Business Process Management and IT Architecture Design

The “T” case study

Dr. Jana Koehler
Olaf Zimmermann
IBM Zurich Research Laboratory
ZRL BIT at a Glance

- **IBM Zurich Research Lab (ZRL)**, Rüschlikon/ZH
- **Computer science** department:
  - Security/Cryptography and Systems Management
  - Business Process Management (BPM) and Distributed Systems

- **Business Integration Technologies (BIT) group**:
  - ~10 BPM and Service-Oriented Architecture (SOA) researchers
  - Scientific contributions to modeling languages, formal methods and verification, software engineering, software architecture
  - Key focus: compiler technology for process-oriented languages & IT architecture design methods
  - Main transfer channel: IBM WebSphere product
  - Participation in industry standards, customer consulting
Learning Objectives

- Understand and apply key concepts of Business Process Management
  - BPM Software Lifecycle and technology stack
  - Process modeling using the Business Process Modeling Notation (BPMN)

- Understand and apply selected software architecture and Service-Oriented Architecture (SOA) concepts
  - Software quality attributes
  - Architectural patterns for application development and integration
  - Architectural decisions
Agenda for Today

- **Case study:** Telecommunications Order Management
  - BPM fundamentals
  - Software architecture and SOA fundamentals

- **Your assignment**

- **Webpage** with all relevant information for this lecture:

- **Questions:**
  - koe@zurich.ibm.com (for BPM and BPMN topics)
  - olz@zurich.ibm.com (for software architecture and SOA topics)
“T” Case Study: Context & Business Problems

- **Wholesale subsidiary** of large telecommunications company T (former monopolist, deregulated)
  - Provides wire line and wireless telephony services to retailer subsidiary of T and to 150 other companies, called Virtual Service Providers (VSPs)
  - One physical telephony network, owned and operated by T

- **Strategic imperative** of T:
  - Drive down cost of operations by interacting with VSPs efficiently

- **Response:** single, partially automated *order management system*
  - VSPs are expected to use the order management processes of T to connect, configure, or disconnect telephony services for their end users
  - Multiple channels required, including the World-Wide Web (Internet)
System Context Diagram

VSP 1 (Browser)

VSP 2 (Other System via Web Service)

Internal Channel for T Staff

Order Management System

Customer Database

Telephony Network

Billing System
Functional Requirements

PSTN – Public Switched Telephone Network

VSP 1 (Browser)

VSP 2 (Other System via Web Service)

Internal Channel for T Staff

T


About 20 processing steps, taking up to 24 hours to complete. Steps include:

1. Address validation – complex and requiring several user interactions
2. Resource reservation, e.g. copper transmission path, telephone number

Customer Database

Telephony Network

Billing System
Business Processes

- A process is a naturally occurring or designed sequence of changes of properties of an object or system:

  “A business process describes key procedures within an organization.” (ETH EAI lecture definition)

- Operational aspect emphasized

- Activities are not independent of each other, require resources, result from the interaction of humans and technical systems with each other
  - They create physical/virtual objects and change the state of these objects
  - They serve a purpose (a goal that may remain implicit)

- Business Process Management (BPM) is a structured way to manage the life cycle of business processes including
  - Modeling, analysis and design, execution, monitoring, and optimization
Business Process Management Life Cycle: IBM Model

Capture functional requirements in a process modeling language

Build business applications from pre-built components/services

Compile processes to run directly on specific middleware

Manage software reuse and quality, apply patterns
From Enterprise Application Integration and Workflow Systems to BPM/SOA

## Integrated Technology Roadmap

<table>
<thead>
<tr>
<th>Business View</th>
<th>Methods</th>
<th>Applications</th>
<th>Architecture</th>
</tr>
</thead>
<tbody>
<tr>
<td>Silo</td>
<td>Structured Analysis &amp; Design</td>
<td>Modules</td>
<td>Monolithic Architecture</td>
</tr>
<tr>
<td>Integrated</td>
<td>Object Oriented Modeling</td>
<td>Objects</td>
<td>Layered Architecture</td>
</tr>
<tr>
<td>Componentized</td>
<td>Component Based Development</td>
<td>Components</td>
<td>Component Architecture</td>
</tr>
<tr>
<td>Services</td>
<td>Service Oriented (Application Neutral)</td>
<td>Services</td>
<td>Emerging SOA</td>
</tr>
<tr>
<td>Composite Services</td>
<td>Service Oriented</td>
<td>Process Integration via Services</td>
<td>SOA</td>
</tr>
<tr>
<td>Virtualized Services</td>
<td>Service Oriented</td>
<td>Process Integration via Services</td>
<td>Grid Enabled SOA</td>
</tr>
</tbody>
</table>
The BPM/SOA Portfolio

- All technology vendors work towards comprehensive and fully integrated offerings
- IBM WebSphere BPM
Business Process Modeling Notation (BPMN)

- First initiative to develop a notation: September 2000
- Version 1.0 of the BPMN Standard (OMG): May 2004
  - a set of graphical symbols, informal semantics, many examples, first vendor support
- Version 1.1 with minor bug fixing: August 2007
  - metamodel (unique exchange format), formally described semantics

_A first impression_

- **BPM for business and IT users**
- _graphical flow at business level_
- _textual refinements towards IT and runtime_
Activity-Oriented Modeling of Behaviors

- The most commonly used and best understood approach
- Well established in various disciplines
  - Software modeling: UML activity diagrams
  - “Business informatics”: Event-driven Process Chains (EPC) & workflows

- Major conceptual building blocks shared by all approaches:
  - Activity (task, function, action) and its refinement into a subprocess
  - Explicitly prescribed order of activities in a flow diagram
  - Special types of diagram nodes to capture flow branching behavior (gateways, rules, connectors, control nodes)
    - parallel + alternative branching, cycles

➤ We can only cover a small subset of the available modeling elements
Main Diagram Elements: Activities

- A distinguished step in the process
  - Takes a set of inputs and converts them into a set of outputs (either or both sets may be empty)
  - Encapsulates a specific business function

- Task (atomic process step, not further refined in the current model)
- Subprocess (refined with another BPMN diagram)
Main Diagram Elements: Events

- start event
- end event

- An example of a bad definition (from the BPMN 1.1. Spec):
  - … something that “happens” during the course of a business process
  - Can start, interrupt, or end the flow

- Better definition (from the ARIS Method 7.0, 4-98, 10/2006)
  - “By an event we understand the fact that an information object has taken on a business-relevant state which is controlling or influencing the further procedure of the business process.”
Main Diagram Elements: Gateways

*Exclusive Gateway (XOR – Decision/Merge)*

*Parallel Gateway (AND – Fork/Join)*

- Semantics described in terms of token flow (from Petri Net Theory)
- State of the business process = distribution of tokens (marking) in the diagram
Process Model Quality vs. Control-flow Errors

- **Deadlock**
  - Process is blocked forever, some activities never execute
  - Occurs for example when paths originating from an XOR gateway are joined by an AND gateway

- **Lack of Synchronization**
  - Results in multiple executions of activities (and risks of uncontrolled data access)
  - Caused for example when paths originating from an AND or OR gateway are joined by an XOR gateway

*Sound process*

= No deadlocks +

No lack of synchronization
The “T” Order Management Process I

Your Assignment
The "T" Order Management Process II
The “T” Order Management Process III
Important Non-Functional Requirements (NFRs)

1. The software system supporting the two order management processes must be accessible both over a private industry-sponsored network and the Internet. The detailed VSPs and backend system landscape changes over time.

2. Business volumes are approximately 20,000 “create” requests and 15,000 “move” requests per month.
   - Given up to 20 steps per process, and a peak hour load of 30% above average, this equates to a peak load of about 4,550 steps executed per hour (based on core business hours of ten hours per day, 20 days per month).

3. Initially, a process must be able to persist from first step to last for three hours; however, this time will be extended to up to 24 hours in the future.

4. VSPs are spread across a number of time zones, operating 23 hours per day and seven days per week.

5. Average response time targets vary by process step, typically 3-5 seconds; 95% of the user interactions need to complete execution in 5-8 seconds.

6. Domain-specific architecture design challenges include: 1. Address validation completeness and timeliness, 2. Releasing reserved resources (copper transmission path, telephone number) when a process instance fails or customer walks away.

7. Communication patterns and protocols must support multiple platforms.
Architecture Overview Diagram (First Design Iteration)

Solution: Service-Oriented Architecture (SOA) with Process & Service Layers

- **Process Layer**
  - "Create" Process
  - "Move" Process

- **Service Layer**
  - "Validation" Service
  - "Pricing" Service
  - "Transmission Reservation" Service

- **Systems**
  - Customer Database
  - Telephone Network
  - Billing System

VSP 1 (Browser) uses:
- Move
- Create

VSP 2 (Other System via Web Service) uses:
- Move
- Create

Internal Channel for T Staff uses:
- Move
- Create
Software Architecture Fundamentals

- **Responsibilities** of a *software architect* in custom application development:
  - Synthesizes technical solution from requirements (supported by methods)
  - Technically leads project and estimates development efforts
  - Coaches developers and other technical staff

- **Key concepts**: *Quality attributes*, *architectural patterns*, *architectural decisions*
  - Quality attributes are architecturally significant requirements
  - ... that drive selection of architectural patterns
  - ... which is captured in the form of architectural decisions

- **Foundations** date back to late 1990s (academia and industry)
  - Buschmann et al. “Patterns of Software Architecture” (POSA)
Software Quality Attributes (QAs)

- **How** does a system provide its functionality (not what it does)
  - Reliability, usability, efficiency (e.g., performance, scalability), maintainability, and portability areas defined in ISO/IEC specification 9126-2001

- Requirements in case study deal with QAs tacitly or explicitly:
  - QA “Accuracy”: orders must not be lost, transmission reservations must be undone
  - QA “Efficiency” (performance): response times specified by NFR 5
  - QA “Interoperability”: multiple platforms to be supported as per NFR 7 (which ones?)

- Practical challenges:
  - Many attributes, many conflicts between them; many attributes hard to quantify
  - Often under-specified (unknown) or over-specified (overly ambitious)
  - Often stated on inadequate level of abstraction, e.g., per system (not per function)
Architectural Patterns

- Proven, common solution to known, recurring architecture design problem:
  - Context captured by *intent* section
  - QAs discussed in *forces* section
  - A *problem statement* is given often in question form
  - There is a sketch of the *solution* (not a complete design!)

- Examples in case study:
  - “Layers” pattern from [POSA] structures the architecture overview diagram
  - See [http://www.vico.org/pages/PatronsDisseny/Pattern%20Layers/index.html](http://www.vico.org/pages/PatronsDisseny/Pattern%20Layers/index.html)

- Practical challenges:
  - Many eligible pattern languages
  - Link to requirements covered insufficiently
  - Transition to platform-specific design not addressed (or only in the form of examples)
Architectural Decisions

- Capture rationale justifying a design, in addition to design itself
- Example in case study:
  - “We selected the Layers pattern to make the order management architecture future proof and to be able to add VSP channels in a flexible manner”

- Practical challenges:
  - Retrospective decision capturing takes time and does not yield sufficient benefits
  - Relation to other architectural concepts and viewpoints (quality attributes, patterns) not understood well and not supported in methods and tools
Larger Example: Process Manager Pattern

“How do we route a message through multiple processing steps when the required steps may not be known at design-time and may not be sequential?”

“Use a central processing unit, a Process Manager, to maintain the state of the sequence and determine the next processing step based on intermediate results.” [EIP]

Process manager takes care of process instance creation and deletion, control flow routing, error handling in timeout situations, etc.

## Architectural Decision (AD) to Use Process Mgr. Pattern

<table>
<thead>
<tr>
<th>Subject Area</th>
<th>Process layer design</th>
<th>Topic</th>
<th>Workflow</th>
</tr>
</thead>
<tbody>
<tr>
<td>AD</td>
<td>Service Composition Paradigm</td>
<td>AD ID</td>
<td>2</td>
</tr>
<tr>
<td>Decision</td>
<td>We decided for the Process Manager pattern from the [EIP] book</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Issue or Problem</td>
<td>How to control the “Create PSTN” and “Move PSTN” processing?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Assumptions</td>
<td>Process model and requirements NFR 1 to NFR 7 are valid and stable</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Motivation</td>
<td>Process control is a mandatory requirement with high architectural significance</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Alternatives</td>
<td>Object-oriented programming, proprietary EAI tools</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Justification</td>
<td>The problem statement and the sketched solution of the pattern fit the requirements of the project (multiple processing steps, not sequential, correctness QA and timeout management NFR).</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Implications</td>
<td>Need to educate the team on process management and SOA</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Derived Requirements</td>
<td>Security requirements: to be able to correlate process instances and VSPs, users have to be identified in the order management system</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Related Decisions</td>
<td>Next, we have to decide on an integration style for link between processes and services and on implementation technologies for the selected patterns.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Summary of Status Quo in Case Study

- T has compiled a short list of vendors (professional services) and hosted an information day today. During the information day, T has presented:
  - Business strategy and system context
  - Order management requirements (functional and non-functional)
  - Design decisions already made including a partially specified business process and an architecture overview diagram

- You learned about:
  - Business process management and modeling with BPMN
  - Architectural decision making based on quality attributes and patterns
The Assignment

- T issues a “Request for Proposal (RFP)” and expects a response within a week:
  - You are taking business analyst and software architect role in one of the professional services companies on the short list

- Your RFP response should contain:
  1. Refined business process model (mandatory)
  2. Second iteration of architecture overview diagram (mandatory)
  3. Technical questions about methods, languages, and tools (optional)
  4. Value and benefits argumentation (mandatory)
  5. Cost estimate (optional – handle with care)
Task 1: Model the “Create New Service” Subprocess

- If a customer requests a new PSTN service, we first need to validate if the service is really offered at the requested time. A service offer may be restricted to a specific advertising period or it may not be available when the customer wants to have it provided. If the service request is valid, we continue to perform three checks, otherwise we close the request.

- The three checks that are independent of each other, but must all three be performed
  - Eligibility of the customer for the service (e.g. is the customer eligible for an age-specific service or a service that can only be used in a bundle with other services?)
  - Credit history of the customer (is he paying well, what customer status does he have and does this imply constraints on the requested service?)
  - Technical prerequisites (is the service available at the requested location?)

- The result of these checks is that the service is either possible or not. In the “no” case, we want to review other possibilities with the customer and iterate the subprocess again (i.e., begin with validating another service alternative, check the 3 criteria, etc.)
Task 2: Refine Architecture with Integration Patterns

- Decide for an *integration style* to connect the activities in the two order management processes (“Create PSTN Service”, “Move PSTN Service”) to the atomic services (“validation” and “transmission reservation”) and justify your decision.
  - File Transfer pattern?
  - Shared Database pattern?
  - Remote Procedure Invocation pattern?
  - Messaging pattern?

- Input
  - System context diagram, architecture overview diagram (iteration 1/2), NFRs

- Output:
  - Documentation of architectural decision and updated architecture overview diagram
Task 3: Method Aspects and Technical Questions (Optional)

- Which requirements engineering method do you propose, and why?
  - Which other ones did you consider, and why did you decide against them?

- Are you content with the introduction of a process manager into the architecture (and why)?
  - Which service composition paradigm to use – workflow, other?
  - Which programming language – Java, BPEL, other?
  - Which workflow or other engine (open source, commercial)?

- Which remote communication protocol would you use to support the chosen integration style, making the services available to the processes? Name the relevant technology standard.
Task 4 and 5

- Task 4 – Value and benefits argumentation (mandatory):
  - What are the benefits of your solution?
  - Why should you get the deal?

- Task 5 – Cost estimate (optional – handle with care):
  - Can you give a first indication how much it will cost to build your solution?
  - How long will it take?
## Additional Project Information

### Cultural environment
- Developed country, multicultural society. High amount of immigration/emigration.
- Telecommunications company positions itself as a thought leader and early adopter of emerging technology in its geography, ready to operate in risk-reward sharing mode.
- Java skills available, Business Process Execution Language (BPEL) education planned.

### Team
- About 100 people already on board, who worked on previous order management system (in production, but not process-oriented): ~5 project managers, ~5 architects, ~20 business requirement analysts, ~50 developers, ~20 testers.

### Existing assets
- 20 backend systems to be integrated in total, Java adapters available.
- 100,000 lines of legacy code (Java and other), e.g., validation and reservation EJBs.
- Technology platforms to be supported: Microsoft .NET clients, JEE clients and servers, SQL and relational database, proprietary telephony hardware, commercial billing package.
Background Reading

- The ORYX Editor at Hasso Plattner Institute (any other tool or way of drawing is fine, too)
  - [http://bpt.hpi.uni-potsdam.de/Oryx](http://bpt.hpi.uni-potsdam.de/Oryx)
- The four integration styles listed in part 2 of your assignment are captured in pattern form in “Enterprise Integration Patterns” by Hohpe and Woolf
  - See [http://www.eaipatterns.com/IntegrationStylesIntro.html](http://www.eaipatterns.com/IntegrationStylesIntro.html)
- For layering details, have a look at Fowler’s book “Patterns of Enterprise Application Architecture” (PoEAA)
- Distributed computing and Web services concepts such as WSDL, SOAP, and BPEL are featured in many books, including:
  - Lecture view: Alonzo et al. “Web Services”
  - Industry project view: Zimmermann et al. “Perspectives on Web Services”