Enterprise Applications Integration Overview

EMS Users Group Meeting
Albany, NY
April 10, 2002
By: Joe Moran
Executive Consultant
Manager, Transmission Enterprise Services

Presentation Summary

- Introduction
- Common Problems
- Benefits of EAI
- Goals of EAI
- Integration Architectures
- Component Based Application Integration
- Integration Technology Overview
Introduction

• What is EAI?
  • “EAI provides the ability to read from and write to all of the applications and data sources across the enterprise”
  • Supports unified views of information and allows the update of information across systems
  • Provides for a single, consistent representation of common information
  • EAI is used to automate business processes

Introduction

• Types of EAI
  • Intra-enterprise integration (within the company)
    • Presentation level (standardizing user interface)
    • Database level (data warehouse or federated database)
    • Application level (A2A – uses a standardized messaging mechanism to move information between applications and databases)
  • Inter-enterprise integration (B2B) – between businesses
Common Problems

• Redundant Data
  - Multiple sources of redundant information located across the enterprise
  - Increased data maintenance
  - Inconsistency of information

• Inability to consolidate/integrate critical operations to achieve economies of scale
  - Lack of standards in existing systems
  - Inability for existing systems to “talk” to one another

Common Problems

• Islands of information
  - Lack of (poor) integration of services
  - Multiple data sources and storage facilities
  - Reduced efficiency
Islands of Information

Benefits of EAI

- Reduces cost of adding/changing components
- Allows for consistency of information used by all groups within the utility
- Reduces data maintenance by minimizing duplicate data sources
- Reduces errors caused by translating and re-entering data.
- Minimizes number of stand-alone and redundant systems.
- Improve business information efficiency
**Goals of EAI**

- Technology
  - Reusability
  - Scalability
  - Flexibility
  - Reliability
  - Availability
  - Performance

- Maintenance
  - Effort
  - Training needs
  - Ease of fixing identified problems

**Desired Outcomes**

- Reliable operation
- Reduce time needed to change/add components
- Provide a consistent view of information across the business systems
- Improve relationship between information systems and business processes
- Achieve seamless flow of information across the enterprise
Industry Standards

- EPRI Control Center API (CCAPI) and Common Information Model (CIM)
- IEC TC 57
  - WG 13 – EMS APIs
  - WG 14 – DMS APIs

EAI Design Concepts

- Transaction Managers
  - Message Bus/Broker (request/reply, publish/subscribe, queues)
  - Process Automation (automating work flows)
- Toolkits and Metadata
  - Adapters (pre-built Interfaces)
  - Data definitions
- Data transformation (validate, translate data)
- Message routing
- Connectivity (persistency, secure messages, etc.)
- Events handling
System Integration Approaches

• Presentation Level
• Application Level
  ▶ Component based integration
• Data Level
  ▶ Data warehouse
  ▶ Federated databases

Integration Architectures

Major methods of systems integration include:
  ▶ Point-to-point
  ▶ Message Bus/Message Broker configuration
  ▶ Distributed integration architecture
    ◆ Pure distributed architecture
    ◆ Federated architecture
Component Based Application Integration

- **Component Model**
  - Defines the basic architecture of a component
    - Specifies structure of its interfaces
    - Specifies the mechanism by which it interacts with its container
- **Primary Component Models include:**
  - Common Object Request Broker Architecture (CORBA)
  - Distributed Common Object Model (DCOM)
  - Enterprise Java Beans (EJB)

Middleware

- Middleware can be described as:
  - The transport software used to move information from one program to one or more additional programs
  - Provides the infrastructure necessary for applications to exchange data, regardless of the environment in which they are running.
**Middleware Products**

- Transaction processing monitors
- Remote procedure calls
- Object request brokers
- Message-oriented Middleware (MOM)

**Message Queuing**

![Message Queuing Diagram]

- Application 1
- Application 2
- System A
- System B
- Queue
- Message Queuing Subsystem
Advantages of Message Queuing

• Simple connection from one application to another
• Interface is modular. Changes on one side barely affect other components
• Guaranteed delivery
• Supports transactions
• Relatively easy to use

Advanced Middleware Products/Vendors

• Vitria Technologies
• TIBCO
• IBM MQIntegrator
• Compaq’s Business Bus
• BEA (E-link)
• MS BizTalk Server
• Sagavista
• SeeBeyond
State of Use in Utilities

- Starting to see moves towards use of CIM
- Starting to see more data and component integration and away from point to point
- Some movement towards components – mostly starting in network apps
- EMS still sold as an integrated solution
- Some movement towards unbundling some functionality

Thank you