PortSight.com

Agenda

- **Motivation for Universal WS Tier**
- **Architecture Overview**
- **Issues & Solutions**
- **Demo**
- **Benefits and Further Development**

Motivation for Universal WS Tier

- There's a need to share data with partners and customers, to have 360° view of data
- Users want to access data from Enterprise Portals and mobile devices
- Various data sources (ERP, CRM, CMS...)
- Building ad hoc WS on top of each system:
  - much effort, various interfaces
  - cannot provide joined view
  - more complicated management

www.PortSight.com

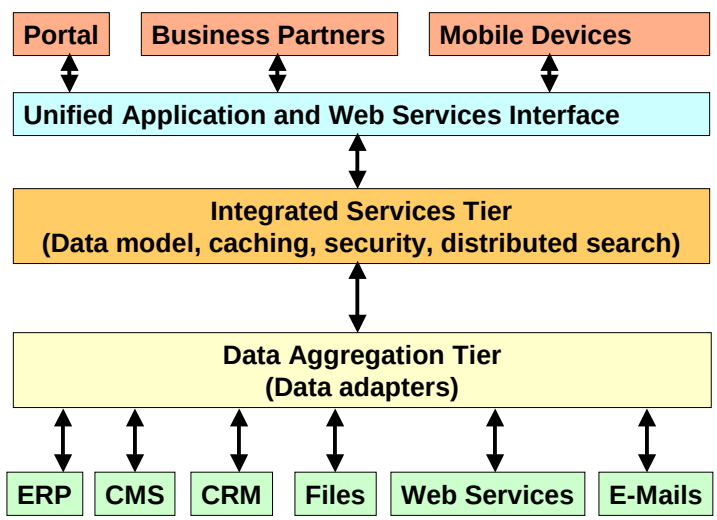
EAI versus Information Integration

- Enterprise Application Integration – sharing data between applications
- Information Integration – sharing data between applications and users
  - BEA Liquid Data
  - IBM Xperanto
  - Nimble Integration Suite
  - and others... (SQL Server 2003?)

Architecture Overview (1)

- Architecture Requirements:
  - Data from various sources (structured, semi-structured and unstructured)
  - Unified interface for reading, searching and updating data
  - Unified security and metadata management
  - Standards-based
  - Short response time

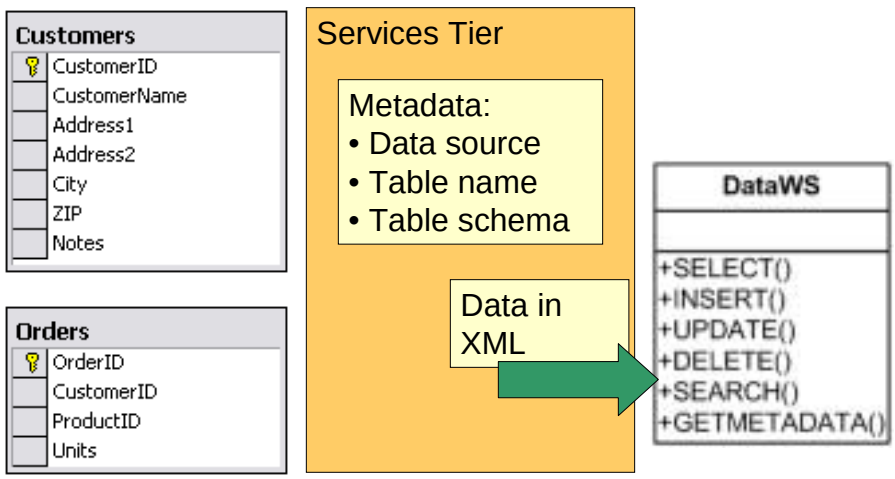
Architecture Overview (2)



First Idea (1)

- Describe data source connection
- Store name and schema of the table
- Publish it through WS

First Idea (2) - Overview



First Idea (3) – DataWS Interface

- SELECT(Table, Conditions) as XML
- UPDATE(Table, UpdatedData)
- INSERT(Table, InsertedData)
- DELETE(Table, Conditions)
- SEARCH(Table, Conditions, SearchConditions) as XML
- GetMetadata(Table) as XML containing XML Schema

Issue 1 – Joining Data (1)

Your distributor wants to see customers from Prag and their orders of products with ID=5.

Option 1: XQuery

- ☺ You can join semi-structured XML data from various sources
- ☹ It still isn't standard
- ☹ May be slow for large amounts of data

Issue 1 – Joining Data (2)

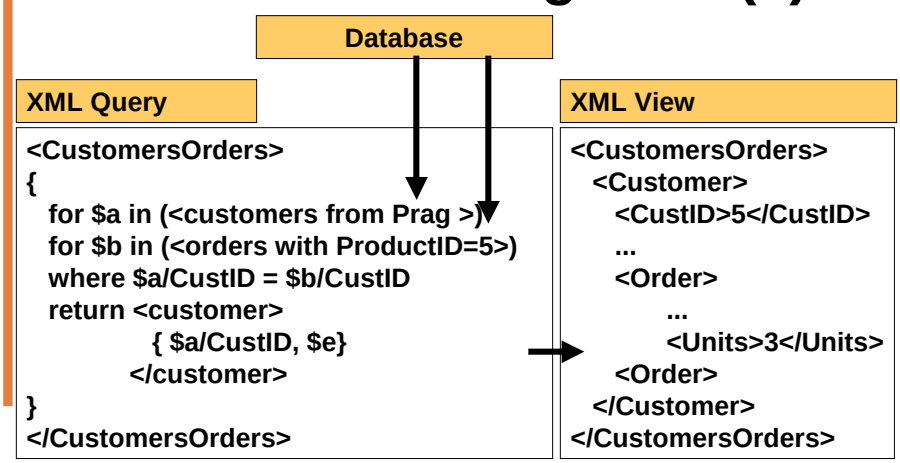
Option 2: XML support in your RDBMS

- ☹ Your RDBMS must support it for distributed queries
- ☺ Optimized

Option 3: XML support in ADO.NET (or other data access library)

- ☹ Not so optimized as RDBMS
- ☹ Not so flexible for semi-structured data

Issue 1 – Joining Data (3)



www.PortSight.com

Issue 2 – Exploring Data

Your partners/employees don't know your data structures.

Solution: Query Builder with proprietary XML-based language and user interface

- Can be mapped to XQuery or SQL (optimization)

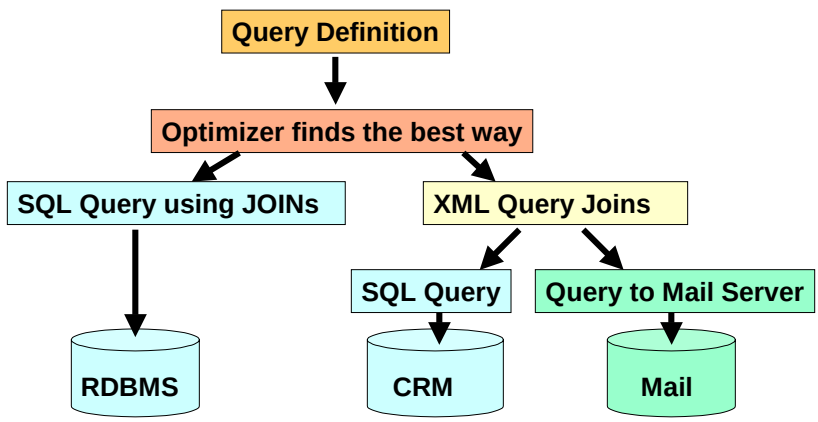
```
<customers>
      <where>City="Prag"</where>

<fields><field>CustomerName</field></fields
>
    <orders>...</orders>
</customers>
```

www.PortSight.com

Issue 3 – Optimization

Query Optimization:



Demo

What we will use:

- **Microsoft .NET Framework**
- **Microsoft XQuery Prototype**
- **Microsoft Excel to consume WS data**

www.PortSight.com

Issue 4 - Security

- **WS Tier** – emerging standards (WS-Security)
- **Data source security** – security metadata containing rules for each user (“Distributors can read and update customers from their country.”)
- **Future: XACML** – allows other systems to inherit security rules from the original system (data source)

www.PortSight.com

Issue 5 – Namespaces and Structure Changes

- There may be tables or attributes with the same name – each table takes its namespace to sent XML data:

```
<Siebel.Customers> X <SAP.Customers>
```

- Concordance engine – matching records:

```
Siebel.CustName = 'John Smith'
X   SAP.CustomerName = 'J. Smith'
```

- The data structure may change -> use versioning:

```
<CRM.1.Customers>
```

  - It can be used for backward compatibility

www.PortSight.com

Issue 6 – Updating/Deleting Data

- XQuery is not going to support updates in the first version ☹
- You can send all changed data/data to be deleted back to the server and it will map them to appropriate data sources.
- Microsoft XML Updategrams – proprietary ☹

Issue 7 – Process Oriented Services (1)

- Placing order may not be only inserting a new record (data tier)
- But:
  - You have metadata -> you can generate specific classes for the data source
  - Developer can then inherit and implement special functionality (call CRM system API etc.)

Issue 7 - Process Oriented Services (2)

Booking a ticket – you may require to call existing WS (WS tier)

- **Option 1:** You can convert and automatically redirect the data to appropriate WS
- **Option 2:** You add related WS to your metadata so that user knows about them

Final Universal Data WS Interface

- **SELECT**(ProprietaryQuery) as XML
- **XQUERY**(XQuery) as XML
- **UPDATE**(UpdatedData)
- **INSERT**(InsertedData)
- **DELETE**(DeletedData)
- **SEARCH**(ProprietaryQuery,SearchExpression) as XML
- **GETMETADATA**(ProprietaryQuery) as XML containing XML Schema, relationships and optionally XACML descriptors for required data

www.PortSight.com

Further Development

- You can extend the system with:
  - Selecting presorted data (optimization)
  - Caching capabilities
  - WSRP/WSIA interface
  - Provisioning/Auditing system
  - Components for UI – easy implementation to your client applications
  - XSL-based system for viewing data and converting them to other formats (like HTML, standardized XML medical data, etc.).

Recommended and used sources:

- www.portsight.com/technology
- www.bea.com -> BEA Liquid Data
- www.ibm.com -> XPeranto, DB2, Information Integration
- www.oasis-open.org -> WS-Security, XACML, WSRP, WSIA
- www.nimble.com -> Nimble Integration Suite
- tiberi.us - great intro to XQuery
- xqueryservices.com – XQuery .NET library

www.PortSight.com

Questions & Answers

Thank You For Your Time!