Patterns for e-business: Leveraging Architectural Patterns in Defining Enterprise Architecture & Solution Architectures

Guru Vasudeva
Architecture Center of Excellence - Federal
IBM Global Services

e-mail: vasudeva@us.ibm.com
Phone: 301-803-3181

Patterns for e-business web site:
http://www.ibm.com/developerworks/patterns
Agenda

- What are Patterns?
- IBM Patterns for e-business Overview
- Leveraging architectural patterns in Solution Architecture definition
- Applying architectural patterns at Enterprise Architecture level
What are Patterns?

✓ A pattern is a "generalized, named problem to solution mapping"

✓ A pattern captures a successful solution to a repeating problem in a particular context

✓ In general, a pattern is documented using the following sections:
  – **Name:** A name used for identification
  – **Problem:** A repeating problem that occurs in a domain
  – **Solution:** Best practice solution to that problem
  – **Consequences:** Advantages and disadvantages of the recommended solution
  – **Examples:** A few examples where the recommended solution has already been applied

✓ Christopher Alexander's research on buildings and town design is often considered the pioneering work on pattern based thinking
Why are patterns important?

✓ A pattern is a reusable generalization (or abstraction) that can be used as the starting point in future solutions.

✓ Benefits of patterns

- Provide a mechanism to capture knowledge and experience
- Provide a common vocabulary amongst architects and designers
- Facilitate reuse of approaches that have been successful elsewhere and thus contribute towards the following aspects of a project:
  - Reducing risk
  - Increasing quality
  - Improving delivery time

"One thing expert designers know not to do is to solve every problem from first principles. Rather, they reuse solutions that have worked for them in the past. When they find a good solution, they use it again and again. Such experience is part of what makes them experts."

Design Patterns, Gamma, Helm, Johnson & Vlissides 1995
There are four main categories of "patterns" in Software Engineering:

**Reference architectures**
- Industry-related:
  - Insurance
  - Banking
- Technical:
  - ESS Reference Arch.

**Architectural Patterns**
- Business Application Focused:
  - Patterns for e-business
- Abstract Technology Focused:
  - Layers
  - Pipes & Filters

**Design Patterns**
- Abstract Factor
- Proxy
- Facade

**Analysis Patterns**
- Party
- Organization
- Account

A reference architecture is often composed of multiple architectural patterns. Similarly, an architectural pattern may leverage multiple design patterns and analysis patterns.
Agenda

✓ What are Patterns?
✓ IBM Patterns for e-business Overview
✓ Leveraging architectural patterns in Solution Architecture definition
✓ Applying architectural patterns at Enterprise Architecture level
Patterns for e-business identify and document proven architectural patterns & design guidelines gleaned from successful project.

These patterns are like Lego blocks that help us to assemble solutions rapidly and achieve high levels of best practice & component reuse within an enterprise.

“Patterns for e-business – A Strategy for Reuse” Book
Jonathan Adams, Srinivas Koushik, Guru Vasudeva, and George Galambos

Patterns for e-business: Business Patterns

Business patterns identify the most commonly observed interactions between Users, Businesses, and Data.

- **Self Service** (a.k.a. User to Business)
  - Users accessing transactions on a 24x7 basis

- **Collaboration** (a.k.a. User to User)
  - Users working with one another to share data and information

- **Information Aggregation** (a.k.a. User to Data)
  - Data from multiple sources aggregated and presented across multiple channels

- **Extended Enterprise** (a.k.a. Business to Business)
  - Integrating data and processes across enterprise boundaries
Patterns for e-business: Integration Patterns

Complex applications can be built by combining multiple Business patterns together. This is accomplished by using Integration patterns as the "glue" between Business patterns.

**Access Integration**

Provide seamless and consistent access to business functions.

Typical functions provided include Single-signon, Personalization, Transcoding etc.

**Application Integration**

Connect, Interface or Integrate databases and systems

Typical integration can be based on Function, Type of Integration, Mode of Integration and by Topology
Patterns for e-business: Cardinality between different types of Patterns

**Business & Integration Patterns**

- **Self Service**
  (a.k.a. User to Business)

- **Collaboration**
  (a.k.a. User to User)

- **Information Aggregation**
  (a.k.a. User to Data)

- **Extended Enterprise**
  (a.k.a. Business to Business)

**Cardinality between different types of Patterns**

- **Business or Integration pattern**
- **Application pattern**
- **Runtime pattern**
- **Runtime pattern**
- **Runtime pattern**
Each of the Business & Integration patterns can be implemented using one or more application patterns. Each Application pattern describes structure, placement, & integration between logical tiers of the application.
Legend used for expressing Application Patterns

**Application node containing new or modified components**

**Read only data**

**Application node containing existing components with no need for modification or which cannot be changed**

**Transient data**
- Work in progress
- Cached committed
- Staged

**A process that is external and transparent to the current application. However the interfaces to this process are well-defined**

**Read write data**

**Meta Data**
- Templates
Application Patterns for **Self-Service**

**Stand Alone Single Channel**

1. Pres. → App.  synch

**Directly Integrated Single Channel**

2. Pres. → App.  synch

3. App. → App.  synch/asynch

**Router**

5. Pres1 → Pres2 → App1  synch

6. Pres1 → Decomp → App1  synch/asynch

**Decomposition**

6. Pres1 → Decomp → App1  synch/asynch

**Agent**

7. Pres1 → Agent  synch/asynch

8. Agent → App1  synch/asynch
Patterns for e-business: You choose between one or the other Application Pattern based on Business & IT Drivers.

The following Business & IT drivers guide organizations to choose Self-Service::Router Application Pattern

✓ Business Drivers
   – Improve organizational efficiency
   – Reduce the latency of business events
   – Easy to adapt during mergers and acquisitions
   – Integration across multiple delivery channels

✓ IT Drivers
   – Minimize total cost of ownership (TCO)
   – Leverage existing skills
   – Leverage legacy investment
   – Back end application integration
   – Minimize enterprise complexity
   – Maintainability
   – Scalability
Application Patterns for Information Aggregation

Population – Single Step

Population – Crawl and Discover

Population – Multiple Step

Population – Summarization
Application Patterns for Extended Enterprise

Document Exchange

Exposed Application

Exposed Business Services

Managed Public Process
Application Patterns for Collaboration

Point-to-Point Collaboration

Collaboration Client

Local Settings

Collaboration Client

Local Settings

Store and Retrieve Collaboration

Collaboration Client

Local Data

Collaboration Server

Shared Data

Directed Collaboration

Collaboration Client

Local Address Book

Collaboration Client

Local Address Book

Collaboration Server

Shared Application

Sync

Shared Application

Sync

Sync

Shared Application

Sync

Shared Application

Sync

Managed Collaboration

Collaboration Client

Workflow Application

Collaboration Server

Workflow Rules

Local Address Book

WIP Data

Directory

Shared Data

Local Address Book

Directory

Shared Data

© IBM Global Services, 2003

Architecture Center of Excellence - Federal

17
Application Patterns for Access Integration

Pervasive Device Access

Web Single Sign-On

Extended Single Sign-On

Personalized Delivery
Application Patterns for **Application Integration (Process Focused)**

**Direct Connection**
- Application Tier
- Application Tier
- Synchronous
- Asynchronous

**Broker**
- Application Tier
- Broker Rules Tier
- Application Tier
- Application Tier
- 1:N
- Read-write
- Work in Progress
- Read-write

**Managed Process**
- Application Tier
- BPM Rules Tier
- Application Tier
- Application Tier
- Read-write
- Work in Progress
- Read-write
- Read-write

© IBM Global Services, 2003
Application Patterns for Application Integration (Data Focused)

Propagation

Replication

Operational Data Store

Federated Repository

App 1

App 2

App 3

App N

Transform Rules

Population

Feedback

Population

Transform Rules

Unified Query

App 1

App 2

App N
Patterns for e-business: Runtime Patterns

Runtime Patterns capture the logical topology of the architecture that is necessary to support a selected Application Pattern. It identifies software, middleware, and hardware nodes required based on a set of Service Level Characteristics (e.g. availability, security, etc.)
Self-Service: Application Patterns, Runtime Patterns & Product Mapping

7 Application Patterns for Self-Service
- Stand Alone Web-Up
- Directly Integrated Web-Up
- As-is Host
- Customized Presentation
- Router
- Decomposition
- Agent

Self-Service::Router Application Pattern
Select based on Business & IT Drivers

Self-Service::Router Product Mapping & Redbook

Self-Service::Router Runtime Pattern
Select based on Service Level Characteristics

© IBM Global Services, 2003
Extended Enterprise: Application Patterns & Runtime Patterns

5 Application Patterns for Extended Enterprise

- Document Exchange
- Exposed Application
- Exposed Business Services
- Managed Public Process

Managed Public Process Application Pattern

Select based on Business & IT Drivers

Managed Public Process Runtime Pattern

Select based on Service Level Characteristics
Patterns for e-business: Composite Patterns

Customer Requirements

Does a Composite Pattern already exist?

Composite patterns

Business patterns

Integration patterns

Business Actors & Interactions
Patterns for e-business: Composite Patterns

Composite Pattern for Electronic Commerce Solutions
- Self Service
- Collaboration
- Information Aggregation
- Extended Enterprise

Composite Pattern for Portals
- Self Service
- Collaboration
- Information Aggregation
- Extended Enterprise

Composite Pattern for Enterprise Intranet Portal
- Self Service
- Collaboration
- Information Aggregation

Composite Pattern for a Collaboration ASP
- Collaboration

Blue – Mandatory Business Patterns
Red – Optional Business Patterns - Variations

© IBM Global Services, 2003
Agenda

- What are Patterns?
- IBM Patterns for e-business Overview
- Leveraging architectural patterns in Solution Architecture definition
- Applying architectural patterns at Enterprise Architecture level
Technique for leveraging Patterns for e-business in Solution Architecture definition

This technique and the associated steps outlined below are further elaborated with case studies in our book “Patterns for e-business – A Strategy for Reuse”

- Step 1: Develop High Level Business Description
- Step 2: Develop Solutions Overview Diagram
- Step 3: Identify Business Patterns
- Step 4: Identify Integration Patterns
- Step 5: Identify Composite Patterns
- Step 6: Identify Application Patterns
- Step 7: Summarize all the Application Patterns Required
- Step 8: Integrate a package into the Solution
- Step 9: Develop the Technical Architecture
Example: e-Gov GeoSpatial Information One-Stop Initiative

The Geospatial Information One-Stop will provide access to the federal government's spatial data assets in a single location and help make state and local spatial data assets more accessible. Federal agencies will also make their planned and future spatial data activities available to state and local governments to promote collaboration and reduce duplicative efforts. Data standards developed through an intergovernmental process will result in data that can be used multiple times for multiple purposes, saving taxpayer money. It will also help empower the private sector by communicating the characteristics of a desired standardized data product.
e-Gov GeoSpatial Initiative – Step 1 & 2 Develop SOD

- Search
  - Metadata About available Information (ODS)
- Download & Exploit
  - Virtual Knowledge Store (Federation)
- Request for New Data
  - Supply Chain Management
- Various National Libraries
- Sensors / Data Providers
- National Technical Means
- Commercial Sources
- Data Provider Specific Tasking

GeoSpatial Information
Users from Local & State Govt.
e-Gov GeoSpatial Initiative – Step 3 Identify Business Patterns

Self-Service

Search

Metadata About available Information (ODS)

Virtual Knowledge Store (Federation)

Download & Exploit

Request for New Data

Supply Chain Management

Information Aggregation

Extended Enterprise

Various National Libraries

Sensors / Data Providers

National Technical Means

Commercial Sources

Data Provider Specific Tasking

GeoSpatial Information Users from Local & State Govt

Extended Enterprise Information Aggregation Self-Service
e-Gov GeoSpatial Initiative – Step 4 Identify Integration Patterns

Self-Service

Access Integration

Search

Download & Exploit

Metadata About available Information (ODS)

Virtual Knowledge Store (Federation)

Request for New Data

Supply Chain Management

Extended Enterprise

Various National Libraries

Sensors / Data Providers

National Technical Means

Commercial Sources

Data Provider Specific Tasking

Self-Service

Extended Enterprise
e-Gov GeoSpatial Initiative – Step 6 & 7 Identify Application Patterns

- **Search**
- **Download & Exploit**
- **Request for New Data**
- **Supply Chain Management**
- **Virtual Knowledge Store (Federation)**
- **Metadata About available Information (ODS)**
- **Application Integration:: Federated Repository**
- **Information Aggregation:: Multi-Step Population**
- **Access Integration:: Single-Sign On & Personalized Delivery**

- **Self-Service ::Agent**
- **Self-Service ::Router**
- **Extended Enterprise :: Managed Public Process**
  - Various National Libraries
  - Sensors / Data Providers
  - National Technical Means
  - Commercial Sources

**GeoSpatial Information Users from Local & State Govt**

© IBM Global Services, 2003
Identify runtime patterns for the following selected application patterns and combine them into an integrated architecture.

**Access Integration:: Web Single Sign-On**

- **Client Tier**
- **Single Sign-On Tier**
  - Synchronous
- **Application1**
- **Application2**
  - Read-Write

**Access Integration:: Personalized Delivery**

- **Client Tier**
- **Personalization Rules**
- **Application1**
- **Application2**
- **Application Integration:: Federated Repository**
  - App 1
  - Unified Query
  - App 2
  - App N

**Self-Service:: Agent**

- **Pres1**
  - Synchronous/asynchronous
- **Agent**
  - WIP
- **App 1**
- **App 2**
- **App N**

**Extended Enterprise:: Managed Public Process**

- **Public Process Rules**
- **Public Tier**
- **Private Processes**
- **Public Processes**

- **Partner A**
  - Backend Application
  - Synchronous/Asynchronous
  - N:1

- **Partner B**
  - Backend Application
  - Synchronous/Asynchronous
  - 1:N

- **Asynchronous, Server-specified or mutually agreed message format**

**Application Integration:: Federated Repository**

- **Unified Query**
  - App 1
  - App 2
  - App N
Agenda

- What are Patterns?
- IBM Patterns for e-business Overview
- Leveraging architectural patterns in Solution Architecture definition
- Applying architectural patterns at Enterprise Architecture level
 Enterprise Architecture

From the OMB FEA and Reference Models perspective Layered Architectural Patterns documented by Patterns for e-business can be used to establish a clear linkage between BRM and SRM/TRM.
Applying architectural patterns at Enterprise Architecture level

At Enterprise Architecture level, Patterns for e-business can be used for analyzing a portfolio of applications. Based on this analysis one can identifying frequently occurring architectural patterns in the enterprise and thus establish prescriptive target architectures for those category of applications.

- Step 1: Develop enterprise level Application Function Model
- Step 2: Identify Business Patterns
- Step 3: Identify Integration Patterns
- Step 4: Based on this analysis, identify frequently occurring architectural patterns in the application portfolio
- Step 5: Establish prescriptive target architectures for frequently occurring category of applications in the portfolio
Insurance Company Example: Application Function Model

Customer Facing Functions:
- Quotes
- Policy Inquiry
- Billing Inquiry
- Electronic Bill Payment
- Claim Submission
- Application Submission
- Claims Inquiry

Agent Productivity Tools:
- Contact Management
- Customer Acquisition
- Commission Management

Employee Self-Service:
- Personal Profile
- 401k Management
- Benefits Management

Disability LOB Engines:
- Rating
- Claim Processing
- Billing
- Policy Processing
- Actuarial

Life LOB Engines:
- Rating
- Claim Processing
- Billing
- Policy Processing
- Actuarial

P&C LOB Engines:
- Rating
- Claim Processing
- Billing
- Policy Processing
- Actuarial

Operational Data Stores:
- Integrated Customer Info
- Integrated Product Info

Analysis:
- Enterprise Data Warehouse
- Decision Support Systems
- Direct Marketing

Corporate Functions (Back Office):
- Accounting
- Financials
- Actuarial
- Law & Compliance
- HR Management

Insurance eMarketplaces
Insurance Company Example: Identify Business Patterns

Customer Facing Functions
- Quotes
- Electronic Bill Payment
- Policy Inquiry
- Claim Submission
- Billing Inquiry
- Application Submission
- Claims Inquiry

Agent Productivity Tools
- Contact Management
- Customer Acquisition
- Commission Management

Employee Self-Service
- Personal Profile
- 401k Management
- Benefits Management

P&C LOB Engines
- Rating
- Underwriting
- Billing
- Policy Processing
- Claims Processing
- Actuarial

Disability LOB Engines
- Life LOB Engines

Operational Data Stores
- Integrated Customer Info
- Integrated Product Info

Analysis
- Enterprise Data Warehouse
- Decision Support Systems
- Direct Marketing

Corporate Functions (Back Office)
- Accounting
- Financials
- Actuarial
- Law & Compliance
- HR Management
Insurance Company Example: Identify Integration Patterns

Customer Facing Functions
- Quotes
- Electronic Bill Payment
- Policy Inquiry
- Claim Submission
- Billing Inquiry
- Application Submission
- Claims Inquiry

Agent Productivity Tools
- Contact Management
- Customer Acquisition
- Commission Management

Employee Self-Service
- Personal Profile
- 401k Management
- Benefits Management

Extended Enterprise
- Disability LOB Engines
- Life LOB Engines
- P&C LOB Engines
  - Rating
  - Underwriting
  - Billing
  - Policy Processing
  - Claims Processing
  - Actuarial

Operational Data Stores
- Integrated Customer Info
- Integrated Product Info

Analysis
- Enterprise Data Warehouse
- Decision Support Systems
- Direct Marketing

Corporate Functions (Back Office)
- Accounting
- Financials
- Actuarial
- Law & Compliance
- HR Management

Access Integration

IBM

© IBM Global Services, 2003
Architecture Center of Excellence - Federal 39
Patterns for e-business can be customized based on agency technology choices to provide prescriptive guidelines (not just WHAT technologies to use but also HOW to use it)

**Technical Reference Model (TRM):** Defines Agency Technology and Product Choices

**Customized Runtime Patterns** reflect the technology choices from the TRM and provide clear guidance for Systems Integrators

**Technical Reference Model:**
Also defines the technology adoption life cycle

- **Current**
- **2 Years**
- **5 Years**

- **Current**
- **Tactical**
- **Strategic**
- **Retirement List**
- **Mature Technologies & Standards**
- **Containment List**
- **Emerging Technologies**
Leveraging Patterns for e-business in defining Enterprise Architecture and Solution Architectures provides a number of benefits

- Easily link Business Architecture to Technology Architecture
- Promotes convergence of multiple business applications on similar architecture by providing clear direction to application development groups and systems integrators.
- Knowledge Management: Promotes sharing and reuse of best practices between application groups.
- Reduces risk and cost of multiple projects in an organization.
- Improves the quality and delivery speed of multiple projects in an organization.
- Enables server consolidation through convergence on similar architecture thus resulting in reduced Total Cost of Ownership (TCO).
- Leverage scarce technical skills across multiple similar projects.
- Enables effective architecture governance process.
Thank You

Further information can be found at:


- “Patterns for e-business – A Strategy for Reuse” Book
  Jonathan Adams, Srinivas Koushik, Guru Vasudeva, and George Galambos