From Conceptual to Executable
BPMN Process Models
A Step-by-Step Method

Marcello La Rosa & Marlon Dumas
Queensland University of Technology – University of Tartu
m.larosa@qut.edu.au, marlon.dumas@ut.ee
What’s this tutorial about?

Conceptual process model

Executable process model

* “And Then A Miracle Occurs”
Who’s this tutorial for?

1. BPM practitioners seeking to bridge business – IT
2. BPM instructors / teachers
3. Business process modeling and automation researchers

Basic knowledge of BPMN assumed
The BPM lifecycle
The BPM Lifecycle (revisited)
The well-known gap...
The result: two sides of the story

Conceptual “to-be” process models
- are made by domain experts
- provide a basis for communication amongst relevant stakeholders
- must be understandable
- must be intuitive and may leave room for interpretation
- contain purely a relevant set of process information

Executable process models
- are made by IT experts
- provide input to a process enactment system - BPMS
- must be machine readable
- must be unambiguous and should not contain any uncertainties
- contain further details that are only relevant to implementation

“to-be executed” process model
Bridging the gap: one task at a time

1. Identify the automation boundaries
2. Review manual tasks
3. Complete the process model
4. Adjust task granularity
5. Specify execution properties

Part I

Part II

Adapted from teaching material of Remco Dijkman, TU/e.
Our running example

Seller
Our running example
1. Identify the automation boundaries

**Principle:** not all processes can be automated.

-> Start by identifying each task's type:

1. Automated tasks
2. User tasks
3. Manual tasks
In BPMN: specify task markers

Automated tasks

User task

Manual task
In our example...

Diagram showing the flow of a process with automated, user, and manual sections.
2. Review manual tasks

**Principle:** if it can’t be seen by the BPMS, it doesn’t exist.

-> Find ways to support manual tasks via IT:
   - via user task
   - via automated task

-> Isolate them and automate the rest
Alternative: isolate manual tasks
Alternative: isolate manual tasks

Segment 1

Segment 2

Segment 3
Quiz: let’s consider this process fragment

Prescription fulfillment process:
• Once the prescription passes the insurance check, it is assigned to a technician who collects the drugs from the shelves and puts them in a bag with the prescription stapled to it.
• After that, the bag is passed to the pharmacist who double-checks that the prescription has been filled correctly.
• After this quality check, the pharmacist seals the bag and puts it in the pick-up area.
• When a customer arrives to pick up their prescription, a technician retrieves the prescription and asks the customer for their payment.

Assume the pharmacy system automates this process. Identify the type of each task and link manual tasks to the system.
Possible solution
BPMN elements irrelevant for execution

- Physical data objects
- Messages bering physical data objects
- Data stores (both physical and electronic)
- Pools & lanes
- Text annotations

Remove or neglect, depending on BPMS
3. Complete the process model

**Principle**: exceptions are the rule.
-> Add exception handlers

![Flight information](image)

*It happened for real!*

**Principle**: no data = no decisions, no tasks handover.
-> Specify all **electronic** business objects
In our example...
In our example...
4. Adjust task granularity

**Principle**: BPMSs add value if they coordinate handovers of work between resources.

- Aggregate any two consecutive tasks assigned to the same resource
- Refine tasks that are too coarse-grained
Look around

Candidate tasks for aggregation may not necessarily be consecutive due to a sub-optimal order of tasks in the conceptual model.
An exception to the rule
Our example...

After Step 4
End of Part I
Part II: the “last mile”

Marcello La Rosa & Marlon Dumas

Queensland University of Technology, University of Tartu
m.larosa@qut.edu.au, marlon.dumas@ut.ee
Bridging the gap: one task at a time

1. Identify the automation boundaries
2. Review manual tasks
3. Complete the process model
4. Adjust task granularity
5. Specify execution properties
5. Specify execution properties

- Process variables, messages, signals, errors
- Task and event variables and their mappings to process variables
- Service details
- Code snippets
- Participant assignment rules and user interface structure
- Task, event and sequence flow flow expressions
- BPMS-specific: work queues, forms, connectors...
Let’s take a step back: BPMSs
Business Process Management System
Process modeling tool

• To create and modify executable process models (by specifying execution properties)
• To store and retrieve automation solutions from a process model repository
• May import from conceptual process modeling tools
Example process modeling tools
Execution Engine

- Instantiates executable process models (also called “cases”)
- Orchestrates distribution of work items to process participants and software services in order to execute a business process from start to end
- Logs execution data
Worklist Handler

• Imagine it as an “inbox”
• Offers work items to process participants and allows participants to commit to these work items
• Handles participants’ work queues and work item priorities
• May provide social network capabilities
Example worklist handlers

Bonita Soft Bonita Open Solution
Administration & Monitoring Tools

- To manage automation solutions
- To configure access to system components
- To monitor participants availability and performance of process cases
Example monitoring & administration tools

IBM BPM Process Portal
External Services

• Expose a service interface with which the engine can interact
• The engine provides the invoked service with the necessary data it will need to perform the activity for a specific case
• Examples: rules engine, email or Twitter notification, DB connector, CRM connector...
Example external services

Bosch Visual Rules editor
Evolution of the BPMS Landscape
## BPMS Landscape

<table>
<thead>
<tr>
<th>Big vendors</th>
<th>Other closed-source</th>
<th>Commercial open-source</th>
<th>Community open-source</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM BPM</td>
<td>Appian BPMS</td>
<td>Bonita Open Solution</td>
<td>Shark</td>
</tr>
<tr>
<td>Oracle BPMS</td>
<td>BizAgi BPM Suite</td>
<td>Camunda Fox</td>
<td>YAWL</td>
</tr>
<tr>
<td>Microsoft BizTalk, Wf</td>
<td>Bosch inubit Suite</td>
<td>Intalio</td>
<td>BPM</td>
</tr>
<tr>
<td>SAP NetWeaver BPM</td>
<td>OpenTex tBPM</td>
<td>JBoss jBPM</td>
<td></td>
</tr>
<tr>
<td>Software AG webMethods</td>
<td>Perceptive BPMONE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pagaystems PegaRULES</td>
<td>Progress Savvion</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>TIBCO ActiveMatrix BPM</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The above list includes both closed-source and open-source solutions under the categories of Big vendors, Other closed-source, Commercial open-source, and Community open-source.
BPMS classification according to BPMN support

1. **Pure BPMN:** (re)designed from the ground up to follow the spec to the letter
   - IBM BPM, Appian BPMS, Camunda Fox

2. **Adapted BPMN:** use a BPMN skin but rely on internal representation – predate BPMN
   - Bonita Open Solution, **BizAgi BPM Suite**

3. **Non BPMN:** proprietary language and semantics
   - Bosch inubit Suite, BPMOne, YAWL
Let’s take a look at a concrete BPMS
Cheat sheet

1. Control flow  >  specify sequence flow expressions…
2. Data flow  >  specify data types and data mappings
3. Resources  >  specify participants assignment rules, service details…
Want to know more?

- Long-awaited BPM textbook
- Covers the entire BPM lifecycle
- Running examples & questions
- 100+ exercises with and without solutions
- Based on BPMN
- Available as Springer eBook, Apple iBook, Amazon...
- Chinese translation coming soon
http://fundamentals-of-bpm.org

- Lecture notes
- A/V recordings
- Quizzes
- Tutorials
- and more...

FUNDAMENTALS OF BPM

SUPPLEMENTARY MATERIAL

Chapter 1: Introduction to BPM
  • Online quiz

Chapter 2: Process Identification
  • Online quiz

Chapter 3: Essential Process Modeling
  • Online quiz

Chapter 4: Advanced Process Modeling
  • Online quiz

Chapter 5: Process Discovery
  • Online quiz

Chapter 6: Qualitative Process Analysis
  • Online quiz
That’s it!

Marcello La Rosa & Marlon Dumas
Queensland University of Technology, University of Tartu
m.larosa@qut.edu.au, marlon.dumas@ut.ee