Outline

- Positioning BPM|one
- Supplemental Assignment (deadline 04-03-2013)
- BPM|one design (≈ Protos)
- Analysis in the context of BPM|one
  - Verification using ProM/Woflan
  - Simulation using BPM|one
  - Process mining using BPM|one
- BPM|one control
- Flexibility
  - Taxonomy
  - Dynamic change bug
Role BPM|one

- Support of whole life-cycle.
- Good pattern support.
- Flexibility through case handling.
- Innovative features such as process mining, simulation, etc.
- In assignment role is limited to modeling and simulation.
- ProM is used for verification and other forms of analysis.
- YAWL is used for enactment.
Protos versus BPM|one


Convention and Tools Guidelines for BPMS Assignment 2013

Editors: J. van Buik (j.c.a.m.buik@tue.nl), D.Schunselaar (d.m.schunselaar@tue.nl)
Last updated: 2013-01-25

Table of Contents

I. Protos
   1. Installing Protos and additional plugins
   2. Create WfMan report (scr)
   3. Performing Simulation
   4. Generate process report (for Assignment)

II. Convention for Modeling Process in Protos
   1. Terminology
   2. Detailed Description
   3. Specific Assignment Issues

III. ProM-6
   1. Installing ProM-6
   2. Using Prom to verify Soundness

IV. YAWL
   1. Installing YAWL
   2. Exporting Organizational Data in YAWL

V. BPM One (optional)
   1. Creating a process model
   2. Creating Backup file
   3. Loading Backup file
   4. Check in process model files
   5. Create Reports
   6. Perform Simulations
   7. Create scratch files
Supplementary Assignment

Goal: understanding the full scope of BPM|one (not just the Protos modeler).

Form: Answer selected questions learned by interacting with system.

Login information via e-mail (check).

Deadline: 4-3-2013!

Contact Dennis/Joos in case of problems.
BPM lifecycle

- Process (re)design
- Process adjustment
- Process execution
- Process monitoring
- Process analysis
- Process measurement

Thanks to Pallas Athena for sharing some of the figures used in this lecture!
Perceptive Process

Optimized processes support your business with the right information in the right place at the right time. Perceptive Process adds structure and agility to process improvement with analysis and design tools, readily adaptable execution capabilities, and world-class workflow and case management in an integrated suite of products. Perceptive fully supports business process management (BPM) through process discovery, design, execution, and improvement.

Proof Beats Speculation

Understand your process flows, and where they’re breaking down, based on actual data — not hunches.

Discover Perceptive Process Mining

CASE STUDY

Province of South Holland

When the Province of South Holland faced budget cuts, they turned to Perceptive Software for a solution that could automate the grants process and help them comply with strict regulations.

Lexmark Acquires Pallas Athena

Lexmark has announced the acquisition of Pallas Athena, a leading provider of BPM, DMS, and process mining software, which will add capabilities to its enterprise business process management (EBPM) portfolio.

Lexmark's acquisition of Pallas Athena strengthens its end-to-end BPM solutions, providing customers with a comprehensive suite of tools to improve efficiency, reduce costs, and gain insights into business processes.
Protos + FLOWer = BPM|one

BPM|one®

Process mining  Process analysis  Process modeling  Process design

Process activation  Case design  Integration  Adaptation  Deploy
Case type model; a model that is opened and edited with the Case Type Editor.

Process model; a model that is opened and edited with the Process Designer.
More than 80% of Dutch municipalities use Protos / BPM|one.

More than 2,000 organizations and 1,000,000 users use Protos / FLOWer / BPM|one.
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trigger</td>
<td>An Activity or a Process starts with a telephone notification.</td>
</tr>
<tr>
<td>Activity</td>
<td>An Activity or Process starts with a telephone notification.</td>
</tr>
<tr>
<td>Sub-Process</td>
<td>A decision in a Process is only taken once a request has been received by post.</td>
</tr>
<tr>
<td>Buffer</td>
<td>Activities or Processes start after information has been received by EDI (Electronic Data Interchange).</td>
</tr>
<tr>
<td>Status</td>
<td>A signal to start an Activity.</td>
</tr>
<tr>
<td>Document</td>
<td>Activities or Processes start after a client has visited the reception.</td>
</tr>
<tr>
<td>Folder</td>
<td>Activities or Processes start after a client has visited the reception.</td>
</tr>
<tr>
<td>Data Element</td>
<td>Activities or Processes start after a client has visited the reception.</td>
</tr>
<tr>
<td>Data Group</td>
<td>Activities or Processes start after a client has visited the reception.</td>
</tr>
<tr>
<td>Application</td>
<td>Activities or Processes start after a client has visited the reception.</td>
</tr>
<tr>
<td>Role</td>
<td>Activities or Processes start after a client has visited the reception.</td>
</tr>
<tr>
<td>Role Group</td>
<td>Activities or Processes start after a client has visited the reception.</td>
</tr>
<tr>
<td>Team</td>
<td>Activities or Processes start after a client has visited the reception.</td>
</tr>
<tr>
<td>Team</td>
<td>Activities or Processes start after a client has visited the reception.</td>
</tr>
</tbody>
</table>

**Symbols**

- **Role**: A symbol representing a role within a process or activity.
- **Group**: A symbol representing a group within a process or activity.
- **Data Perspective**: A symbol representing the data perspective in a process or activity.
- **Hierarchical**: A symbol representing the hierarchy in a process or activity.
- **Place/Condition**: A symbol representing the place or condition in a process or activity.
- **Trigger**: A symbol representing the trigger event in a process or activity.
- **Task**: A symbol representing a task within a process or activity.
Start and end of a process
Activity (=task)

- Data
- Analysis
- Role
- Group
- 2-4 eyes principle
Splits and joins
Relating data to activities

colors: used-by contains
Relating roles to activities
Triggers

- Triggers are mandatory for assignment.
- For simulation it is better to replace them by dummy activities to model the additional delays.
- No resource triggers!
Analysis

• Verification (via ProM)
• Simulation
• Process mining
Note exports RTF (for making report), HTML, and Woflan. Use Woflan export for verification.
Extract “scr file”

save scr file and open with ProM (or Woflan)
Export in Protos

Protos D:\courses\BPMS-2012\etc\Protos models\handling-orders.pal - [Main Process]

File | Edit | Alignment | View | Analysis | Tools | Window | Help
--- | --- | --- | --- | --- | --- | --- | ---
New... | Ctrl+N |  |  |  |  |  |  |
Open... | Ctrl+O |  |  |  |  |  |  |
Save | Ctrl+S |  |  |  |  |  |  |
Save As... | Ctrl+L |  |  |  |  |  |  |
Process Model Properties... |  |  |  |  |  |  |  |
Report Options |  |  |  |  |  |  |  |
Report |  |  |  |  |  |  |  |
Import... |  |  |  |  |  |  |  |
Send to... |  |  |  |  |  |  |  |
Return review model... |  |  |  |  |  |  |  |
Printer Options... | Ctrl+I |  |  |  |  |  |  |
Print | Ctrl+P |  |  |  |  |  |  |
1 handling-orders.pal |  |  |  |  |  |  |  |
2 compl.pal |  |  |  |  |  |  |  |
3 D:\courses\BPMS-2012\model1-seq.pal |  |  |  |  |  |  |  |
4 insuranceNOK.pal |  |  |  |  |  |  |  |
5 D:\courses\BPMS-2012\insurance.pal |  |  |  |  |  |  |  |
6 insurance-test.pal |  |  |  |  |  |  |  |
7 D:\courses\BPMS-2012\insurance.pal |  |  |  |  |  |  |  |
8 D:\courses\BPMS-2012\insurance.pal |  |  |  |  |  |  |  |
Exit |  |  |  |  |  |  |  |
Rich Text Format (RTF)... |  |  |  |  |  |  |  |
Hypertext Markup Language (HTML)... |  |  |  |  |  |  |  |
CRUD... |  |  |  |  |  |  |  |
RACI... |  |  |  |  |  |  |  |
XLS/HTML... |  |  |  |  |  |  |  |
ExSpect... |  |  |  |  |  |  |  |
Woflan... |  |  |  |  |  |  |  |
XML... |  |  |  |  |  |  |  |

Update

Ship Goods

Receive payment
Download ProM 6.2 from http://www.promtools.org/prom6/

Downloads

Windows installer
EXE, 25.045 KB

tar.gz archive (all platforms)
TAR GZ, 21.925 KB

Example log files
ZIP, 270 KB

ProM getting started
PDF, 11.417 KB

ProM tutorial
PDF, 3.981 KB

ProM exercises
PDF, 3.171 KB

ProM package documentation
HTML

Nightly builds
HTML

Useful links:
- processmining.org
- sourceforge project page

Hosting provided by:
SOURCEFORGE.NET®

We are using YourKit's Java Profiler. YourKit is kindly supporting open source projects with its full-featured Java Profiler. YourKit, LLC is creator of innovative and intelligent tools for profiling Java and .NET applications. Take a look at YourKit's leading software products: YourKit Java Profiler and YourKit .NET Profiler.

ProM is a project by the Process Mining Group, Eindhoven Technical University. © 2010, page designed by Eric Verbeek and Christian W. Günther.

ProM 6.1
Processes are an integral part of today's world, driving services and internal functionalities in...
Let us first consider a correct workflow model
Import

import scr file
(Cosa Petri Net file)
Resulting Petri net object

- View Petri net
- Analyze net
- Delete net

Petri net object resulting from import
Analyze with Woflan

- select Woflan plugin
- run plugin on Petri net object
Woflan Diagnosis on Net "Main_Process (imported from Main_Process.scr) [Reduced]"

The net is a sound workflow net.
Analyze structural properties
Select structural properties

Structural Analysis Plug-in

Select Analysis Plugins
To select a plugin, click the name of the plugin on the list below. To deselect it, click again the name of the plugin. If a plugin depends on other plugin(s), the plugins will automatically be selected.

- S-Components
- Relaxed soundness info (using LoLA for Windows)
- T-Components
- Siphons
- T-P Handles
- Traps
- Free-choice info
- Non-free-choice clusters
- Transition Invariants
- P-T Handles
- Extended-free-choice info
- Non-extended-free-choice clusters
- Non-relaxed sound transitions (using LoLA for Windows)
- Place Invariants

[Cancel] [Continue]
A place invariant
A transition invariant
A TP handle
Analyze behavioral properties
Select behavioral properties
Open net is not live, but bounded
One dead marking
No dead transitions
Reachability graph
Introducing an error
Error in more detail
Woflan diagnostics

Woflan Diagnosis on Net "Main_Process (imported from Main_Process.scr) [Reduced]"

The net is not a sound workflow net.

Soundness requirements

Option to complete
Whatever happens, an instance can always mark the sink place

Proper completion
On completion, only the sink place is marked, and it is marked only once

No dead tasks
No transition is dead

The short-circuited net is bounded, contains no dead transitions, but is not live. As a result, completion is not always possible.

The following transitions are not live in the short-circuited net:

1. receive_payment
2. update
3. archive
4. _x_
5. replenish
6. reminder
7. ship_goods
8. check_availability
9. register
10. send_bill

The following diagnostic information assumes that there exists a part of the state space from which completion is still possible. Clearly, to avoid losing the option to complete, behavior should be restricted to this part. Thus, any transition leaving the part should be disabled.
Not sound: Not always option to complete

The net is not a sound workflow net.

Soundness requirements

Option to complete
Whatever happens, an instance can always mark the sink place

Proper completion
On completion, only the sink place is marked, and it is marked only once

No dead tasks
No transition is dead
Woflan Diagnosis of net Main_Process (imported from Main_Process.scr) [Reduced]

1. receive_payment
2. update
3. archive
4. _t_
5. replenish
6. reminder
7. ship_goods
8. check_availability
9. register
10. send_bill

The following diagnostic information assumes that there exists a part of the net that is not covered, and any uncovered place cannot be related to any aspect of the core functionality.

Disabling the following transitions at the following (reachable) markings elimi- nates the non-live places:

1. Transition reminder at marking [c6,c7].

The following diagnostic information presents places that are not covered, and any uncovered place cannot be related to any aspect of the core functionality.

1. c3
2. check_availability completed
3. c1
4. c7

The following diagnostic information presents place-transition pairs for which multiple disjoint paths exist from the place to the transition. Such pairs relate strongly to non-live behavior, and may indicate a root cause for the non-liveness.

1. Place c6, transition reminder
2. Place c7, transition archive
3. Place c7, transition reminder
Another example
Petri net translation
Process definition is sound
Example with error
Petri net translation
Process definition is not sound

Woflan Diagnosis on Net "complaints (imported from complaints-bad.scr) [Reduced]"

The net is not a sound workflow net.

Soundness requirements

Option to complete
Whatever happens, an instance can always mark the sink place

Proper completion
On completion, only the sink place is marked, and it is marked only once

No dead tasks
No transition is dead

The short-circuited net is unbounded. As a result, completion cannot be proper.

The following places are unbounded in the short-circuited net:

1. c3
2. c5
3. send_form started

The following diagnostic information assumes that there exists a bounded safe haven (a bounded strongly connected component which includes the initial marking). One possible safe haven is:
Another example
Petri net translation
Process definition is sound

Woflan Diagnosis on Net "travel (imported from travel-good.scr) [Reduced]"

The net is a sound workflow net.
Example with two errors

XOR

AND
Petri net translation
P-T handles
Process definition is not sound

Woflan Diagnosis of net travel (imported from travel.scr) [Reduced]

Woflan Diagnosis on Net "travel (imported from travel.scr) [Reduced]"

The net is not a sound workflow net.

Soundness requirements

Option to complete
   Whatever happens, an instance can always mark the sink place
Proper completion
   On completion, only the sink place is marked, and it is marked only once
No dead tasks
   No transition is dead

The short-circuited net is bounded, but contains dead transitions. As a result, the net contains dead tasks.

1. send
2. cancel_or_join
3. AND_join
4. start_of_trip
Alternative: Woflan
XOR
AND
AND
XOR
AND
Other Petri net tools

Make sure to view net first to have layout.
WoPeD (http://www.woped.org/)
Simulation

**Process Designer sim**:

- **Process**:
  - Start
  - Status
  - Activity A
  - Status1
  - Activity B
  - Status2
  - Check
  - Status3
  - End

**XOR**

**Process Model properties**:

- **Simulation** tab:
  - Number of sub-runs: 10
  - Length of sub-run: 100
  - Number of start-runs: 0

**Arrival time of cases**

- Stochastic function: Negative exponential
- Mean: 2.5
Activity properties Activity A

- **Name**: Activity A
- **Type**: Basic

Executor
- **Role A**
- **Multiple**: False
- **Sub-Process**: False
- **Start**: False
- **Process Model**: False

Responsible

Processing time
- **Stochastic function**: Negative exponential
- **Mean**: 2

Activity properties Activity B

- **Name**: Activity B
- **Type**: Basic

Executor
- **Role B**
- **Multiple**: False
- **Sub-Process**: False
- **Start**: False
- **Process Model**: False

Responsible

Processing time
- **Stochastic function**: Negative exponential
- **Mean**: 2

Role properties Role A

- **Number of persons**: 1
- **Cost**: 0

Role properties Role B

- **Number of persons**: 2
- **Cost**: 0
0 min. on average 1 resource

2 min. on average 2 resources

2 min. on average

0 min. on average

0 min. on average

50%

50%

50%
Start simulation

switch on simulation
animation of simulation run
spreadsheet with results
Utilization A [0.72, 0.85]  exact value = 0.8
Utilization B [0.82, 0.93]  exact value = 0.8
Work time A [1.88, 2.27]  
Work time B [1.86, 2.17]
confidence intervals are too broad
Flow time [13.6, 21.6]
## Longer runs

### Roles Utilisation rate

<table>
<thead>
<tr>
<th>Role</th>
<th>Mean</th>
<th>Lower 90%</th>
<th>Upper 90%</th>
<th>Lower 99%</th>
<th>Upper 99%</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;&lt;No role&gt;&gt;</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Role A</td>
<td>0.778074196</td>
<td>0.746140272</td>
<td>0.810008121</td>
<td>0.721360943</td>
<td>0.83478745</td>
</tr>
<tr>
<td>Role B</td>
<td>0.778396761</td>
<td>0.754163502</td>
<td>0.80263002</td>
<td>0.735359552</td>
<td>0.821433969</td>
</tr>
</tbody>
</table>

### Activities

#### Queue time

<table>
<thead>
<tr>
<th>Activity</th>
<th>Mean</th>
<th>Lower 90%</th>
<th>Upper 90%</th>
<th>Lower 99%</th>
<th>Upper 99%</th>
</tr>
</thead>
<tbody>
<tr>
<td>start</td>
<td>5,620665122</td>
<td>4,743413115</td>
<td>6,497917129</td>
<td>4,062703907</td>
<td>7,178626337</td>
</tr>
<tr>
<td>Activity A</td>
<td>1,994292214</td>
<td>1,927547357</td>
<td>2,061037072</td>
<td>1,875756265</td>
<td>2,112828164</td>
</tr>
<tr>
<td>Activity B</td>
<td>2,19422361</td>
<td>2,329970603</td>
<td>3,108874118</td>
<td>2,027773065</td>
<td>3,411071656</td>
</tr>
<tr>
<td>check</td>
<td>2,005389544</td>
<td>1,96981813</td>
<td>2,040960958</td>
<td>1,942216268</td>
<td>2,06856282</td>
</tr>
<tr>
<td>end</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

#### Work time

<table>
<thead>
<tr>
<th>Activity</th>
<th>Mean</th>
<th>Lower 90%</th>
<th>Upper 90%</th>
<th>Lower 99%</th>
<th>Upper 99%</th>
</tr>
</thead>
<tbody>
<tr>
<td>start</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Activity A</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Activity B</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>check</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>end</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

### Status

#### Wait time

<table>
<thead>
<tr>
<th>Status</th>
<th>Mean</th>
<th>Lower 90%</th>
<th>Upper 90%</th>
<th>Lower 99%</th>
<th>Upper 99%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Status</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Status1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Status2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Status3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

#### Wait + queue time

<table>
<thead>
<tr>
<th>Status</th>
<th>Mean</th>
<th>Lower 90%</th>
<th>Upper 90%</th>
<th>Lower 99%</th>
<th>Upper 99%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Status</td>
<td>5,620665122</td>
<td>4,743413115</td>
<td>6,497917129</td>
<td>4,062703907</td>
<td>7,178626337</td>
</tr>
<tr>
<td>Status1</td>
<td>2,719422361</td>
<td>2,329970603</td>
<td>3,108874118</td>
<td>2,027773065</td>
<td>3,411071656</td>
</tr>
<tr>
<td>Status2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Status3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

### Total

#### Lead time

<table>
<thead>
<tr>
<th>Mean</th>
<th>Lower 90%</th>
<th>Upper 90%</th>
<th>Lower 99%</th>
<th>Upper 99%</th>
</tr>
</thead>
<tbody>
<tr>
<td>17.04434467</td>
<td>15.83490162</td>
<td>18.25378771</td>
<td>14.89642669</td>
<td>19.19226265</td>
</tr>
</tbody>
</table>

#### Work time

<table>
<thead>
<tr>
<th>Mean</th>
<th>Lower 90%</th>
<th>Upper 90%</th>
<th>Lower 99%</th>
<th>Upper 99%</th>
</tr>
</thead>
<tbody>
<tr>
<td>5,989399959</td>
<td>5,838558433</td>
<td>6,140241484</td>
<td>5,721512004</td>
<td>6,257287913</td>
</tr>
</tbody>
</table>
Only 10% of the checks trigger rework

Significant drop in utilization of Role B resources and flow time.

exact value = 0.44
Process Mining

CSV file
### Column selection

<table>
<thead>
<tr>
<th>Column</th>
<th>Sample Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td></td>
</tr>
<tr>
<td>Case ID</td>
<td>5351, 5382, 5383, 5400, 5401, 5577, 5658, 5664, 5685, 5717</td>
</tr>
<tr>
<td>B</td>
<td></td>
</tr>
<tr>
<td>Activity</td>
<td>Aanvraag registratie, Eigen onderzoek, Rapportage &amp; beschikking, Toetsen en beslissen, Verzending\dossiervorming, Wachten terugmelding zorgaanb, Administratieve verwerking, Slotfase, Onderzoek CI, Retour</td>
</tr>
<tr>
<td>C</td>
<td></td>
</tr>
<tr>
<td>Timestamp</td>
<td>8-1-2007 0:00, 15-2-2007 0:00, 30-3-2007 0:00, 4-4-2007 0:00, 10-4-2007 0:00, 6-2-2008 12:00, 6-2-2008 12:05, 11-1-2007 0:00, 19-4-2007 0:00</td>
</tr>
</tbody>
</table>
Filter settings

<table>
<thead>
<tr>
<th>Include?</th>
<th>Activity</th>
<th># events</th>
<th># start events</th>
<th>Start activity?</th>
<th># end events</th>
<th>End activity?</th>
</tr>
</thead>
<tbody>
<tr>
<td>✓</td>
<td>Aanvraag registratie</td>
<td>842</td>
<td>842</td>
<td>✓</td>
<td>9</td>
<td>✓</td>
</tr>
<tr>
<td>✓</td>
<td>Administratieve verwerking</td>
<td>616</td>
<td>0</td>
<td>□</td>
<td>3</td>
<td>✓</td>
</tr>
<tr>
<td>✓</td>
<td>Eigen onderzoek</td>
<td>6</td>
<td>3</td>
<td>✓</td>
<td>0</td>
<td>□</td>
</tr>
<tr>
<td>✓</td>
<td>Onderzoek CIZ</td>
<td>4</td>
<td>1</td>
<td>✓</td>
<td>1</td>
<td>✓</td>
</tr>
<tr>
<td>✓</td>
<td>Rapportage &amp; beschikking</td>
<td>872</td>
<td>30</td>
<td>✓</td>
<td>18</td>
<td>✓</td>
</tr>
<tr>
<td>✓</td>
<td>Retour</td>
<td>60</td>
<td>0</td>
<td>□</td>
<td>1</td>
<td>✓</td>
</tr>
<tr>
<td>✓</td>
<td>Slotfase</td>
<td>828</td>
<td>0</td>
<td>□</td>
<td>828</td>
<td>✓</td>
</tr>
<tr>
<td>✓</td>
<td>Toetsen en beslissen</td>
<td>910</td>
<td>0</td>
<td>□</td>
<td>6</td>
<td>✓</td>
</tr>
<tr>
<td>✓</td>
<td>Verzending\dossiervorming</td>
<td>844</td>
<td>0</td>
<td>□</td>
<td>8</td>
<td>✓</td>
</tr>
<tr>
<td>✓</td>
<td>Wachten terugmelding zorgaanb.</td>
<td>515</td>
<td>0</td>
<td>□</td>
<td>2</td>
<td>✓</td>
</tr>
</tbody>
</table>
Start process discovery

<table>
<thead>
<tr>
<th>Model</th>
<th>User</th>
<th>Date</th>
<th>Version</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>appeal demo</td>
<td></td>
<td>2010-02-28 18:48:12</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>appeal demo</td>
<td></td>
<td>2010-02-28 18:50:20</td>
<td>0.1</td>
<td>New</td>
</tr>
<tr>
<td>complaints</td>
<td></td>
<td>2010-03-14 10:23:08</td>
<td>0.1</td>
<td>New</td>
</tr>
<tr>
<td>complaints-bad</td>
<td></td>
<td>2010-03-14 12:46:35</td>
<td>0.2</td>
<td>New</td>
</tr>
<tr>
<td>insurance</td>
<td></td>
<td>2010-03-01 18:30:32</td>
<td>0.9</td>
<td>New</td>
</tr>
<tr>
<td>insurance</td>
<td></td>
<td>2010-03-14 12:52:50</td>
<td>0.5</td>
<td>New</td>
</tr>
<tr>
<td>OriginalWMOlog</td>
<td></td>
<td>2010-03-14 18:54:36</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>sim</td>
<td>Administrator</td>
<td>2010-03-14 18:32:33</td>
<td>0.2</td>
<td>New</td>
</tr>
<tr>
<td>simerror</td>
<td>Mine a Process Model</td>
<td>2010-03-14 12:52:39</td>
<td>0.2</td>
<td>New</td>
</tr>
<tr>
<td>travel</td>
<td>Linked for animation to</td>
<td>2010-03-14 10:22:25</td>
<td>0.1</td>
<td>New</td>
</tr>
<tr>
<td>travel-bad</td>
<td>Mining properties</td>
<td>2010-03-14 12:52:33</td>
<td>0.2</td>
<td>New</td>
</tr>
<tr>
<td>trip</td>
<td>Delete</td>
<td>2010-02-27 19:58:02</td>
<td>0.3</td>
<td>New</td>
</tr>
<tr>
<td>trip</td>
<td>Rename</td>
<td>2010-02-27 18:36:23</td>
<td>0.5</td>
<td>New</td>
</tr>
</tbody>
</table>
More information

• Later in this course:
  – Lecture on simulation
  – Lectures on verification
  – Lectures on process mining

• Other courses:
  – Business Process Simulation (2I175)
  – Advanced Process Mining (2I166)
  – Capita Selecta Architecture of Information Systems (2I199)
  – Seminar Architecture of Information Systems (2I196)
BPM|one control ("FLOWer functionality")
Modeling and Analysis versus Implementation and Enactment
Case Handling Paradigm

- Characteristics:
  - skip, redo, ... roles
  - data-driven
  - separation of authorization and work distribution (could ≠ should)
- Pallas Athena/Perceptive Software fully implements the case handling concept through BPM|one control (formerly known as FLOWer).
## Differences between workflow management and case handling

<table>
<thead>
<tr>
<th>Focus</th>
<th>Workflow management</th>
<th>Case handling</th>
</tr>
</thead>
<tbody>
<tr>
<td>Focus</td>
<td><em>Work item</em></td>
<td><em>Whole case</em></td>
</tr>
<tr>
<td>Primary driver</td>
<td><em>Control flow</em></td>
<td><em>Case data</em></td>
</tr>
<tr>
<td>Separation of authorization and distribution</td>
<td><em>No</em></td>
<td><em>Yes</em></td>
</tr>
<tr>
<td>Separation of case data and process control</td>
<td><em>Yes</em></td>
<td><em>No</em></td>
</tr>
<tr>
<td>Data linked to roles</td>
<td><em>No</em></td>
<td><em>Yes</em></td>
</tr>
<tr>
<td>Types of roles associated with tasks</td>
<td><em>Execute</em></td>
<td><em>Execute, skip, redo, …</em></td>
</tr>
</tbody>
</table>
Focus: *Whole case*

- Avoid **context tunneling**, i.e., supply as much information about the case as possible. Do not restrict the view to data relevant for executing the next task in line.
- Allow a case worker to select, view and modify cases without explicitly selecting a work item.
Primary driver: **Case data**

- Case data is not something without any structure hidden inside applications!
- The **data elements present** determine the logi(sti)cal state of the case.
- By adding and modifying case data, the case progresses.
- By filling out a form zero or more work items may be executed!
Separation of authorization and distribution: Yes

- Many workflow management systems do not distinguish between authorization (can do) and distribution (should do).
- Moreover, authorization and distribution are typically directly linked to the process model.
- By defining authorization and distribution independently and separate from the process definition, reuse and flexibility are increased.
Separation of case data and process control: No

- The basic assumption driving most workflow management systems is a strict separation between data and process. Only the control data is managed.
- The strict separation between case data and process control simplifies things but also creates integration problems.
- For case handling the logistical state is derived from the date element present, therefore data and process cannot be separated!
In most workflow management systems case data is under the supervision of the applications and therefore the relation between data and workers is not modeled explicitly.

By linking roles to data elements, case data can be made available in every step of the process.
Types of roles associated with tasks: *Execute, skip, redo, …*

- The execute role specifies the authorization in an ideal situation. However, real case work involves skipping and rework.
- The *execute, skip and redo roles* do not need to coincide.
- Note that to discuss skipping and rework, loops need to be removed and replaced by redo’s and multiple instances (cf. patterns).
Case Handling: “Data-Driven” in more detail …

• State is determined based on data rather than only the steps executed.
• The addition of data (e.g. via forms not linked to tasks or swapped in from external applications) triggers state changes.
Mandatory and restricted data
Step 1 is on the “wavefront” and can be completed by entering Data 1 (mandatory)

<table>
<thead>
<tr>
<th>Label1:</th>
<th>Value1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Label2:</td>
<td></td>
</tr>
<tr>
<td>Label3:</td>
<td></td>
</tr>
<tr>
<td>Label4:</td>
<td></td>
</tr>
<tr>
<td>Label5:</td>
<td></td>
</tr>
</tbody>
</table>

mandatory

restricted
Step 2 is on the “wavefront” and requires Data 1 and Data 2 to complete.
Step 4 (not 3) is on the “wavefront” because all mandatory data for Step 3 was there.

Label1: Value1
Label2: Value2
Label3: Value3
Label4: Value4
Label5: 

mandatory

restricted
All four activities have completed

Label1: Value1
Label2: Value2
Label3: Value3
Label4: Value4
Label5: 

mandatory
restricted

Data 1
Data 2
Data 3
Data 4
Data 5

Step 1
Step 2
Step 3
Step 4
Convert model and refine
data element "Name" is a string used in all five forms
in this step Insurance_OK and Insurance_ID need to be entered
in this step Damage_OK and Damage need to be entered
only Name is mandatory
Insurance_ID and Insurance_OK are mandatory
NOT (Damage_OK AND Insurance_OK)
“used by”
Mandatory and restricted data

Check_Damage: Properties
- Name: Check_Damage
- Element type: Formfill Action
- Form: Form Check Dam...
- Mandatory: Damage 123
- Restricted: Check_Damage

Insurance_OK: Properties
- Initial: TRUE
- Derived: Check_Insurance
- Restricted: Check_Insurance
six cases
Enter name of claimant: "M. van der Aalst"
all mandatory data elements for remaining activities are present
redo check damage step to change amount of damage
unconfirmed values
overview form (can be opened without executing some activity)

change name

cannot be changed here (restricted)

overview form (can be opened without executing some activity)
Conclusion BPM|one

- Process (re)design
- Process adjustment
- Process execution
- Process monitoring
- Process measurement

Process analysis
Workflow Flexibility
Workflows, what people want ...

flexibility       control       support
Classical trade-off

- Groupware
- Ad-hoc workflow
- Case handling
- Production workflow

High flexibility vs. high support

"Do whatever you want but get no support"

"Support but no flexibility"
Four types of flexibility

• **Flexibility by Design**
  – allow for multiple execution paths at run-time

• **Flexibility by Deviation**
  – ability to deviate without changing the model

• **Flexibility by Underspecification**
  – ability to execute models that are completed at run-time

• **Flexibility by Change**
  – ability to modify the model at run-time
Flexibility by Design
Flexibility by Deviation

start \rightarrow a \rightarrow b \rightarrow c \rightarrow end
Flexibility by Underspecification
Flexibility by Change

start → a → b → c → end

start → a → c → end
Overview flexibility mechanisms

- **Defer**
  (decide to decide later)
  - Flexibility by Design
  - Flexibility by Under-specification

- **Deviate**
  (decide to ignore model)
  - Flexibility by Deviation

- **Change**
  (decide to change model)
  - Flexibility by Change

- Ad-hoc change
- Evolutionary change
Process Definition Completeness

<table>
<thead>
<tr>
<th>Partial</th>
<th>Full</th>
</tr>
</thead>
<tbody>
<tr>
<td>Design-time</td>
<td>Run-time</td>
</tr>
<tr>
<td>Design</td>
<td>Change Deviation</td>
</tr>
<tr>
<td>Underspecification</td>
<td>Late Binding Late Modeling</td>
</tr>
</tbody>
</table>

Flexibility Configuration
Examples Challenge: Consider a procedural language + change

- Changes: *ad-hoc* (one instance) and *evolutionary* (whole process).
- Attempts to combine the best of both worlds.

**Problems:**
- Users cannot model!
- Difficult to support.
s1
prepare_shipment
s2
send_goods
s3
prepare_shipment
send_goods
send_bill
record_shipment
p1
p2
p3
send_goods
send_bill
p4
p5
record_shipment
p6
Dynamic change bug

prepare_shipment
send_goods
send_bill
record_shipment

prepare_shipment
send_goods
send_bill
record_shipment

p1
p2
p3
p4
p5
p6
Instead of exercises make the Supplementary Assignment