



**WELCOME TO AMSTERDAM**



**Guidelines on how to XML-Enable  
Legacy Systems for Real Time  
Business-to-Business Integration**

**Edward Budgen**

Monday, March 26

8:30 a.m. to 12:00 p.m.

[www.xmlworld.org](http://www.xmlworld.org)

# SilverStream

Enabling Powerful Business Applications for the Web

## Using XML to Enable Legacy Systems for eBusiness Integration (eBI)

Eddie Budgen

[ebudgen@silverstream.com](mailto:ebudgen@silverstream.com)

+32 478 958754

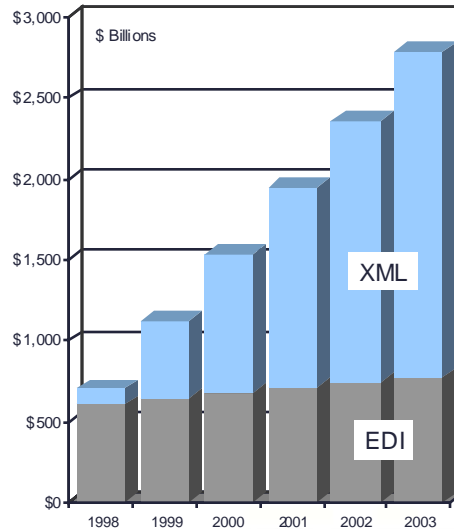
## Session Topics

- Background
  - eBusiness Integration is not new
  - Value of existing business systems
- Architecting eBI for XML
  - eBusiness Integration Platform
  - Layered XML interfaces & Mapping
  - Enabling Legacy Systems
- Example Implementation
  - XML Enabling IBM Hosts systems
  - Demonstration

## The B2B Market

- 24% of all B2B purchasing will be online by 2003 (up from 7% today)

Source: The Boston Consulting Group (BCG)



SilverStream®

## Market Landscape

- eBusiness is "Your Business"
  - About relevance and crossing boundaries
  - Big corporation phenomenon
  - Still very early stage
- Market shifting
  - Internet 1.0 to 2.0
  - Getting serious
- Standards matter
  - XML, J2EE
- Categories converging
  - Integration is critical
  - Patchwork to Platform



SilverStream®

## Internet "2.0"

### Internet 1.0

### Internet 2.0

Web sites



eBusiness

Simple marketing



Sophisticated  
1-to-1 relationships

Simple interactivity



Access to all  
business transactions

Retail eCommerce



All forms of commerce

## eBusiness Market Transition

### EDI

### eBusiness Integration

Stove-piped trading  
communities



Diverse trading  
opportunities

Fixed exchange  
formats



Hybrid exchange  
formats

Vendor managed



Public and Private  
management

Batch oriented



Straight-thru processing

Long time-to-market



Compressed time-to-market  
requirements

## eBusiness Integration

- eBI facilitates business applications to interact with each other across the internet
  - Both inside (EAI) and outside (B2B) the firewall
  - e.g. Ariba (cXML) transacting with SAP (OAG), an EDI (ACCORD) insurance claim transacting with a home grown Claims system, etc
- XML offers an open standards way to define application interfaces that expose business transactions to other requests across the internet
  - Web enablement of host based transactions
  - EDI Automation
  - Straight Through Processing (STP)



SilverStream®

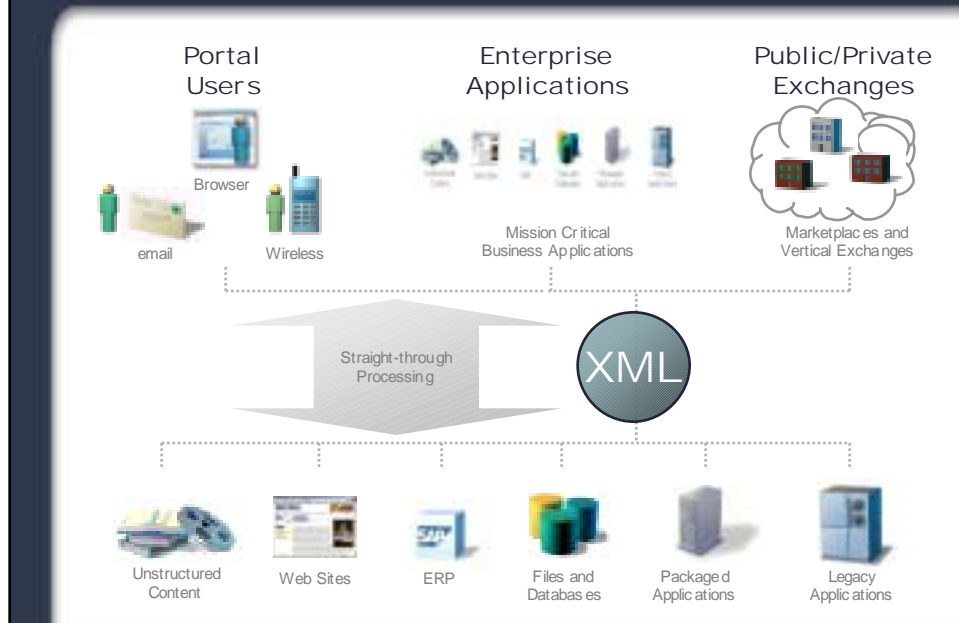
## eBusiness Integration

- Repurpose existing business applications
  - No need to build all new
  - Leverage existing business value and repurpose for internet based integration
  - Need to leverage existing platforms and communications infrastructure
- Smoother integration
  - Reduced cost and effort to integrate applications on disparate platforms e.g. Host, AS/400, Unix, Java, etc
  - Consistent XML interface for 'downstream' processing, web based or not – XML API

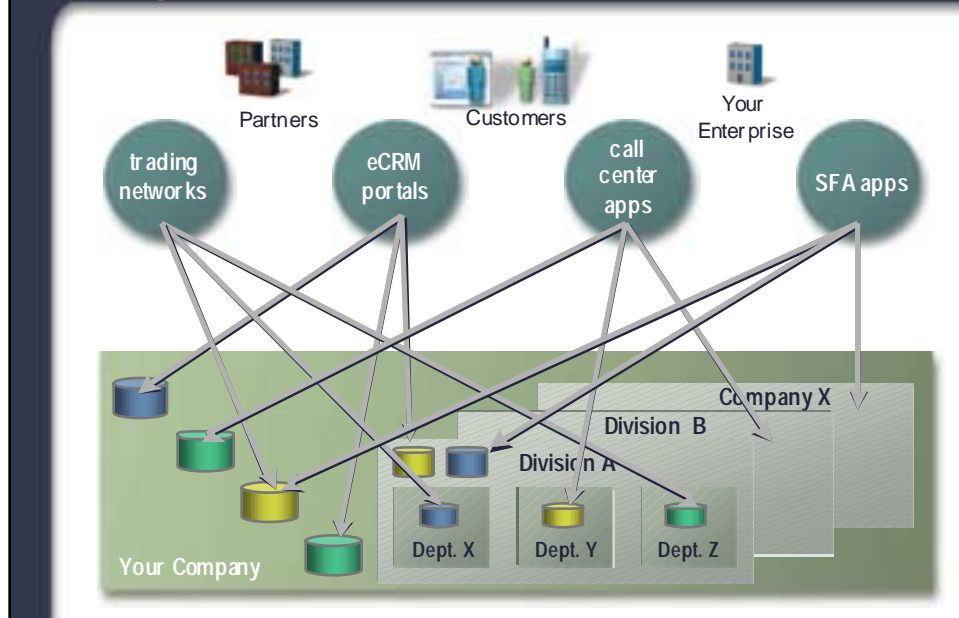


SilverStream®

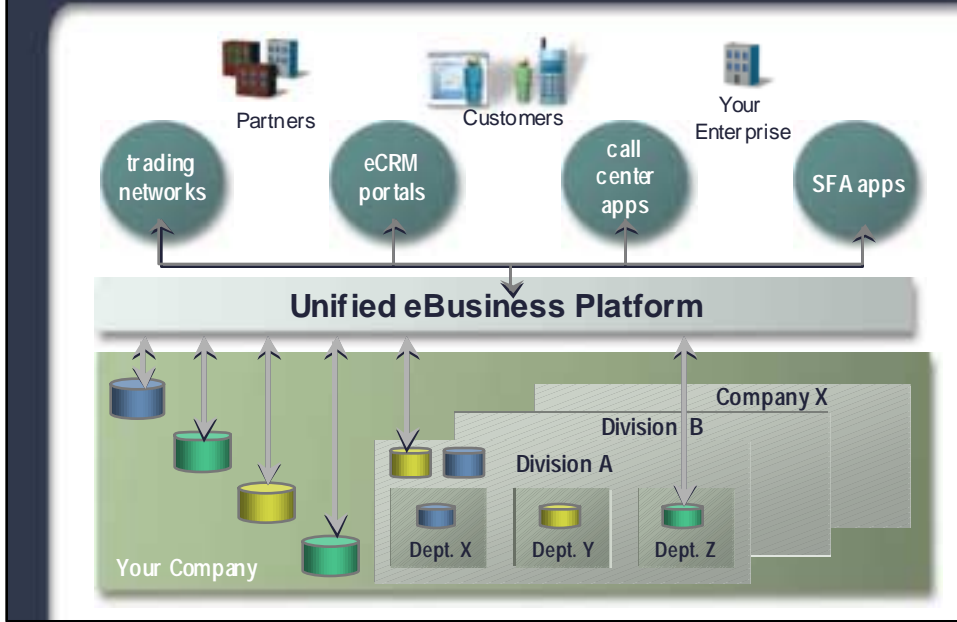
## eBusiness Integration Audiences



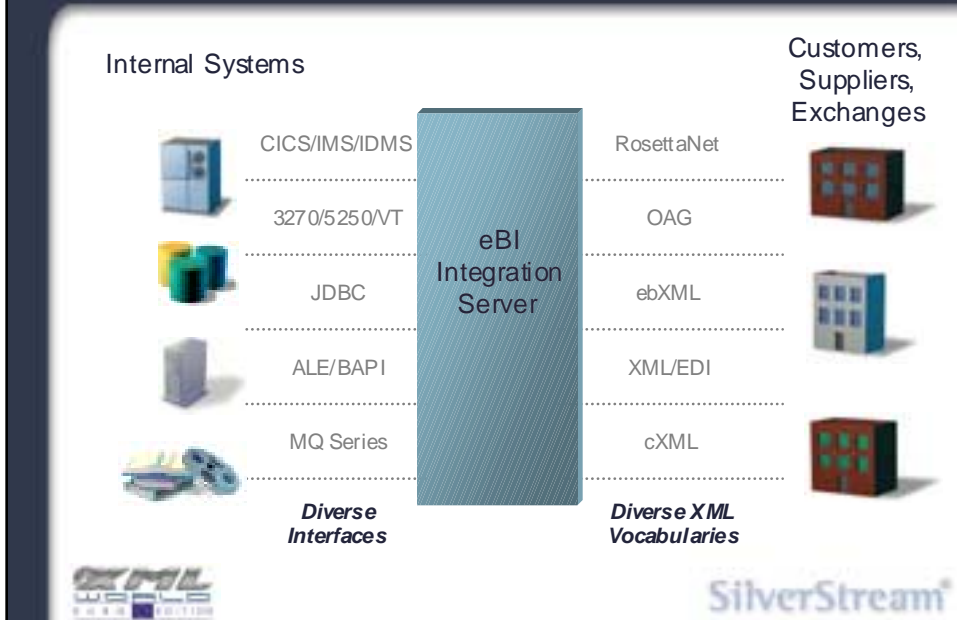
## eBusiness Applications: Integrated Information and Services



# eBusiness Integration



# eBusiness Integration Using XML



## eBI Solution Requirements

- Time to Market
  - Need software tools and development methods that offer
    - Rapid and easy development – including business users
    - Comprehensive connectivity – with existing infrastructure
    - Robust, scalable and secure runtime on enterprise platforms
- Straight Through Processing (STP)
  - Brings both information and transactions to a consistent XML interface across the internet
  - Reaches into back-end business transaction systems
- Standards based architecture
  - J2EE – portability of business applications and components
  - XML – transparency of integration across the internet



SilverStream®

SilverStream®

**SilverStream**

Architecting eBusiness Integration for XML





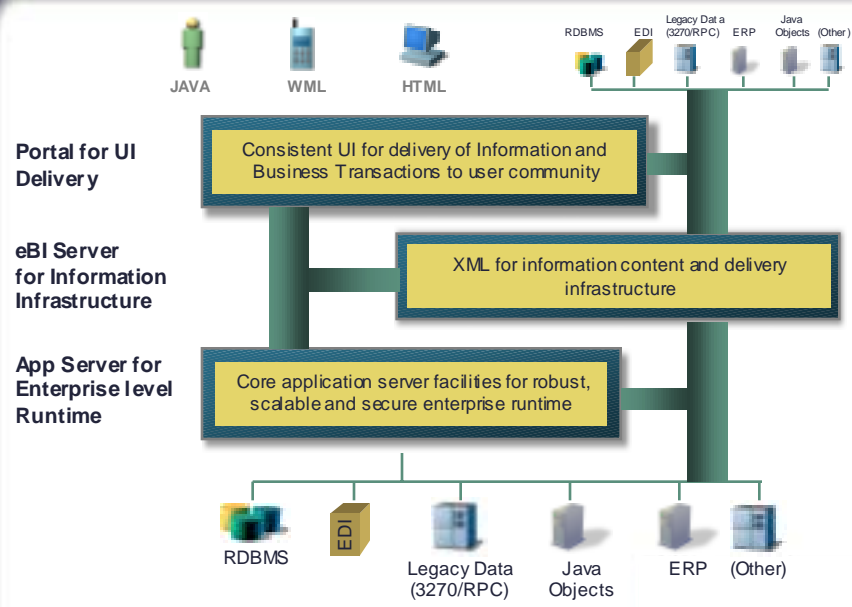
## XML – The Standard

- W3C driving XML and related standards
  - XML, XSL, XSLT, XPATH, XLINK, XPOINTER
- Many Industry consortia defining XML standard exchange documents/processes
  - RosettaNet – Electronic components and Semi-conductor
  - ebXML – eBusiness document and process standards
  - WISE & Accord – Insurance
- ..... Many others
- Software standards development
  - WSDL – Web Services
  - SOAP – XML RPC mechanism
  - UDDI – Services directory
- Core use within J2EE

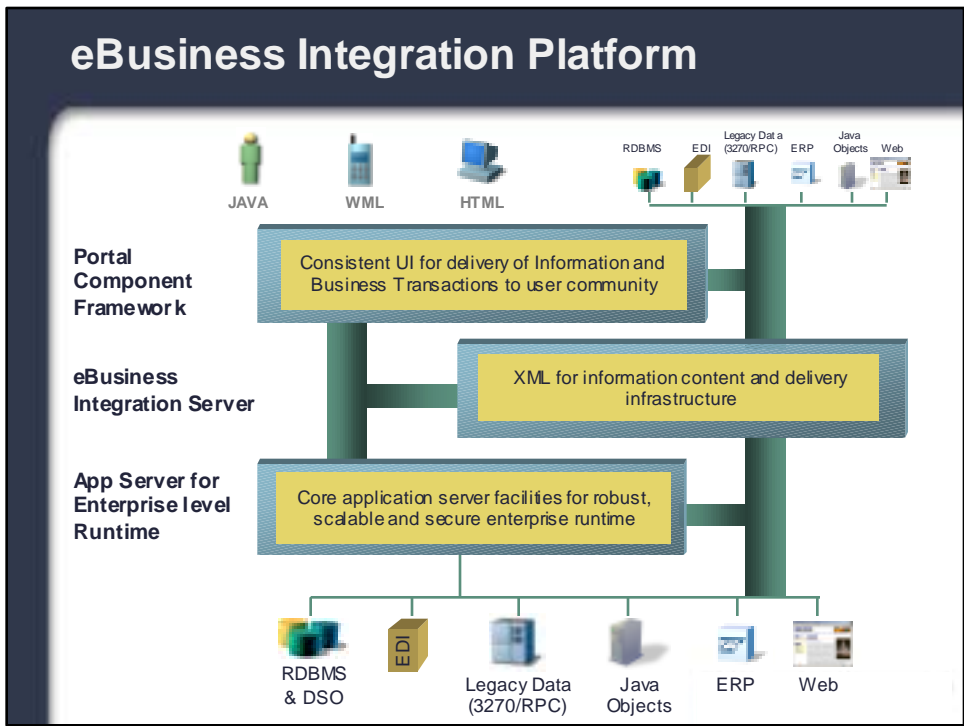


SilverStream®

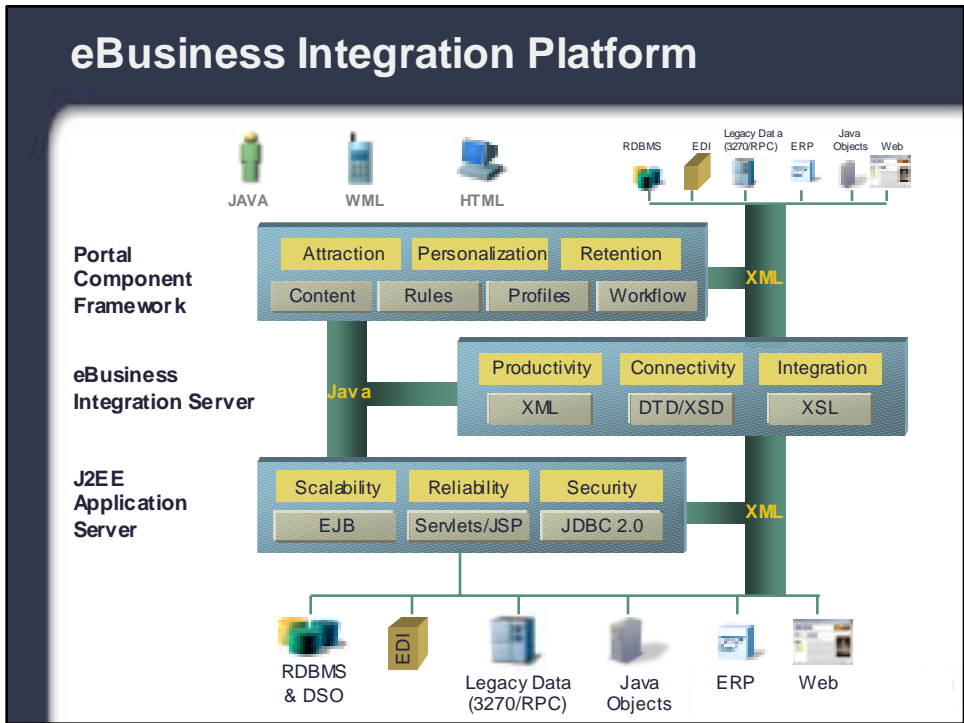
## eBI Platform Architecture



# eBusiness Integration Platform



# eBusiness Integration Platform



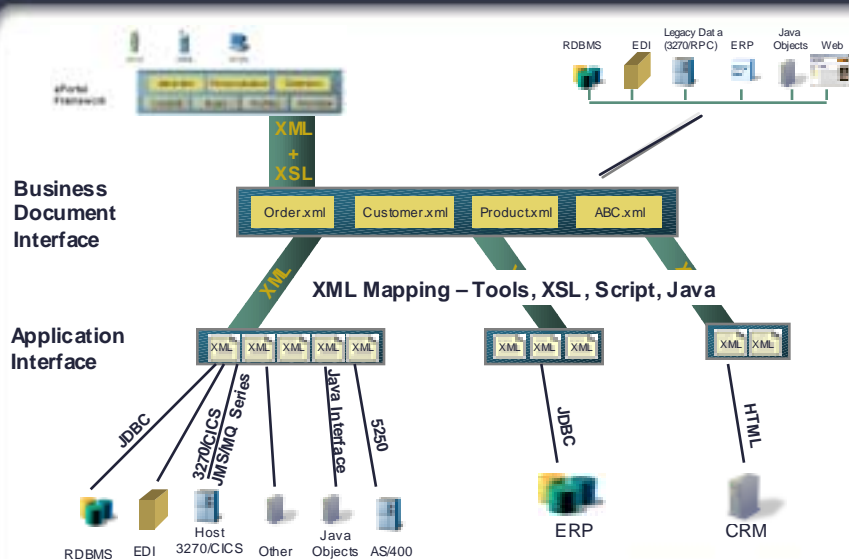
## Straight Through Processing (STP)

- Bring transactions to the Portal
  - Provide All business services to your user via a consistent, relevant and personalized interface
  - Combine two stages into one solution
    - 1) Info delivery & 2) Business services/transactions
- Layered XML Interfaces
  - Business Document Interface – business owner
  - Application Interface – application owner
  - XML Mapping layer – choice of architecture and technology
- Detach eBusiness evolution from back-end
  - Web services exposed as public/private interfaces
  - Native connection to leverage existing business applications



SilverStream®

## Straight Through Processing



## STP Architecture

- Separate business and application interfaces using XML
  - Can apply most appropriate resource to tasks
  - Isolates business uses from application specific issues (platform, development)
- Mapping from XML business interface to application represents “meta layer”
  - Isolates upstream business processing from application processing changes – future proofing
  - XML offers understandable mapping
- Leverages existing platform infrastructures
  - Can evolve infrastructure at different pace from new web developments



SilverStream®

## XML Information Infrastructure

- Business Document Interface
  - XML schema/documents defined by users to represent business documents and processes
  - Public standards available (OAG, RosettaNet, cXML, etc.) for inter-Enterprise exchange
  - Typically needs business specific extensions within the firewall
- Application Interface
  - XML schema/documents more closely aligned with application information structures than business documents
  - Requires native interface to application platforms
- Mapping
  - Need efficient mapping from application to business XML schema – meta layer: Tools, XSL, Scripting, Java



SilverStream®

## eBI Server: Requirement #1

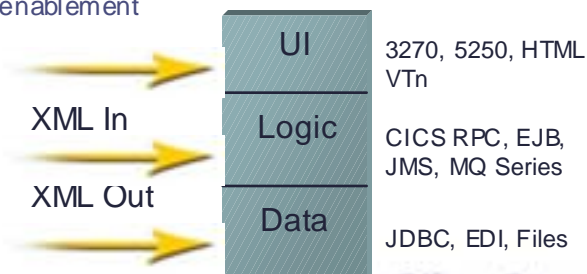
- High-productivity designer interfaces for connecting XML-enabled applications
  - Systems analysts need visual controls
    - The GUI stuff
  - App programmers need scripting
    - Jscript, JavaScript – really ECMA
  - Software engineers need full Java
    - Integration with existing Java Business Objects
    - Can use Java APIs to extend functionality
    - Extensible Expression Builder to return value to the end user



SilverStream®

## eBI Server: Requirement #2

- High-productivity designer interfaces for connecting XML-enabled applications
- Tools for XML-enabling a wide variety of enterprise applications
  - User Interface based enablement
  - Logic based enablement
  - Data based enablement



SilverStream®

## eBI Server: Requirement #3

- High-productivity designer interfaces for connecting XML-enabled applications
- Tools for XML-enabling a wide variety of enterprise applications
- Enterprise deployment platform
  - Component wrappers (Servlet, JMS, EJB etc.)
  - Recoverable, scalable, secure
  - Responsive and extendable

Deploy to J2EE Application Servers  
WebSphere, SilverStream, WebLogic, .....



SilverStream®

## Public & Private Services

- Public Services
  - Web Service:
    - Synchronous service base depends on business and transaction model
    - XML Standards arriving – WSDL, UDDI, SOAP
  - Message Service
    - Asynchronous or message base – loosely coupled
    - Should implement same XML standards
- Considerations
  - Isolation from internal change
  - Easier to establish electronic relationships
  - Includes business transactions
  - Not complete yet!



SilverStream®

## Public & Private Services

- Private Services
  - Internal proprietary service model
  - Can implement any model
- Considerations
  - Proprietary means more effort but perhaps better fit for your organization?
  - Better performance?
  - Needs specific effort for all relationship additions and changes



SilverStream®

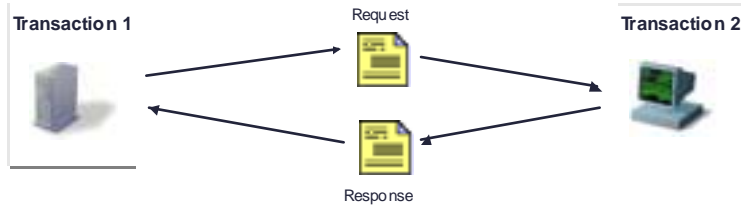
## Public & Private Services

- Can implement mixed model and migrate in future
  - Begin implementation of Web Services model for selected new business services
  - Wrap/integrate internal model for XML request/response
    - May require development of bridges
    - Requires intelligent routing of requests/responses
    - Less effort than rewriting as leverages existing business systems
  - Can migrate to public services as technology, software and experience dictates – XML acts as isolation layer inside as well as externally



SilverStream®

## Common Integration Design



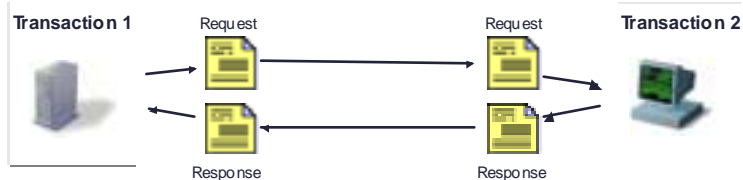
Why is this vulnerable?

If either business or transaction interface (database schema or terminal screen) changes, all integration services that have been hard-baked to that interface must be modified.



SilverStream®

## Good Integration Design



Indirection provides an important isolation layer between app interfaces in exchange for an acceptable amount of additional effort

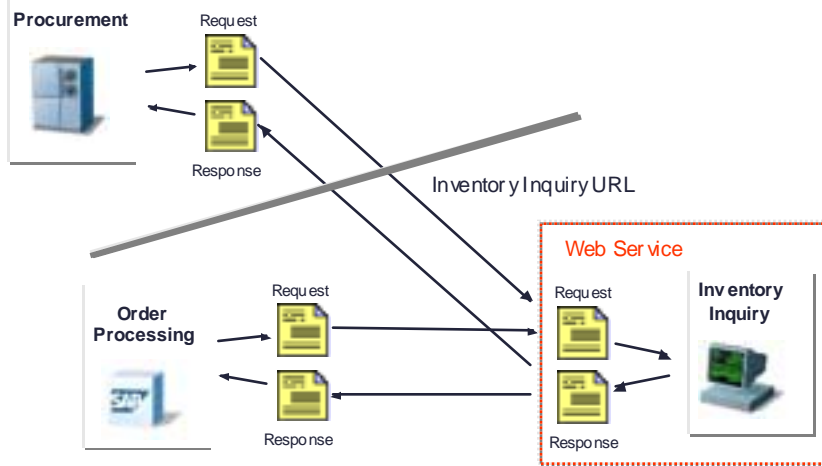
Both applications are protected from change in the other. XML is used for the request/response pair (e.g., SOAP)



SilverStream®



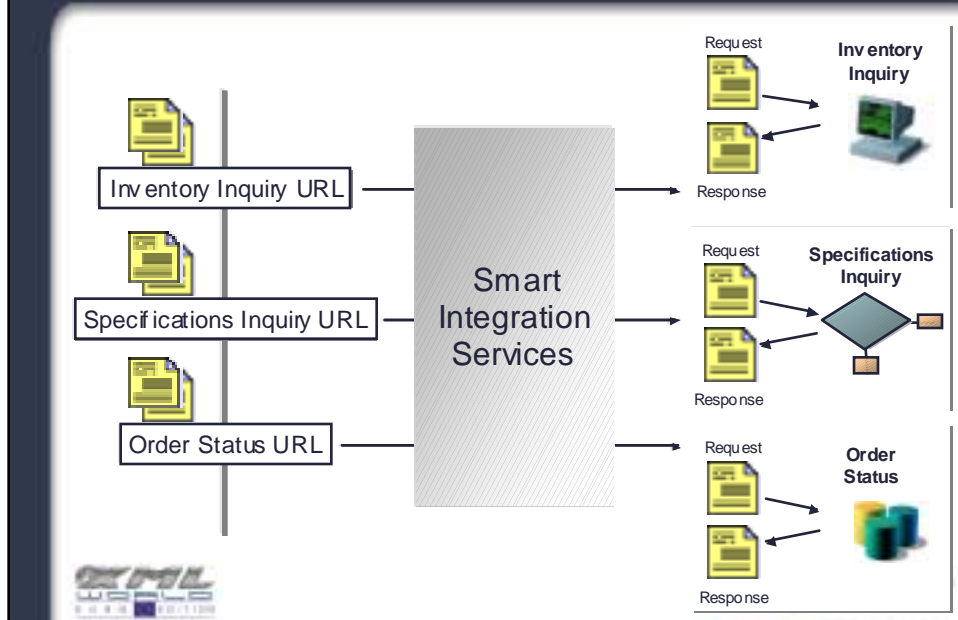
## Web Service Example



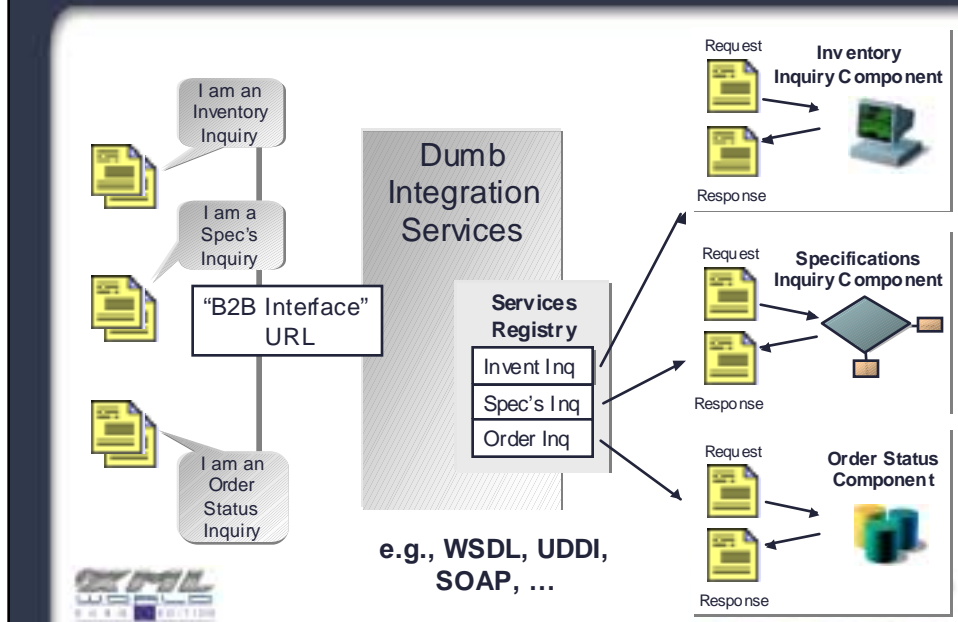
## Web Services Summary

- Select (or create) appropriate XML Request/Response documents for each new application resource:
  - Database tables
  - Terminal screens
  - COBOL copybooks, ....
- Map the defined Request/Response documents to their underlying app resources
- Integrate with other XML business standards through XML-XML mapping – Wrap for SOAP

## No Standards



## Standards



# Sample SOAP Message

## Sample SOAP Message Embedded in HTTP Request

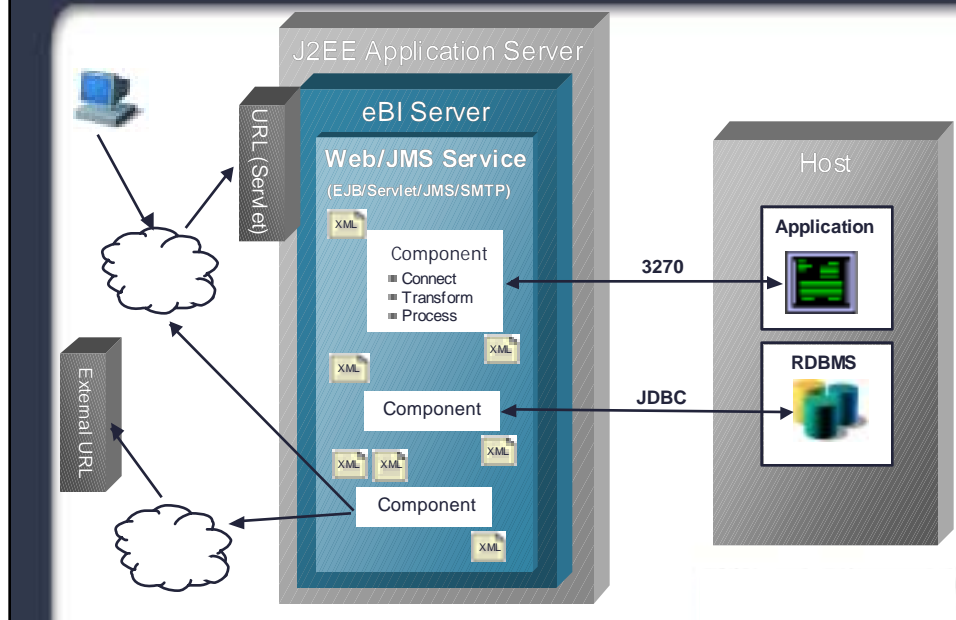
```
POST /YourCompanyGeneralInquiry HTTP/1.1
Host: www.yourco.inquiry.com
Content-Type: text/xml;
charset="utf-8"
Content-Length: nnn
SOAPAction: "Some-URI"
```

```
<SOAP-ENV:Envelope xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/"
SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <SOAP-ENV:Body>
    <m:ProductInquiry xmlns:m="Some-URI">
      <SKU>LOR8437</SKU>
      </m:GetLastTradePrice>
    </SOAP-ENV:Body>
  </SOAP-ENV:Envelope>
```

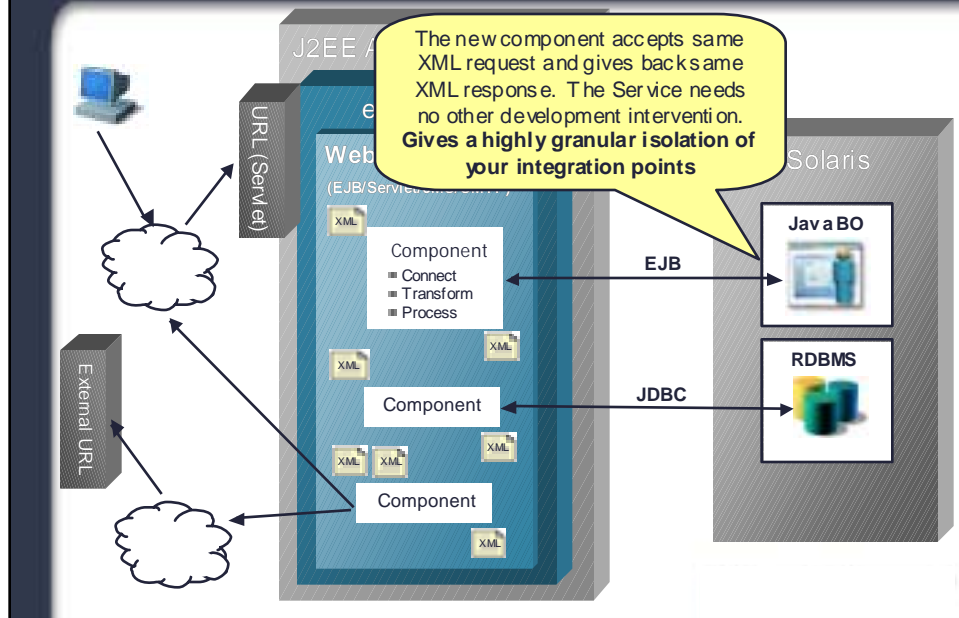
Some internal service call



# eBI Component Architecture (Example)



## Component Architecture (Example)



SilverStream®

## SilverStream

Example – XML Enabling IBM Host Systems

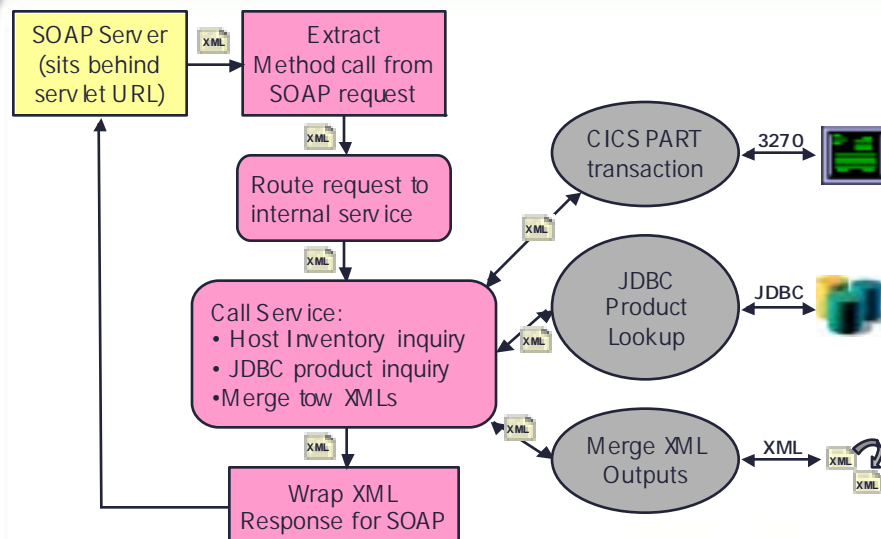
## Example - SOAP based Inventory Inquiry

- Create “Master” SOAP receiver
- Extract SOAP service request
- Execute Web Services against host (CICS via 3270) and RDBMS
- Collect responses
- Wrap in SOAP envelope and respond



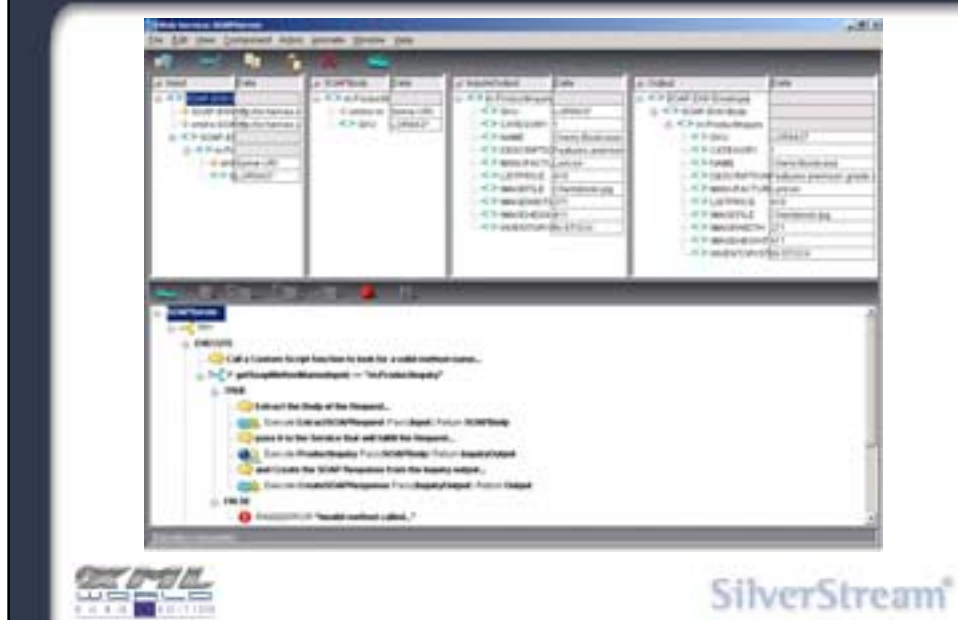
SilverStream®

## Component Diagram

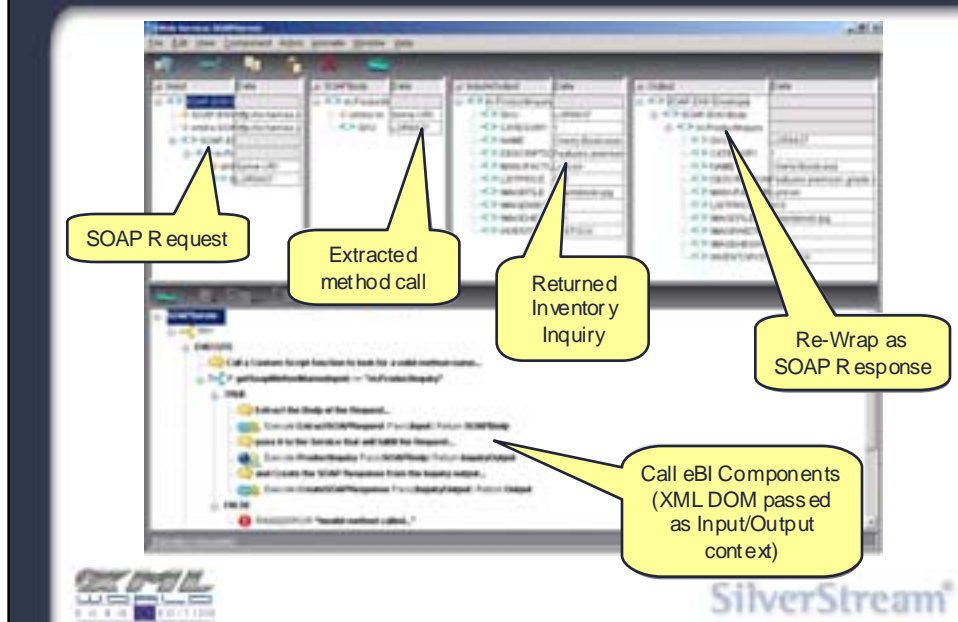


SilverStream®

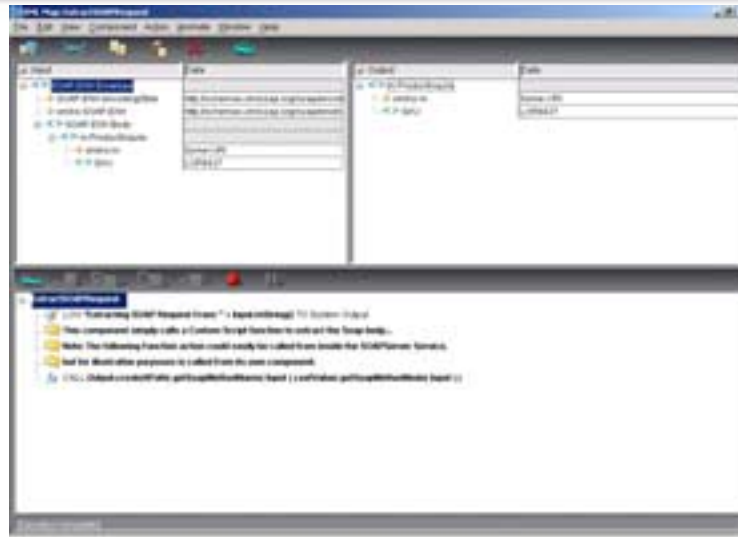
## SOAPServer "Master" Service



## SOAPServer "Master" Service

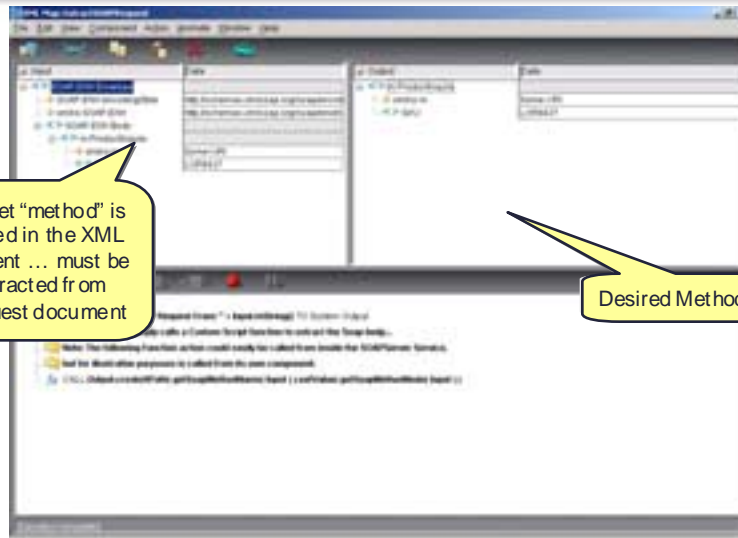


## ExtractSOAPRequest



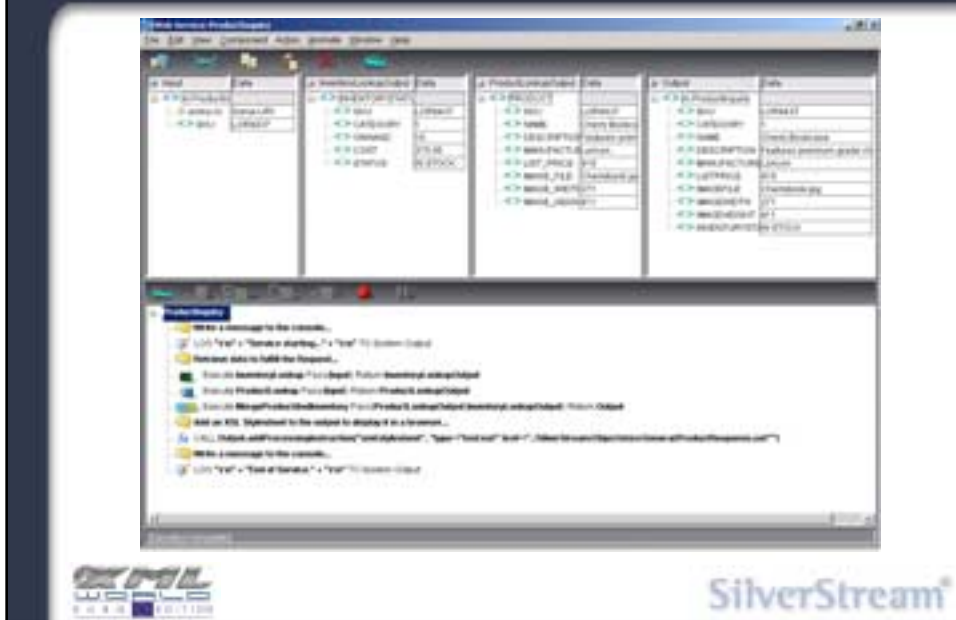
SilverStream®

## ExtractSOAPRequest

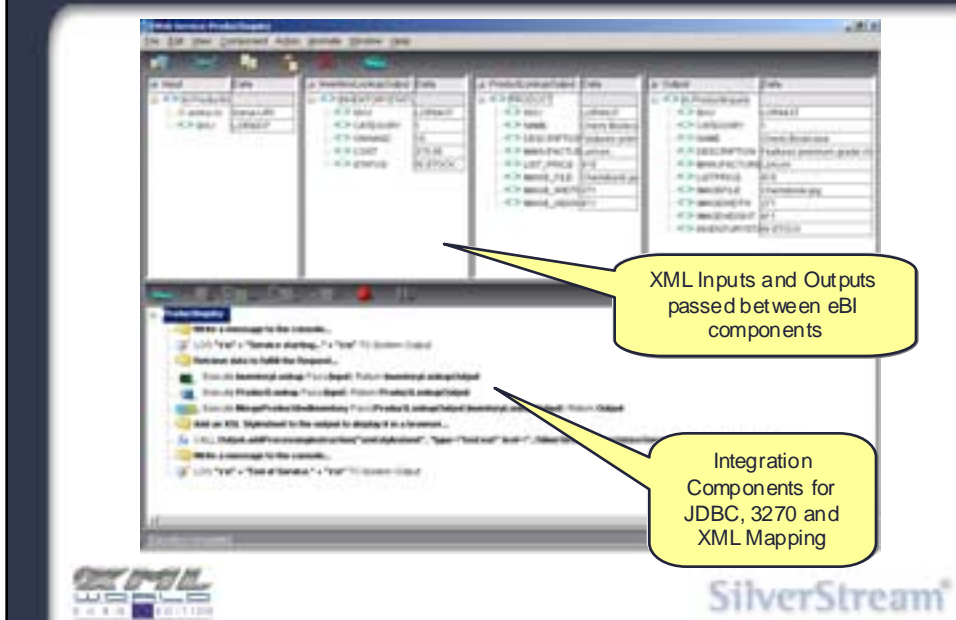


SilverStream®

## ProductInquiry - Service



## ProductInquiry - Service





## ProductLookup – JDBC Component

The screenshot shows the configuration interface for the ProductLookup component. A yellow callout bubble points to the 'SQL Statement' field, containing the following SQL code:

```
SELECT * FROM PRODUCT WHERE PRODUCT_ID = ?
```

Other fields visible include 'Result Mapping' and 'Result Type'. The SilverStream logo is at the bottom right.

SQL Statement integrating XML's search tag (Passed in with SOAP Request)

## InventoryLookup – 3270 (CICS)

The screenshot shows the configuration interface for the InventoryLookup component. Two grey arrows indicate XML data flow: 'XML In' pointing to the 'XML In' field and 'XML Out' pointing to the 'XML Out' field. A yellow callout bubble points to the 'SQL Statement' field, containing the following SQL code:

```
SELECT * FROM PRODUCT WHERE PRODUCT_ID = ?
```

The SilverStream logo is at the bottom right.

XML interaction with 3270 online transaction (CICS) – XML In and XML Out

## MergeProductAndInventory – XML Map

The screenshot shows the XML Map interface for 'MergeProductAndInventory'. It features three data tables: 'Input', 'Map', and 'Output'. The 'Input' table contains product details like SKU, Category, Name, Description, Manufacturer, List Price, Image File, Image Width, Image Height, and Inventory Status. The 'Map' table shows the mapping of these input fields to the 'Output' table. The 'Output' table contains the resulting XML structure. Below the tables is a code editor with the following XML code:

```

<?xml version="1.0" encoding="UTF-8" ?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <MergeProductAndInventory>
      <Product>
        <SKU>1411</SKU>
        <Category>General</Category>
        <Name>Cherry Brandy</Name>
        <Description>Features premium grade cherry</Description>
        <Manufacturer>London</Manufacturer>
        <ListPrice>415</ListPrice>
        <ImageFile>CherryBrandy.jpg</ImageFile>
        <ImageWidth>411</ImageWidth>
        <ImageHeight>211</ImageHeight>
        <InventoryStatus>Out of Stock (in order)</InventoryStatus>
      </Product>
    </MergeProductAndInventory>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

A yellow callout bubble points to the code editor with the text: "Merge XML from Product and Inventory components".

**SilverStream**

## CreateSOAPResponse – XML Map

The screenshot shows the XML Map interface for 'CreateSOAPResponse'. It features two data tables: 'Input' and 'Output'. The 'Input' table contains product details like SKU, Category, Name, Description, Manufacturer, List Price, Image File, Image Width, Image Height, and Inventory Status. The 'Output' table shows the resulting XML structure. Below the tables is a code editor with the following XML code:

```

<?xml version="1.0" encoding="UTF-8" ?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <MergeProductAndInventory>
      <Product>
        <SKU>1411</SKU>
        <Category>General</Category>
        <Name>Cherry Brandy</Name>
        <Description>Features premium grade c</Description>
        <Manufacturer>London</Manufacturer>
        <ListPrice>415</ListPrice>
        <ImageFile>CherryBrandy.jpg</ImageFile>
        <ImageWidth>411</ImageWidth>
        <ImageHeight>211</ImageHeight>
        <InventoryStatus>Out of Stock (in order)</InventoryStatus>
      </Product>
    </MergeProductAndInventory>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

A yellow callout bubble points to the code editor with the text: "Re-wrap XML Output for SOAP Response".

**SilverStream**

## Business Systems Connections

- Mainframe:
  - 3270, CICS (ECI), MQ Series, IMS (OTMA), DB2, ....
- Midrange
  - AS/400 – 5250, Data Queue, CICS, DB2, ....
  - Unix/Vax – VTn, RPC, JMS, ....
- New Age
  - Java Beans, EJB, XML, ....
- Data:
  - EDI (EDIFACT/X12), RDBMS, Flat File, ...
- Online:
  - HTML, XML
- Packaged Applications
  - SAP, Siebel, Peoplesoft, Ariba, ....



SilverStream®

## Standards Development

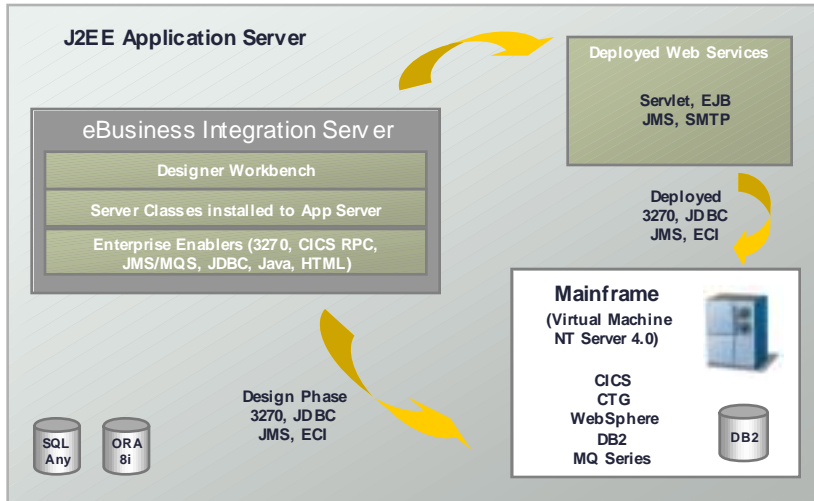
- Business standards
  - Mainstream adoption
    - ebXML (Electronic Business), Rosettanet (Supply chain management), tpaML (Third party agreements), ....
- Web Services
  - Formalization and adoption of XML based standards – WSDL, UDDI, SOAP, ....
  - Corollary standards
    - S2ML (Security), WfMC (Workflow), LDAP, ....
- J2EE evolution
  - Java Connector Architecture (JCA) – standards based connectivity (Java & XML)
  - Java Transaction Service (JTS), Java Transaction Architecture (JTA) – Distributed transactions



SilverStream®

# Demonstration Environment

## Windows 2000 Professional



SilverStream®

## SilverStream Summary

## Keys to success

- **Adopt Standards**
  - J2EE & XML as core technologies
  - Industry standard vocabularies
- **Implement layered architecture with XML**
  - Business Documents Interface
  - Application Interface
- **Architect for long term and build for immediate return**
  - Architect it right – standards based platform
  - Pilot today for reduced time to market and fast business benefits
  - Partner with experience
- **Leverage existing business value**
  - Repurpose existing business systems, don't wait to rebuild/buy



SilverStream®

## Summary

- **Legacy integration critical to eBusiness initiatives**
  - Can't wait to rebuild
  - Must leverage existing business systems
- **XML has matured**
  - Standards adoption rate increasing (ebXML, Rosettanet, etc.)
  - Application and process integration in Web Services (WSDL, UDDI, SOAP, etc.)
- **XML for Straight Through Processing**
  - Bring the transaction to the internet – eBusiness Integration
  - Layered XML architecture isolates eBusiness from back end systems
- **Can begin implementation today**
  - Standards based products exist
  - Experienced solution providers in the market



SilverStream®