Composite Application Architecture

March, 2002
Adgenda

• Business Scenario
• Application Federation
• Service Delivery and Consumption
• Composite Application Architecture
• Standards and Summary
Goal – combine sell side services with those of the buying organization to create a seamless procurement environment.
Participants

• Buying Organization
  – Be able to do business with a large number of suppliers
  – Define how services are used
  – Alter services based on internal policies
  – Be insulated from choice of sell-side platform

• Product Supplier
  – Maintain product catalog
  – Publish services to assist buyer
  – Control how products are presented
  – Be insulated from choice of buying platform
XML Web Services make it easy to access services in independent domains.

Federation provides key source of power of Web Services.

Each domain defines:
- Authentication
- Transactions
- Management
- Service Definition

Goal – combine these services into cohesive applications.

Federated Application Challenge

Partner 1
Partner 2
Partner 3
Areas of Federation

• Base Level
  – Authentication, User Domains
  – Session Management
  – Transactions

• Operations and Management
  – Maintenance and Versioning
  – Availability
  – Visibility

• Service Delivery and Consumption
  – Interface Definition
  – Granularity
  – Behavior and Sequencing
Coarse-grained Services

- Current XML Services Fairly Low Level
  - GetStockQuote
  - CheckInventoryLevel
  - TemperatureConversion
- Large Grained Services Required
  - Analyze a portfolio
  - Configure a product
  - Provision a telephone switch
- Service Characteristics
  - Stateful/Transactional
  - Asynchronous
  - Interactive
Interactive Services

• Computational Service – purely machine to machine, calling application responsible for user interaction

• Interactive Service – purely machine to machine to human, called service handles user interaction, calling application can influence this interaction
Service Adaptation

• Adaptation needed to reuse large grained services
  – Without adaptation, services too monolithic
  – Regain flexibility of fine-grained services
  – Consumer determines relevant adaptations

• Adaptation points
  – Service API – returned document
  – Service behavior and flow
  – Service UI
Coarse-grained Adaptation

Producer

WSIA Application

WSIA Runtime

Integrator

Data and presentation properties

HTTP Proxy

WSIA Application

WSIA Runtime

Users

HTTP

WSIA Application

HTTP

Stock Application

WSIA Runtime

52 week high: 123.00
low = 55.25
Fine-grained Adaptation

Application Provider

Stock Plot Application

WSIA Application

WSIA Runtime

Application Distributor

Users

WSIA Data

WSIA Pres

WSIA Control

HTTP (SOAP)

HTTP (HTML)
Simple Application Architecture

- Authorization
- Personalization
- Transaction Management
- Page Generation
- Business Logic
- Session Management

Service Access

Simple Web Services

- Computational
- Stateless
- Non-transactional
Composite Application Architecture

- Service Access
- Domain Federation (Authentication, Transactions, Sessions)
- Service Proxying
- Service Composition
- Service Adaptation
- Personalization
- Business Logic
- Transaction Management
- Session Management
- Page Generation
- Authorization
- UI Composition

- Interactive
- Stateful
- Transactional
- Federated

Domain 1
Domain 2
Domain 3

Real World Web Services
Relevant Standards

• Messaging Standards
  – SOAP(W3C), BTP (OASIS), Routing, Security, …
• Workflow Driven Standards
  – WSFL (IBM), XLANG (Microsoft)
• User Driven Standards
  – Web Services for Interactive Applications (OASIS)
  – Goals:
    • XML and web services centric model for interactive web applications.
    • Enable businesses to distribute web applications through multiple revenue channels, and
    • Enable new services or applications to be created by leveraging existing applications across the Web
Q&A