ESB - An Introduction to an Enterprise Workflow Framework

Tom Bender
Tendril Networks

tommybender@gmail.com
Introduction

• Adoption of SOA Architecture?
  • Commercial, Open Source, Rolled your own?

• Adoption of ESB Solution?
  • Commercial, Open Source, Rolled your own?

• What are your goals in attending this session?
  • What, Where and Why of ESB?
  • Strengths and Weaknesses?
  • Workflow or ESB code samples?
Introduction

Tom Bender
Introduction
Introduction
Introduction
Introduction
Introduction
Introduction
Introduction
Introduction
Introduction

Systems Architect
Introduction

Industrial Machine Vision

Ann Arbor, Michigan
Introduction

Quantitative Expert Trading Systems
Los Angeles, California
Introduction

Constraint-based Expert Systems & Sales Force Automation
Golden, Colorado
Introduction

Consultant
Boulder, CO
Introduction

Local Commerce

Jabber IM

Denver, CO
Introduction

Satellite Scheduling & Geospatial Information Systems

Longmont, CO
Introduction

Wireless Sensor Networks

802.15.4

Boulder, CO
Introduction

• Dave Chappelle?
Technology Quote

"Things should be made as simple as possible, but no simpler."
- Albert Einstein
Agenda

- Problem Scope
- Definition of Terms
- Principles of a Service Oriented Architecture
- Properties of an Enterprise Service Bus
- Evolution and Architecture of the Enterprise Service Bus
- ESB Adoption: Partitioning and Encapsulating SOA
- ESB Related Standards: Impacts and Considerations
- Enterprise Service Bus Market: Open Source and Commercial
- Workflow Examples using ESB Technology
Problem Scope

- Integration is a business necessity
- Accidental architecture is pervasive
- Integration brokers are not sufficient
Problem Scope

Integration is a Business Necessity

- Increase in the need for integration
- Every business has integration needs
  - Data
    - Sharing and transformation of data
  - Process
    - Coordination of multiple systems
Problem Scope

Accidental architecture

➢ Stove pipe software is extremely prevalent
  • Highly specialized
  • Tight coupling
  • Stand alone
  • Disparate
  • No standards compliance
  • General lack of consistency
  • High human requirement is common
Problem Scope

Integration Brokers Are Insufficient

- Hub and spoke architecture
  - Does not support coordination across systems
  - Extremely high investment
  - Proprietary
    - Specialization for particular markets
    - No two vendors are compatible
    - Requires retraining and/or new hiring
  - Highly centralized
  - Monolithic
Problem Scope

Custom Implementations Are Insufficient

➢ Dependent upon

• Budgets
• Knowledge base
• Domain of focus
• Time: Opportunity Cost
• Build or Buy?
### Integration Approaches

<table>
<thead>
<tr>
<th>Application and Integration logic separated</th>
<th>Application and Integration logic not separated</th>
</tr>
</thead>
<tbody>
<tr>
<td>Traditional EAI</td>
<td>Application Servers</td>
</tr>
<tr>
<td>ESB</td>
<td>MOM</td>
</tr>
</tbody>
</table>

**Hub-and-Spoke Integration**

**Distributed Integration**
Definition of Terms

- **Service:**
  - is a coarse-grained, discoverable, and self-contained software entity that interacts with applications and other services through a loosely coupled, often asynchronous, message-based communication model.

- **Service Oriented Architecture:**
  - a collection of services with well-defined interfaces and a shared communications model is called a service-oriented architecture (SOA). A system or application is designed and implemented as a set of interactions among these services.

- **Enterprise Service Bus:**
  - provides a light weight, loosely coupled, event-driven SOA with a highly distributed universe of named routing destinations across a multi-protocol message bus.
Definition of Terms

- **Workflow:**
  
  > A workflow management system (WFMS) is a software component that takes as input a formal description of business processes and maintains the state of processes executions, thereby delegating activities amongst people and applications. - Tom Baeyens, jBPM Founder
Definition of Terms

- Hype Cycle (Gartner)
Principles of SOA

“Four Tenets of SOA” as proposed by Don Box:

- Boundaries are Explicit
- Services are Autonomous
- Services share Schema and Contract, not Class
- Compatibility is based upon Policy
Properties of an ESB

- Light Weight
- Loosely Coupled
- Event-Driven
- Highly Distributed
- Selective Deployment
- Abstract Endpoints
- Intelligent Routing
- Data Transformation (inbound/outbound)
- Reliable Messaging
- Multi-Protocol Message Bus
Properties of an ESB

- Light Weight
  - Better, Faster, Lighter Java - Bruce Tate
  - Principles for Fighting Bloat
    - Simplicity
    - Cohesion
    - Transparency
    - Extensibility
    - Heresy
    - Question Technology
Properties of an ESB

- Loosely Coupled
  - Maximum independence between modules
    - Changes in one module will not unknowingly affect other modules.
  - Isolation and cohesion improve software re-use
    - Simple, determinant behavior improves reliability
  - Simplified Support and Maintenance
  - Interface lets you couple to a capability rather than an implementation
  - Event notifications can decouple sources and sinks in space, time, and topology, and provide a defined interface for the separation of concerns between different handlers.
Properties of an ESB

- Event Driven [Architecture]
  - Supports massive scale
    - Thread-based concurrency too heavy weight
      - Thread services a single request
    - Event-based concurrency
      - Thread services multiple requests
        - select/poll
        - asynchronous I/O
        - signals
        - completion ports
  - Dan Kegel - C10K Paper
    - [http://www.kegel.com/c10k.html](http://www.kegel.com/c10k.html)
Properties of an ESB

- **Event Driven [Architecture]**
  - Staged Event-Driven Architecture (“SEDA”)
    - Matt Welsh, Assistant Professor Harvard University
      - [http://www.eecs.harvard.edu/~mdw](http://www.eecs.harvard.edu/~mdw)
      - NBIO, SEDA
    - NBIO: Non-blocking IO for Java
      - JSR-051 – java.nio.* in J2SE 1.4.2
Properties of an ESB

- Highly Distributed
  - Services *may* be distributed across a network
    - Analyze the costs and benefits
      - Peter Deutsch – 8 Fallacies of Distributed Computing
  - Enterprise Application Integration
    - Driving Market Force for ESB
      - Heterogeneous Applications
      - Already distributed
  - Legacy System Integration
    - Already Distributed
      - Integration of *working* systems
Properties of an ESB

- The Eight Fallacies of Distributed Computing: Peter Deutsch
  - The network is reliable
  - Latency is zero
  - Bandwidth is infinite
  - The network is secure
  - Topology doesn't change
  - There is one administrator
  - Transport cost is zero
  - The network is homogeneous
Properties of an ESB

- **Highly Distributed**

  ➢ Supports Multiple Distributed Network Topologies
    - Centralized: Client-Server
    - Centralized: Hub-and-Spoke
      ✓ Database Servers, J2EE Containers and Web Servers
    - Decentralized P2P: Pure P2P Architectures, Mesh Networks
    - Decentralized: Super-Peer Architectures
      ✓ P2P Architectures
Properties of an ESB

- Supports Multiple Network Topologies

Figure 1: Centralized systems
Figure 2: P2P systems
Figure 4: Super peer architecture
Properties of an ESB

- Selective Deployment
  - Decoupled and independent services *may* support deployment of selected services.
Properties of an ESB

- Abstract Endpoints

![Diagram](image-url)
Properties of an ESB

- Abstract Endpoint & Service
Properties of an ESB

- Intelligent Routing
  - Itinerary-based Routing
    - Static itinerary travels with the message
  - Content-based Routing
    - Dynamic itinerary is determined at runtime based upon simple business logic
  - Context-based Routing or Orchestration
    - Dynamic itinerary is determined at runtime based upon complex business logic and application state.
Properties of an ESB

- Multi-Protocol Message Bus
  - XML - The Language of Integration
    - XML Schema
    - XML Binding
  - Transports & Connectors
    - TCP
    - HTTP
    - SOAP
    - FTP
    - RMI
    - ...

Evolution and Architecture of the Enterprise Service Bus

Adapted from *Enterprise Service Bus* - David A. Chappell
ESB Adoption: Partitioning and Encapsulating SOA

- Best Practices
  - Better, Faster, Lighter Java
  - Adopt XML as the language of integration
  - Think and plan in terms of SOA
  - Consistent definition of a “Service”
  - Manage “state” consistently
  - Separate processing logic from message routing rules
  - Incremental adoption
  - Propagate ESB based on business case
ESB Adoption: Partitioning and Encapsulating SOA

- Ski Clothing Company ("Classic Small Business")
  - QuickBooks
    - Accounting & Inventory
  - UPS Online
    - Shipping
  - Joe’s Credit Card Processing
    - Credit Card Processing
  - Custom Web Commerce Site
    - Online Catalog and Standalone Shopping Cart
ESB Adoption: Partitioning and Encapsulating SOA

Ski Clothing Company Topology

Hosting Provider

Windows
.NET Commerce
MS SQL Server

Internet

Ski Clothing Company

Windows UPS Online
Windows QuickBooks Credit Card
Web Admin

UPS Online
QuickBooks
Credit Card
Web Admin
ESB Adoption: Partitioning and Encapsulating SOA

- Ski Clothing Company Topology

```
Hosting Provider
- Windows
- .NET Commerce
- MS SQL Server

Internet

Ski Clothing Company
- Windows
- UPS Online
- QuickBooks
- Credit Card
- UPS Online
- Web Admin
```

- Conceptual Services
  - Bar Code Scanning Interface to Add Inventory
  - Automate Web Commerce Transaction
  - Automate Shipping Interface
  - Automate Simple CRM
ESB Adoption: Partitioning and Encapsulating SOA

- Bar Code Scanning/RFID Interface to Inventory
ESB Adoption: Partitioning and Encapsulating SOA

Automate Web Commerce Transaction

1. Procurement Service
   - Procurement endpoint
   - https

2. Credit card Service
   - Credit card endpoint
   - vm

3. Inventory Service
   - Inventory endpoint
   - vm

4. Shipping Service
   - Shipping endpoint
   - vm

5. Logging Service
   - Logging endpoint
   - vm

Hosting Provider

Ski Clothing Company
ESB Adoption: Partitioning and Encapsulating SOA

- Automate Simple CRM
  - Web Commerce Transaction
ESB Adoption: Partitioning and Encapsulating SOA

- Automate Simple CRM
  - Auto Responder

![Diagram showing the workflow process involving Auto Responder, Email endpoint, POP3, vm, and SMTP endpoints.](image)
ESB Related Standards: Impacts and Considerations

- Java Business Integration ("JBI")
- Java Connector Architecture ("JCA")
- Java Management Extensions ("JMX")
- Web Services ("WS-*")
  - BPEL - Business Process Execution Language
- WfMC - Workflow Management Coalition
- IP Protocol
ESB Related Standards: Impacts and Considerations

- Java Business Integration
  - JBI provides the core standards required to build SOA-based integration server software. It focuses on standardising the interoperation semantics (and associated bindings and interfaces) between what it calls service engines and a normalized message router, which links the engines together.
ESB Related Standards: Impacts and Considerations

- JBI Components
  - Service Engine
    - The term Service Engine (SE) is used to refer to the portion of the JBI environment (or framework) that is responsible for exposing an application programming model to developers.
    - JBI does not define an application programming model but rather defines a set of SPIs that enable the development of standard Web Service “containers” that can seamlessly utilize multiple communication infrastructure ranging from HTTP/SOAP [ref.] to JMS/MOM [ref.] and including specialized AS1/AS2 EDI [ref.] communications stacks.
  - Binding Component
    - A Binding Component may choose to implement one or more communications protocols thus offering connectivity services to SEs and thereby enabling SEs to expose their services to local and remote consumers as well as enabling the consumption of remote and local Web Services.
ESB Related Standards: Impacts and Considerations

- **JMX & Web based Admin tools**
- **SE Life cycle & Services**
- **Reconfiguration Deployment**
  - Control
  - Monitoring
- **BC Life cycle & Services**
- **Service Engine (SE) Framework**
  - BPEL SE
  - Other SEs
- **Normalized Message Service**
- **Binding Comp (BC) Framework**
  - WS-I BC
  - Other BCs
- **JBI Environment**
- **WSDL-based Service Providers & Consumers**

**Service Providers & Consumers**
- WS-I
- Other BCs
- BPEL SE
- Other SEs
ESB Related Standards: Impacts and Considerations

- Java Connector Architecture
  - JCA Container
    - Contains a JCA Resource Adapter (“RA”)
  - Resource Adapter
    - Application specific behavior
      - Connection Management
      - Security
      - Transactional Resources
ESB Related Standards: Impacts and Considerations

ESB

ESB JCA container

Management & Invocation

Endpoint

JCA

Adapter

JCA managed connection

Application
ESB Related Standards: Impacts and Considerations

- Java Management Extensions
  - Management and Monitoring of Applications and Resources
    - JMX Server
    - JMX Clients
    - Protocol Adapters and Connectors
    - JMX Connector Interface
    - Managed Beans ("MBeans")
ESB Related Standards: Impacts and Considerations

- Web Services

ESB Related Standards: Impacts and Considerations

- Specifications
  - Messaging Specifications
    - SOAP
    - WS-Addressing
    - MTOM (Attachments)
    - WS-Enumeration
    - WS-Eventing
    - WS-Transfer
    - SOAP-over-UDP
ESB Related Standards: Impacts and Considerations

Security Specifications

- WS-Security: SOAP Message Security
- WS-Security: UsernameToken Profile
- WS-Security: X.509 Certificate Token Profile
- WS-SecureConversation
- WS-SecurityPolicy
- WS-Trust
- WS-Federation
- WS-Federation Active Requestor Profile
- WS-Federation Passive Requestor Profile
- WS-Security: Kerberos Binding
- Web Single Sign-On Interoperability Profile
ESB Related Standards:
Impacts and Considerations

➢ Reliable Messaging Specifications
  • WS-ReliableMessaging

➢ Transaction Specifications
  • WS-Coordination
  • WS-AtomicTransaction
  • WS-BusinessActivity

➢ XML Specifications
  • XML
  • Namespaces in XML
  • XML Information Set

➢ Business Process Specifications
  • BPEL
ESB Related Standards:
Impacts and Considerations

➢ Metadata Specifications
  • WSDL
  • WS-Policy
  • WS-PolicyAssertions
  • WS-PolicyAttachment
  • WS-Discovery
  • WS-MetadataExchange

➢ Management Specifications
  • WS-Management
  • WS-Management Catalog

➢ Specification Profiles
  • Devices Profile
  • WS-I Basic Profile
ESB Related Standards:
Impacts and Considerations

Fundamentals of Service Orientation, Attachmate
ESB Related Standards: Impacts and Considerations

Workflow Management Coalition ("WfMC")

- Founded in August, 1993
- Non-profit, international organization of workflow stakeholders
- [http://www.wfmc.org](http://www.wfmc.org)
- Mission
  - Increase the value of customers’ investment with workflow technology
  - Decrease the risk of using workflow products
  - Expand the workflow market through increasing awareness for workflow
- Standards
  - Wf-XML
    - Super Framework pluggable process definition languages (e.g. BPEL, XPDL)
  - XPDL - Process Definition Language
    - [http://www.wfmc.org/standards/XPDL.htm](http://www.wfmc.org/standards/XPDL.htm)
  - BPEL4WS is another pluggable process definition language
Enterprise Service Bus Market: Open Source and Commercial

- Open Source
  - Mule
    - [http://mule.codehaus.org](http://mule.codehaus.org)
  - ServiceMix
    - [http://servicemix.org](http://servicemix.org)
  - jBPM - JBoss
    - [http://jboss.org/products/jbpm](http://jboss.org/products/jbpm)
  - OSWorkflow
    - [http://opensymphony.com/osworkflow](http://opensymphony.com/osworkflow)
Enterprise Service Bus Market: Open Source and Commercial

- **Open Source**
  - Werkflow
    - [http://werkflow.werken.com](http://werkflow.werken.com)
  - Agila/Twister - Apache Incubator
  - JBI - Sun Microsystems
    - [http://sun.java.com/integration](http://sun.java.com/integration)
  - Apache Synapse
Enterprise Service Bus Market: Open Source and Commercial

Core to Mule is the MuleManager. It manages the configuration of all core services for the Model and the components it manages. Below is an overview of the core components that comprise a Mule manager.
Enterprise Service Bus Market: Open Source and Commercial

Mule Architecture Overview

The primary goal is enable integration between applications using standards, open protocol and well-defined patterns. To achieve this goal, Mule defines a set of components that can be used to perform most of the hard work necessary to get disparate applications and services talking together.

This diagram shows a simple end-to-end topology of how Mule components fit together. For more details on how to design Mule applications see [Mule Topology Guide].
Enterprise Service Bus Market: Open Source and Commercial
Enterprise Service Bus Market:
Open Source and Commercial

ServiceMix

NMR Flows
Enterprise Service Bus Market: Open Source and Commercial

Apache Synapse

- This project implements the patterns that are often called Enterprise Service Bus or Broker. Since these are somewhat contentious terms, we call Synapse a Web service mediation framework and define the aims of the Synapse as follows:
  - a messaging and communications infrastructure;
  - based on the principles of service oriented architectures;
  - that provides a way of connecting disparate systems;
  - that manages the connections and the routing of messages between systems;
  - that intermediates and transforms messages and service interactions independently of the endpoint applications;
  - that is neutral with respect to the languages and platforms that are integrated - providing first class support for C/C++/COBOL/Java/.NET and other application platforms.

http://wiki.apache.org/incubator/SynapseProposal
Enterprise Service Bus Market: Commercial and Open Source

- **Commercial**
  - Sonic Software
    - SonicMQ (JMS), SonicESB, Sonic Orchestration
  - Tibco
    - Rendezvous
  - BEA
    - AquaLogic
  - Fiorano
    - FioranoMQ, Fiorano ESB, Fiorano Business Integration Suite
  - Cape Clear
    - Cape Clear ESB w/Orchestration Studio
  - webMethods
    - Enterprise Services Platform
  - IONA
    - Artix
Workflow Examples using ESB

Enterprise Integration Patterns using Mule

- Simple
  - Message Splitter (259)
  - Message Aggregator (268)
  - Publish Subscribe (106)

- Moderate
  - Composed Message Processor (294)
Workflow Examples using ESB

- Enterprise Integration Patterns
  - Message Splitter (259)
Workflow Examples using ESB

- Enterprise Integration Patterns
  - Message Aggregator (268)
Workflow Examples using ESB

- Enterprise Integration Patterns
  - Publish-Subscribe Channel (106)
Workflow Examples using ESB

- Enterprise Integration Patterns
  - Composed Message Processor (294)
References

- Software Engineering Institute, Carnegie Mellon
  - [http://www.sei.cmu.edu/](http://www.sei.cmu.edu/)
- Enterprise Service Bus
  - David A. Chappell - 2004
- Symphony Soft, Mule
  - Ross Mason & open source community
- Super-Peer Architectures for Distributed Computing
  - Fiorano - 2003
- The Evolution of Internet Scale Event Notification Services: Past, Present, and Future
  - Adam Rifkin and Rohit Khare
- JBI Specification - JSR 208
  - Sun Microsystems
- LogicBlaze, ActiveMQ, ServiceMix
  - James Strachen
- Constructing Software For Service Oriented Architecture, Attachmate
  - Jean-Jacques Dubray, Ph.D.
References

- JBI, The Only Game in Town
  - Ronan Bradley - 2005
- Microsoft Web Site
- Better, Faster, Lighter Java
  - Bruce Tate
- Workflow Management Coalition Specification
  - The Workflow Reference Model
  - [http://www.wfmc.org/standards/docs.htm](http://www.wfmc.org/standards/docs.htm)
- SEDA: An Architecture for Well-Conditioned, Scalable Internet Services
- The State of Workflow
  - Tom Baeyens, jBPM Founder
  - [http://jbpm.org/state.of.workflow.html](http://jbpm.org/state.of.workflow.html)
References

- Fundamentals of Service Orientation, Attachmate
  - Jean-Jacques Dubray
- Enterprise Integration Patterns: Designing, Building, and Deploying Messaging Solutions, Addison Wesley
  - Gregor Hohpe & Bobby Woolf
References

Theory In Practice
Enterprise Service Bus

David Chappell

2004