Agenda

- Introduction
- Services Engineering Process
- Deliverables by Stages
Services Lifecycle requires seamless flow of information between the three areas of responsibilities:

1. Accurately capture the business requirements
2. Develop the IT solutions to business requirements
3. Deploy and maintain the service to business requirements

Initial Priority:

- Accurately capture the business requirements, not only the business capabilities but also the operations needs
- Eliminate the communication gap between the business and development team, enabling the development team providing business with exactly what they requested as well as reduce cost by reusing services
- Provide capability to build out the operations infrastructure on an as needed basis
## High-Level requirements for each of the three areas of responsibilities

<table>
<thead>
<tr>
<th>Area of Responsibility</th>
<th>Requirements</th>
<th>Potential Tools</th>
</tr>
</thead>
</table>
| **Line-Of-Business (LOB)** | • Visually map business process starting from Level 0 down  
• Define each of the business process  
• Identify business owners for each of the processes  
• Identify objectives and current business services gaps  
• Map Input and Output data elements  
• Prioritize business process and business services  
• Capture all aspects of business service definitions  
• Simulate user interface and/or business processes | • Business Process Mapping  
• Requirements Capture  
• Simulation: User Experience  
• Simulation: Business Process |
| **Solution Development** | • Review requirements and identify alternatives for each of the business process  
• Design and estimate each of the components for the business services, examples are Portal Services, Integration Services, Infrastructure Services, Data Services, policy service and business (logic) services  
• Identify reuse opportunities for business services  
• Develop and execute to detailed project plan  
• Track and report progress to Business & IT Management  
• Obtain business sign-off of delivery of each of the business services | • Accept and estimate rqmts.  
• Identify Business Service Components: Portal Services, Integration Services, etc.  
• Service Repository  
• Business Services – Portfolio Management  
• Design, development & build tools  
• QA/Testing tools |
| **Service (IT) Operations** | • Review requirements and identify infrastructure needs  
• Establish systems environment consisting of development, system integrations testing, performance testing, user acceptance and product environments  
• Assist solutions development teams in systems/application configuration, periodic builds, capacity planning, etc.  
• Track and manage dependencies between services and  
• Deploy and manage business services in production  
• Provide application support for business services based on business priority | • Capacity Planning  
• Environment management  
• Business Service Provisioning and configuration  
• Monitoring and dependency management |
Review Requirements

Architecture Review
- Review Requirements
- Review Alternatives & Estimate effort

Prioritize Business Services

Capture Business Services Requirements

Propose Solution

Prioritize and add to Solutions Portfolio

Map high-level Business Processes

Prioritize Business Services

Capture Business Services Requirements

Develop, QA and conduct UAT for business solution

Design solutions – identify reuse opportunity

Assign Resources to Solutions Development Team

Assign Resources to Service Operations Teams

Maintain Solution to Business Requirements

Deploy business solution

Identify infrastructure needs and establish systems environment

Design and Development

Requirements and Analysis

IT Operations
Agenda

- Introduction
- Services Engineering Process
- Deliverables by Stages
Service Engineering Process

- Capture and document high-level business requirements
- Map the business requirements to the business process

Map existing business services to the business process
Every business services could be implemented as one or multiple physical / business services as illustrated. Re-use existing Physical/ Business services wherever possible.

Every service has assets associated with it.
Service Engineering Process

Generate the Service Assembly Model which includes creating the logical deployment model.

The Service Assembly Model shall consist only of the approved services.

- Business Requirements
  - Business Services
  - Business Process
  - Executable Services
  - Business Process
  - Executable Services
  - Service Assets
  - Service Logic
  - Submit developed services
  - Approved Services
  - Approved Services
  - SOA Repository
  - Service Assembly

- Product Domain Models
  - Product A
  - Product B
  - Product C
  - Product 1
  - Product 2

- Portal
- ESB
- EII
- Data Services
- ETL
- Infrastructure
- I&AM
- DQ
Service Engineering Process

Services Assembled is deployed in production by the operations teams.

Each of the deployed services is monitored to map back to business services.

Requirements / Use case
Design Specifications
Inputs & Outputs
Data elements required
Dependent on services
Service used by
Versions
Source code (location)
Builds (location)
Product Type

Product Domain Models
Approved Services
Service Assembly
Service Logic
Service Assets
Executable Services
Service Repository
Business Process
Business Services
Business Requirements

Submit developed services

Develop missing services

Product Domain Models
Portal
ESB
Product A
Product B
Product C
Product 1
Product 2
Data Services
I&AM
EII
ETL
DQ
Infrastructure
Service Engineering Process

Service Matrix

Business Services

Business Process

BAM

Executable Services

Service Logic

Service Assembly

SOA Repository

Service Assets

Develop missing services

Submit developed services

Product Domain Models

Approved Services

Product A

Product B

Product C

Product 1

Product 2

Portal

ESB

Data Services

EII

I&AM

ETL

Storage

DQ
Service Engineering Process: Information Model

Map existing business services to the business process

Sample representation of an Information Model (data flow) by mapping the input, output and the entities (data elements) to each of the business processes
The information model is translated into the data flow for further development and could also be mapped to the business processes.

Re-use existing Physical/Business services wherever possible.

The rest of the process remains the same.

Mapping data services to business processes facilitates developing/identifying shared data services.

Detailed data matrix.
## Services Lifecycle: Actors/Roles

<table>
<thead>
<tr>
<th>Capabilities</th>
<th>LOB</th>
<th>CIO</th>
<th>LOB/IT</th>
<th>PMO</th>
<th>PM</th>
<th>EA</th>
<th>BA</th>
<th>Dev</th>
<th>RM</th>
<th>IT Oper</th>
</tr>
</thead>
<tbody>
<tr>
<td>IT Governance</td>
<td>I</td>
<td>A</td>
<td>C</td>
<td>R</td>
<td>C</td>
<td>C</td>
<td>I</td>
<td>I</td>
<td>I</td>
<td>C</td>
</tr>
<tr>
<td>Application Portfolio Management</td>
<td>C</td>
<td>C</td>
<td>A</td>
<td>R</td>
<td>I</td>
<td>I</td>
<td>I</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Requirements Gathering</td>
<td>C</td>
<td>C</td>
<td>A</td>
<td>I</td>
<td>R</td>
<td>I</td>
<td>R</td>
<td>I</td>
<td>I</td>
<td>I</td>
</tr>
<tr>
<td>Business Process Modeling</td>
<td>C</td>
<td>C</td>
<td>A</td>
<td>C</td>
<td>R</td>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>User Experience Simulation</td>
<td>C</td>
<td>C</td>
<td>A</td>
<td>C</td>
<td>R</td>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data Flow Modeling</td>
<td>C</td>
<td>C</td>
<td>A</td>
<td>R</td>
<td>C</td>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Business Activity Monitoring</td>
<td>C</td>
<td>C</td>
<td>A</td>
<td>C</td>
<td>C</td>
<td>R</td>
<td>I</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Applications / Systems / Services Blueprint</td>
<td>I</td>
<td>A</td>
<td>C</td>
<td>C</td>
<td>C</td>
<td>R</td>
<td>C</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Architecture Frameworks / Standards</td>
<td>I</td>
<td>C</td>
<td>C</td>
<td>C</td>
<td>C</td>
<td>AR</td>
<td>I</td>
<td>I</td>
<td>I</td>
<td>C</td>
</tr>
<tr>
<td>Service Orchestration Modeling</td>
<td></td>
<td></td>
<td>AR</td>
<td>C</td>
<td>I</td>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Service Assembly Modeling*</td>
<td>I</td>
<td>A</td>
<td>I</td>
<td>C</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Service Network Modeling (Deployment Architecture)</td>
<td>C</td>
<td>I</td>
<td>C</td>
<td>I</td>
<td>C</td>
<td>A</td>
<td>I</td>
<td>R</td>
<td>C</td>
<td></td>
</tr>
<tr>
<td>Assign Service Composition/Development</td>
<td></td>
<td></td>
<td></td>
<td>A</td>
<td>R</td>
<td>C</td>
<td>C</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Service Composition</td>
<td></td>
<td></td>
<td></td>
<td>I</td>
<td>C</td>
<td>C</td>
<td>AR</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Service Development</td>
<td></td>
<td></td>
<td></td>
<td>I</td>
<td>C</td>
<td>C</td>
<td>AR</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Composite Applications: Debugging and testing</td>
<td></td>
<td></td>
<td></td>
<td>I</td>
<td>C</td>
<td>C</td>
<td>C</td>
<td>AR</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Deployment Management and Configuration</td>
<td>I</td>
<td>I</td>
<td>R</td>
<td>I</td>
<td>R</td>
<td>C</td>
<td>C</td>
<td>R</td>
<td>A</td>
<td></td>
</tr>
<tr>
<td>Monitoring and Management</td>
<td>I</td>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Business Activity Management</td>
<td>I</td>
<td>A</td>
<td>R</td>
<td></td>
<td>C</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Responsible (R) - Responsible, Accountable (A) - Accountable, Consulted (C) - Consulted, Informed (I) - Informed**
Agenda

- Introduction
- Services Engineering Process
- Deliverables by Stages
Deliverables by Application Lifecycle stages

**Feasibility Analysis**

**Activities**
- Organize business owners and establish project sponsorship
- Define full set of requirements
- Iterate on technical and cost feasibility until final scope is determined
- Create project work plan and identify delivery date
- Obtain signoff on requirements, schedule and cost

**Deliverables**
- Application Requirements
- Project Scope

**Exit Criteria**
- Fully funded project
- Assigned resources for the entire duration of the project

© Copyright 2008, Pai Systems, LLC
Deliverables by Application Lifecycle stages

Feasibility Analysis → Req. Analysis → Design → Dev. → QA → Release

Requirements Analysis

Activities
- Create functional specifications that define how the application will behave
- Complete the User Interface design and site map
- Revise project plan as necessary based on design
- Obtain business signoff on functional specification

Deliverables
- Functional Specification

Exit Criteria
- Business signoff on requirements and delivery date
- Program signoff on Infrastructure usage
- Program commitment to delivery date
- Architecture Review Board signoff (technical team accepting requirements)
Deliverables by Application Lifecycle stages

**Development**

**Activities**
- Application Design
- Data Model
- Design patterns and components

**Deliverables**
- Technical design document

**Exit Criteria**
- Architecture Signoff:
  - Infrastructure
  - Database
  - Network
- User Interface signoff
- Development Environment Ready
- Architecture Review Board signoff

© Copyright 2008, Pai Systems, LLC
Deliverables by Application Lifecycle stages

Development

Activities
• Complete development
• Perform unit level testing

Deliverables
• Development assets
• Configuration Documentation

Exit Criteria
• Unit level testing cleared
• Code complete for all functionality
• Business Analyst documented all use/test cases
• Architecture Review Board signoff

© Copyright 2008, Pai Systems, LLC
Deliverables by Application Lifecycle stages

**Feasibility Analysis**

**Req. Analysis**

**Design**

**Dev.**

**QA**

**Release**

**Design**

**Activities**
- Plan application testing (System, User, Performance)
- Execute System Test
- Execute User Acceptance Test
- Complete Performance test
- Obtain business signoff for launch

**Deliverables**
- Business Approved Application

**Exit Criteria**
- Business signoff on application
- User Acceptance Test cleared
- Program signoff on performance and release plan

© Copyright 2008, Pai Systems, LLC
Deliverables by Application Lifecycle stages

**Design**

**Activities**
- Complete detailed release plan and schedule
- Prepare production environment for launch of new application
- Create Post-Production support matrix and coordinate with myBEA Production Support
- Train users
- Obtain final signoff
- Go Live

**Deliverables**
- Support Matrix
- Released Application

**Exit Criteria**
- Support Team and processes in place
- Live Application
Thank You

Yogish Pai

© Copyright 2007 – Yogish Pai
The author grants a non-exclusive license to everyone to publish this document in full or part by acknowledging the source. Any other usage is prohibited without the express permission of the authors.