Leveraging Web Services
Application Integration

David S. Linthicum
CTO Mercator
Application Integration Types

- Portal -Oriented
- Process Integration -Oriented
- Application Service -Oriented
- Information -Oriented
Information-Oriented Application Integration

User Interface → Logic → Data → Transformation Formatting → Data → Logic → User Interface

B2B

mercator
Intelligent Business Integration
Application Service-Oriented Application Integration

- Method integration to create a composite application
- Sharing business logic
- Sharing code
- Sharing processing
  - Shared programs
  - Shared transactions
  - Shared objects
Value Of Application Service-Oriented Application Integration

► Provides true code reuse infrastructure for many enterprise applications

► Availability of technology and expertise

► Ultimate application integration solution for many trading communities and enterprises
Issues With Application Service-Oriented Application Integration

- Much more complex and expensive than the other approaches
- Takes a lot of time, architecture, and planning
- Enabling technology may not scale to enterprise class applications or fall short in other ways
- Too invasive for most B2B problem domains
Enabling Technology

- Application servers
- TP monitors
- Distributed objects (CORBA/COM)
- Traditional development tools
- Web Services
Application Integration is Evolving

Loosely coupled, independent

- Autonomous Monolithic Applications
- Data Consistency
- Multi-step process

Tightly coupled, interdependent

- Composite Applications (Web Services)
- Autonomous Distributed Applications

Business Process Control

mercator
Intelligent Business Integration
Multistep Process Dominates For Now

1. Order supplies
2. Create Order
3. Ship Order
4. Update Order Status
5. Update Order Status

Order Processing System
Web Site
Logistics and Shipping System
Billing System

Customer
Payment
Bill
Product
Process Integration Will Define Application Integration

Process Model

System A

System B

Company C
Composite Application

- Checking System
- Savings System
- Mortgage System
- Equities System

Integration Logic

Single View of the Customer Web Site

Customer checking on account
## Contrasting Three Forms of Integration

### Data Consistency
- Loosely coupled integration
- Multiple applications
- Asynchronous dominates
- One-way
- Simple abstraction of business rules

### Multi-step Process
- Loosely coupled integration
- Multiple applications
- Can be either Asynchronous or Synchronous based
- One-way and request/reply
- Limited complexity in abstraction of business rules

### Composite Application
- Tightly coupled integration
- Single application
- Synchronous dominates
- Request/reply
- Complex abstraction of business rules
Enterprise Architecture Framework Layers

- Application-Specific Components
- Business Domain Components
- Extended Platform Components
- Platform/Network Components
Platform/Network Components

- Networks
  - Routers
  - Switches
  - Communication Links
  - Firewalls (assumes functionality limited)

- Processors

- Operating Systems
Extended Platform

Information Systems Technology Services
- Object Request Brokers (ORBs)
- Database Management Systems
- Web browsers, servers
- Transaction Monitors
- Legacy system bridges
- Directory services
- Application Servers
- Integration servers
- System management
- Custom-written business-independent services
- **Shared business services and information**
  - Including general-purpose services
    - General ledger
    - Office automation
  - Shared stateful run-time services
  - Reusable code components
- **Key question: business functionality vs. technical support**
Dedicated to one application

- GUI
- Server-side code
- Data (not shared)
Components Are Designed for Reuse Across Applications

- Home Banking
- Call Center
- Branch

- Application Specific Components

- Business Domain Components

- Extended Platform Components

- Platform/Network Components

- UNIX
- NT
- MVS
Go to a Branch Office to Change Address

Application Specific Components

Business Domain Components

Extended Platform Components

Platform/Network Components

Home Banking | Call Center | Branch

Customer

Change Address

Data Access to Checking Acct | Data Access to Savings Acct | Distributed Services

UNIX | NT | MVS

mercator
Intelligent Business Integration
Technology Considerations

- Common user interface
- Common programming model
- Common security model
- **Synchronous integration**
  - DOT - CORBA, DCOM, Java, etc.
- **Asynchronous integration**
  - Message Queuing
  - Publish & Subscribe
- **Transactional integration**
  - CICS ➔ TM & OTM
- Work flow/flow control
- Web application services
- Data transformation/translation
- Component model (containers or composite objects)
- Connectors (adapters)
- Integration with systems management frameworks
## Integration Tools

<table>
<thead>
<tr>
<th>Application Servers</th>
<th>Target delivery of legacy information to the web, with the addition of new logic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Message Brokers / MOM</td>
<td>Target legacy to legacy integration and transform information between applications</td>
</tr>
<tr>
<td>Object Request Brokers / Object Transaction Monitors</td>
<td>Mechanisms that find the object and prepare the implementation for the request, then communicate the data</td>
</tr>
</tbody>
</table>
Progression Of Application Integration Solutions (EAI and B2B)

Application Integration Implementation Order

Value

- Transport
- Message Brokering and Translation
- Process Integration
- Web Services
Integration Brokers Invoked As Web Services

Deployment Time

- UDDI
- WSDL
- ebXML
- Map
- WS Map
- SOAP Adapter
- Type Tree
- Application
- Database
- File
Web Services Invoked by Integration Brokers

Design Time

Application

Database

File

Map

WS Map

SOAP Adapter

Type Tree

UDDI

WSDL

ebXML

mercator
Intelligent Business Integration
Web Services Invoked by Integration Brokers

Run Time

Application

Database

File

Map

Map

SOAP Adapter

Type Tree

Web Service

Application

Database

File

mercator

Intelligent Business Integration
Thank You!