I. Recap SOA Architecture and JKE’s “Open Account” Process

II. Business Process Modeling

III. Programming Models for Business Process Realization (Assembly)

IV. Business Driven Development (BDD)

V. Business Process Management – from end-to-end
I. Recap SOA Architecture and JKE’s Account Opening Process
Key Models for SOA – Enabling Greater Flexibility in Enterprise IT Architectures
Example for Business Process – „Open Account“ at JK Enterprises

- Virtual Company – a premier supplier to retail channel, small business channel, and corporate customers
- High-touch approach to our customers - now with customer centers around the world
- Customers of all types can interact with us in a way that suits their individual needs – corporate customers are true business partners
Use Case for “Open Account” Process

ACCOUNT OPEN PROCESS

1. Submit Application

Customer

1

Determine Eligibility

1

Eligibility System

1

Request Credit Report

1

Credit Reporting Service

1

Request More Documentation

1

Document Management System

1

Review Application

1

Credit Manager

1

Cleanse Application

1

Data Administration

1

Monitor Account Open Activity

1

Finance Manager
SOA Stack for “Open Account” Process

Consumers
- Sales Application Central Office
- Sales Application Regional Office

Business Process
Composition; choreography; business state machines
- Open Account
  - Account Activation
  - Account Verification
  - Determine Applicant Eligibility
  - Address Verification

Services
atomic and composite
- Account Inquiry
  - indirect exposure
- AR Setup
  - indirect exposure
- Account Setup
  - direct exposure
- Create Account
- Determine Eligibility
  - create from scratch
- Address Verification
  - third-party reuse

Service Components
- EJB
- Message Flow
- SCA
- EJB

Operational Systems
(Applications & Data)
- Customer (CICS 2.x)
- Billing (CICS 3.1)
- GL (SAP)

Other Systems
- Billing (CICS 3.1)
- GL (SAP)
- Address Verification
- Customer (CICS 2.x)
- AR Setup
- Account Inquiry
- Account Activation
- Determine Applicant Eligibility
- Address Verification
- Create Account
- Service Components
- Operational Systems
- Business Process
- Consumers

SOA Stack for “Open Account” Process

Diagram showing the layers of a service-oriented architecture (SOA) stack for an “Open Account” process, including consumers, business process, services, service components, and operational systems.
Reference Architecture and “Open Account” Process

Business Innovation & Optimization Services
Facilitates better decision-making with real-time business information

Interaction Services
Portal

Process Services
EJBs

Information Services
Federated Query

Partner Services
Community Manager

Business App Services
Build on a robust, scalable, and secure services environment

Access Services
CICS Access and Siebel Access

Infrastructure Services
Optimizes throughput, availability and performance

IT Service Management
Manage and secure services, applications, and infrastructure

IT Management Console

Integrated environment for design and creation of solution assets
Manage and secure services, applications, and resources

Scaleable, and secure services environment
Facilitates interactions with existing information and application assets

ESB
Facilitates communication between services

DB Access
DB Access
Overview of JKE’s IT Environment
II. Business Process Modelling
Business Models are Changing…

“On such a flat earth, the most important attribute you can have is creative imagination – the ability to be the first on your block to figure out how all these enabling tools can be put together in new and exciting ways to create products, communities, opportunities, and profits.

Thomas Friedman, The World is Flat

“ Service orientation does not begin with technology; it begins with the mind-set of thinking about your business and the world around you in terms of functional components. Becoming more functionally service oriented allows organizations to quickly and economically rearrange the components that make up a business.”

The Future of Business June 2007
Steve Mills, Senior Vice President and Group Executive
IBM Software Group
Providing Linkage Between Operational and IT Views
Reconciling Business Perspectives and IT Perspectives

**Business View**
- Business Analyst

**Process/Operation Model**
- Business Process Model
- Business Component
- Business Data
- Business Services

**IT View**
- IT Architect
- Developer

**Architecture/Execution Model**
- Process Choreography
- UML Class Model
- UML Data, XML, SQL Model
- Service Model, WS-BPEL

**Business Perspective**

**IT Perspective**

Enterprise / IT Architect
Where We Are Heading – Start

Case Study: Procure to Pay Process
Where We Are Heading – Goal

Case Study: Procure to Pay Process
Business Modeling – Methodology and RUP (Rational Unified Process)
The Process Model

Comprehensive Palette to model

Basic, Intermediate and Advanced modes show different levels of detail. Technology editing modes provide validation prior to

Objects with descriptive labels - Role label example

Color coded objects - color by Role

Swimlane view provides different views of the same

Metric Information available to view - Task

Process editor - Free-Form Layout
The Organization Model

- Defines the structure of organization units and locations
- Graphical organization trees allow for visibility into what the relationship between the organizations and the resources are in order to view hand-offs

Attributes of the organization units can be either user defined or adhere to the definitions of a specific runtime.
The Business Measures (Observation) Model

Icons are added automatically showing Triggers, Metrics, Timers, Counters and KPIs, etc. as they are added to the model.

Observation Model

Business Measures, KPIs and their Attributes
Developing the Process Model
Completing the “Account Open” Process Model

Design and Simulation of the “Account Open” Business Process Model

Collaborative Development of the “Account Open” Business Process Model
Business Process Modeling
Capture, Simulate, Analyze & Hand-off to Implementation

- Graphically Model Processes
  - Define: Goal, Scope, Perspective, Audience, Level-of-detail, Content
  - Introduce naming conventions for all process objects (costs, time, resources, decision points, actions, etc)
  - Agree on a maximum number of process levels (3-4) and number of activities per process diagram (15-20)

- Simulate and Analyze
  - Simulate execution with statistical analysis tools
  - Run "what if" scenarios to predict outcomes
  - Identify bottlenecks and workload imbalances
  - Isolate projects that will generate the greatest returns

- Hand off to Implementation
  - Export business and data models for use in IT deployment
  - Direct export of models to IT such as WS-BPEL for execution, XSD for data definitions, WSDL for services interfacing, UML for IT architect refinement

Architectural Benefit:
- Business analysts provide top-down approach to service and process design
- Enable coordination of process development across business stakeholders
- Creation of artifacts to support down-stream implementation
SOA-Based Business Assembly and Execution

- Business Process Choreography orchestrates services to form *deployable* processes:
  - Process model based on WS-BPEL
  - Choreography includes automated and human-based services, business rules, service invocations and control of flow aspects
  - Processes support transactions and compensation

- Service consumers can initiate as well as create SOA-based process solutions
- Mediation creation to transform/route service requests and responses
- Enables inter and intra-enterprise (B2B) service integration over the ESB framework

**Architectural Benefit:**
- Simplified, standards-based business process development
- Support for industry process and data models
- Directly invoke mediations for routing/transforming requests between services
III. Programming Models for Business Process Realization (Assembly)
BPEL (Business Process Execution Language)

- Business Process Execution Language (or BPEL), is a business process modeling language that is executable. It is serialized in XML and aims to enable programming in the large. The concepts of programming in the large and programming in the small distinguish between two aspects of writing the type of long-running asynchronous processes that one typically sees in business processes.

- Industry standard (OASIS) for web services choreography that allows the assembly of process definitions and (web) services
What is BPEL (Business Process Execution Language)

- Use the specification of a Business process
- Assemble the process and (web) services

As Web Services

Orchestrating Web services
BPEL Activities

- Interact with people, other processes, business partners and services
  - Receive Data
  - Send Data
  - Invoke Operation
  - Wait for 1 out of n Events
  - Human Interaction

- Process-internal tasks
  - No-Op
  - Wait
  - Update Variables
  - Execute Java code
  - Terminate/Exit
  - Signal an Error
  - Handle an Error

- Define the control flow
  - Parallel Tasks
  - Sequential Tasks
  - Synchronize Tasks
  - Switch/If
  - Loop

* BPEL 2.0
BPEL Business Process: Microflows and Macroflows

Microflows
One Transaction

Macroflows
Multiple Transactions
And compensation transactions

Transaction Boundary
BPEL Business Process: Compensation
Working with BPEL through Graphical Editor (Eclipse Environment)
Service Orchestration Assembly using Visual Tools for building Components

Defining human tasks

Assemble Service Components

Defining business rules

Creating Business Processes

Human Task Editor

Assembly Editor

Decision Table Editor

BPEL Editor
SCA (Service Component Architecture)

- **Service Component Architecture (SCA)** is a relatively new initiative advocated by major vendors of **Java EE** technology. Its proponents claim it is more natively suited for the delivery of applications that conform with the principles of service-oriented architecture. As such, SCA components are supposedly more **technologically agnostic**. (Wikipedia)

- SCA helps to define mediations in an ESB (Enterprise Service Bus)
  - **Modules** contain “wired” **Service Components**
  - Service Components use Business Objects/Service Data Objects for data
  - Modules have **port** to provide binding to other components
  - **Solutions** are a collections of Modules
SCA Programming Model

- Developer does need to work about low level binding details or to provide code to support this function
Using Modules for Encapsulation and Reuse

Module: Process Account Verify

- BPEL Process: Account Verify
- Store Customer
- Approve Credit
- Customer Credit

Module: Customer Credit

- Business Rule: Customer Credit

Module: Approve Credit Manually

- Human Task: Approve Credit

Module: Update Customer Database

- Interface Map Convert to DB2
- Store Customer

Library: CustLib

- BO: Customer
- IF: StoreCustomer

- BO: DB2Customer
- IF: DB2Adapter

DB2
Using Modules for Encapsulation and Reuse – Change of Storing order (now SAP instead of DB2)
IV: Business Driven Development (BDD)
Enablement of Business Integration

- Business and IT are no longer separate tracks
- Choreography of services
  - The sequencing, selection, and execution of operations
BDD Overview (including Legacy Applications)
Roles to be taken within a Development Process

- **Analyst**
  - Model, simulate, assemble, and monitor processes

- **Architect**
  - Visually model applications and data

- **Developer**
  - Construct, transform, integrate and generate code

- **Tester**
  - Design, create, and execute tests

- **Deployment Manager**
  - Provision, configure, tune and troubleshoot applications

- **Project Manager**
  - Follow a common process
  - Manage and measure projects and portfolios
  - Manage requirements

- **Executive**
  - Align investments with business objectives
  - Analyze and monitor project portfolios
  - Manage change and assets
  - Manage quality
  - Align investments with business objectives
  - Analyze and monitor project portfolios
Business Process Development Life Cycle Tools

Process Execution/Choreography

- Process Requirements
- Participate
- Manage Execution
- Existing Components
- Optimize
- Process Modeling
- J2EE/Java Services
- Composite
- App Devt/Assembly
- Monitor
- Analysis
- Optimize
Areas for Business Driven Development
BDD Details 1 – Governance & Capturing Requirements

- Prioritize proposed, existing and under-construction services based on business priority, risk and return
- Track service level financials
- Provide deep insight into SOA development
- Manage SOA project-team dependencies
- Forecast demand for service creation and updates
- Understand the cost of SOA creation, operations and maintenance
BDD Details 2 – Modeling Business Processes

- Discover and design key business processes
- Capture business data items exchanged between processes & tasks
- Assign tasks to roles that are responsible for their performance.
- Determine and allocate required resources
- Model the business organization & roles organizational units can play
- Determination of any other process/tasks (services) that must be provided by others
BDD Details 3 – Designing Services

- Architecture and design for service implementations
- Trace enterprise requirements to business processes and service implementations
- Define detailed system requirements or service implementations
- Modeling and architectural specification of services (using UML2)
- Discover and consume existing services
BDD Details 4 – Construct and Test the Services

- Build new services from scratch or enable existing applications for services
- Discover and consume existing services
- Test functionality
- Test performance
- Team development and life cycle integrations
BDD Details 5 – Implementing Business Processes

- Implement business processes designed by Business Analysts using BPEL
  - Plug in Services
  - Plug in Human Activities (Staff)
- Composition with services
- Deploy and test
V. Business Process Management – from end-to-end
Overview Service Management

What's happening with the infrastructure?
- Infrastructure and application discovery
- Server monitoring
- Storage monitoring
- Network monitoring
- Data monitoring
- Application monitoring
- Service monitoring

How does this relate to the business service?
- Dashboard
- Application dependency mapping
- Business service management
- Service level management

What actions do we take?
- System reconfiguration
- Data restore
- User identity provisioning
- System and application restart
- Infrastructure deployment
- Service mediation
Business Process Management

**BPM is:**
A discipline combining software capabilities and business expertise to accelerate process improvement and facilitate business innovation.

**BPM Solves:**
1. Process aren't documented
2. Bottlenecks prevent efficiency
3. Limited visibility into performance
4. Complex integration across multiple processes
5. Process change is cumbersome
6. KPIs not defined

**BPM Includes:**
- Integration
- Modeling
- Monitoring

- Software that Enables BPM
  - Forms
  - SOA
  - Workflow

- Models and Maps
- Process Knowledge

- Expertise that Delivers BPM
  - Policies
  - Rules
  - Methodology
Managing Your Business Processes

Clean hand-off from IT with Business Models, Metrics

Constructs for dynamic and adaptive business processes based on an integration platform

Feedback for continuous improvement

Real time management of business processes

Business Process Monitor
The End-to-end Account Opening Solution

People

Information

Process

Account Open Process

Receive Request

Credit Policy

Check Application

Create Account

Confirm Account

Receive Request

Account History Service

Customer

Account History

Customer Orders

Account Opening Service

Customer Application

Account Status

Customer Application

Account Info Service

Customer

Account Info

Customer Account

Internet/Intranet Portal
Building User Interaction Services

Developing and Deploying the “New Account” Application

Building Role-Specific Portlets and Dashboards
Infrastructure Architecture
Composite Application Management

Configuring Service Management Agents

Monitoring Account Opening Performance and Availability
Process Monitoring and Management

Monitoring Credit Risk

Building Components to Monitor Account Opening Duration
Logical Architecture for Business Activity Monitoring

1. Deploy Monitoring Model
2. Process Server (or other CBE sources)
   - Send events
3. Events sent to Monitor Server
4. Monitoring Model executes
5. Generated business situation events
6. Monitor database
7. Business situation events (from 2 & 5)
8. Take action
9. Monitor Business using Dashboards
10. Monitoring statistics imported to Modeler

Process:
- Process Server sends events to Monitor Server.
- Monitor Server generates business situation events.
- Monitor Modeler receives generated business situation events and deploys the monitoring model.
- The modeler executes the monitoring model.
- Business situation events are sent to the Monitor.
- The Monitor database replicates the data.
- Action Manager takes action based on the business situation events.
- Dashboards are used to monitor business.
Integrated Visibility of SOA Resources
A Representative Dashboard Solution
Deploying the Solution Architecture Implementation Topology for JK Enterprises
High Availability in the SOA World

- An application may exist on multiple servers in different locations
  - Applications need to be “availability” aware in case a service within the workflow is unavailable

- SOA applications impact service availability levels
  - SOA introduce new application dependencies, including externally provided services
  - Need to understand the end-to-end view

- Monitoring, management and reporting is required to achieve predictable availability in an SOA environment

- Plan for the unexpected
  - What are the non-functional requirements? What systems are you using? Distributed? Mainframe? Where are they located? How will they be accessed?
  - The more components in the transaction, the greater the risks for failure or human error
Techniques for High Availability and Scalability

1. Faster Machines
2. Replicated Machines
3. Specialized Machines
4. Segmented Workload
5. Request Batching
6. Data Aggregation
7. Connection Management
8. Caching
Closing Remark