Business Process Management Systems (closing & old exams)

prof.dr.ir. Wil van der Aalst
Agenda

• Summary/types of questions
• What next?
• Old Exam 19-6-2008
• Old Exam 6-8-2008

Exam:
Friday 19-4-2013 (9:00-12:00)
Lectures

- Introduction business process management systems [5-2-2013]
- Modeling the control-flow perspective [8-2-2013]
- Modeling the resource perspective [19-2-2013]
- Workflow management systems/YAWL [22-2-2013]
- YAWL & BPMone [26-2-2013]
- Workflow patterns [1-3-2013]
- Overview BPM analysis techniques & Simulation [5-3-2013]
- Verification of workflows (1) [8-3-2013]
- Verification of workflows (2) [12-3-2013]
- Process mining (1): Process discovery [15-3-2013]
- Process mining (2): Conformance checking [19-3-2013]
- Configurable process models/Service-orientation and BPM [22-3-2013]
- Exercises [26-3-2013]
- Question hour/exam preparation [2-4-2013]
Possible questions (1)

- Given a piece of text make a workflow model process definition in terms of the “standard notation” (earlier called book notation, i.e., WF-nets with syntactic suggaring, but no OR’s, no cancellation) or YAWL.

- Given a piece of text also provide a resource classification (roles&groups) and link this to the process definition using the standard notation.

- (Given a workflow model expressed in terms of the book notation, EPC notation, BPMN, Staffware, or YAWL, map it onto the book notation, Staffware, or YAWL.)

- Given a workflow model expressed in terms of Petri nets, standard notation, or YAWL, modify it to meet particular requirements.
Notations (1): Standard notation

Roles:
- Employee (E)
- Manager (M)

Groups:
- Customer Service (C)
- Inspection Services (I)
- Finance (F)
Notations (2): YAWL notation
(may include cancellation regions and OR-splits/joins)

Often without triggers (just control-flow)
Notations (3): WF-net notation (without syntactic sugaring)

Only use inhibitor/reset arcs if explicitly mentioned/allowed !!!!
Possible questions (2)

- Have a good understanding of the functionality of existing BPM systems, in particular YAWL and BPM|one.
- Be able to position YAWL, BPM|one, Staffware, SAP Workflow, COSA, and Oracle BPEL.
- Understand and be able to reproduce the WfMC reference model.
- Understand and master control-flow, resource (push-pull, FIFO, LIFO, SPT, etc.), and trigger concepts.
- Understand and be able to reproduce the life-cycle models for work-items.
- Understand business process simulation concepts (transient versus steady state, subruns, start run, confidence intervals, typical performance indicators).
Possible questions (3)

- Have a good understanding and overview of all workflow patterns.
- Know the scope of the 43 control-flow patterns, 40 data patterns and 43 resource patterns.
- Be able to explain, use, and recognize a selection of the patterns (mentioned hereafter).
- Concretely, for each of these patterns:
  - Given the name, give its description.
  - Given a model in YAWL, BPM|one, Petri nets, Staffware, etc.; identify the patterns used.
  - Know how the patterns are supported by YAWL.
Possible questions (4a)
Selected control-flow patterns (21 out of 43)

- Sequence
- Parallel Split
- Synchronization
- Exclusive Choice
- Simple Merge
- Multi-Choice
- Multi-Merge
- Arbitrary Cycles
- Deferred Choice
- Milestone
- Structured Loop
- Recursion

- Cancel Region
- Structured Partial Join
- Blocking Partial Join
- Cancelling Partial Join
- Generalised AND-Join
- Structured Synchronizing Merge
- Local Synchronizing Merge
- General Synchronizing Merge
- Critical Section
Possible questions (4b)
Selected data patterns (5 out of 40)

• Task Data
• Block Data
• Scope Data
• Case Data
• Data-Based Routing
Possible questions (4c)
Selected resource patterns (11 out of 43)

• Direct Allocation
• Role-Based Allocation
• Separation of Duties
• Distribution by Offer - Single Resource
• Distribution by Allocation - Single Resource
• Shortest Queue
• Resource-Initiated Allocation
• Resource-Initiated Execution - Allocated Work Item
• Delegation
• Stateful Reallocation
• Piled Execution
• Provide an overview of the various analysis techniques: process mining (discovery, conformance, and extension), verification, validation, and performance analysis (simulation).

• Map Petri nets, Petri nets with resets, Petri nets with inhibitors, and standard notation onto transition systems (aka reachability graph, process space/process).

• Apply analysis techniques involving the coverability graph, place invariants, and transition invariants.
Possible questions (6)

- Reproduce the formal definition of soundness.
- Determine soundness for WF-nets/models using the standard notation.
- Determine soundness for WF-nets with reset and inhibitors.
- Replace reset arcs by inhibitor arcs.
- Remove arcs weights.
- Understand and determine properties such as boundedness, safeness, liveness, deadlock-freeness, reversibility.
- Relate soundness to liveness and boundedness.
- When is soundness decidable?
- Determine whether a Petri net/WF-net is a state machine, a marked graph, free-choice, or well-handled/well-structured.
- Give a net that is 1-sound but not 2-sound (or similar).
Possible questions (7)

- Understand the relevance and role of process mining.
- Understand how the Theory of Regions relates to process discovery.
- Generate a transition system based on an event log, e.g., based on the “multiset activities executed in the past” or the "sequence consisting of the next 2 activities".
- Determine all minimal non-trivial regions based on a transition system.
- Show whether a transition system is elementary or not (state separation and forward closure)!
- Construct the minimal saturated Petri net based on regions (without label splitting).
• Compute the conformance (i.e., fitness based on produced, consumed, remaining, and missing tokens) for a given log and WF-net.
• Understand the practical relevance of conformance checking and process discovery.
• Understand the meaning and relevance of SaaS, PaaS, IaaS, SOC, etc.
• Determine all valid configurations of a configurable process model (configurable WF-net).
• Given a set of variants (WF-nets) create a configurable process model (configurable WF-net).
• Understand the relation between process configuration and process mining.
Removed from course (compared to before 2011)

- Detailed knowledge of Staffware, BPMN, and EPC notations.
- Detailed knowledge of various equivalence notions (rooted branching bisimulation, etc.) and corresponding formalizations.
- Application of label splitting to be able to always create a bisimilar Petri net using regions.
New material (compared to before 2011)

- Computing the conformance of model and log (i.e., fitness based on produced, consumed, remaining, and missing tokens).
- Determining all valid configurations of a configurable process model.
- Creating a configurable process model (configurable WF-net) for a set of variants (WF-nets).

Do not just look at old exams, also focus on last couple of lectures/instructions!
Questions?
What's next?
What's next?

- Elective courses (2013-2014)
  - Process Mining (Q4, 1II66)
  - Business Process Simulation (Q2, 2II75)
  - Meta Modeling and Interoperability (Q2, 2II65)
  - Seminar IS (Q2, 2II96)
- Capita Selecta AIS (2II95, only by invitation)
- Internships (for best students)
- Master projects
Overview AIS

Perfect score in evaluation of all Dutch Computer Science groups (2010): 5-5-5-5-5
http://tinyurl.com/43h5zfo
Topics

Process Mining
- Process discovery
- Clustering
- Model extension
- Conformance checking
- Visualization
- Petri nets
- WF-nets
- LTL
- Extended WF-nets (open, with data, with resources)

PAIS Technology
- BPM\one
- Business process management systems
- WebSphere
- Grid technology
- SOA
- Web services
- Configuration
- Declare
- Resource patterns
- Data patterns
- Control-flow patterns
- Adapter generation
- (Short-term) simulation
- Verification (soundness etc.)

Process Modeling/Analysis
- Discovery
- Recommendation
- Prediction
- Control-flow patterns
- Data patterns
- Resource patterns
- Verification (soundness etc.)
- Simulation adapter generation
- Business process management systems
- Grid technology
- SOA
- Web services
- Configuration
- Declare
- Resource patterns
- Data patterns
- Control-flow patterns
- Adapter generation
- (Short-term) simulation
- Verification (soundness etc.)

CPN Tools
- YAWL
- BPEL, BPMN, EPCs, UML, YAWL, etc.
Master projects AIS  

**a) Internal assignments**  
(in areas mentioned before)

**b) External assignments within organizations such as**

- Pallas Athena (NL): process mining, simulation, case handling, and process configuration
- Futura Process Intelligence (NL): process mining and process discovery
- Philips Healthcare (NL): process mining based on event logs of medical devices
- IBM Research (Switzerland/US): workflow patterns and analysis
- IBM Development (Germany/US): case handling and process mining in WebSphere
- SAP AG (Germany/Australia): semantic process mining of ERP systems
- Océ (NL): Petri-net-based modeling and analysis of copiers
- Thales (NL): adapter generation and interface discovery in systems of systems
- IDS Scheer (Germany): process mining and social network analysis
- Academisch Medisch Centrum (NL): workflow management and process mining for hospitals
- ING Group (NL): process redesign and analysis in investment banking
- ILOG/IBM (France): optimization and planning
- Deloitte (NL): IT support for auditing using process mining and process modeling
- Gemeente Harderwijk (NL): process mining and business process modeling
- APG (NL): process mining, workflow management and business process modeling
- PwC (NL): “business process forensics” based on process mining
Interested?

w.m.p.v.d.aalst@tue.nl

+31 40 247.4295/2733

www.vdaalst.com
Exam Business Process Management Systems (2II55)
19-6-2008, 9:00 - 12:00
Consider an electronic bookstore that sells books via the internet. The process starts when a potential customer visits the website. The customer can browse through the catalogue. While browsing she can add things to her shopping cart. Independent of the browsing (i.e., the actual "shopping"), the customer can logon. A new customer needs to register first while known customers can logon without this extra step. A logon attempt may fail when the incorrect usercode/password is given. Customers that are logged on can also logoff and logon again. A customer can add items to/remove items from his/her shopping cart in multiple steps. So the user may add "Book1", then add "Book2", then logon, and then remove "Book2", then logoff, followed by adding "Book 3", etc. At any point in the process, the customer may decide to leave the website or decide to finalize the order. Customers that did not logon need to do so to be able to finalize the order.
To finalize the order, the customer needs to choose a payment and shipping method. There are two shipping methods: rush order and normal order. In case of a rush order additional information is asked after contacting the shipping company (the customer has to select a suitable delivery time from a list of possible times). This information is not needed for a normal order. There are two payment options: credit card and bank transfer. The credit card information needs to be checked by an employee. In case of a bank transfer, the bank is informed of the expected payment and one needs to wait for a trigger from the bank. Both checks may be positive or negative (e.g., the bank transfer failed or the credit card number is invalid). If it is negative, the customer needs to select another payment option.

After finalizing the order (i.e., all information is gathered and the payment is checked), the availability of the books is checked. If they are not available, the customer is informed. This is repeated on a weekly basis by e-mail. Once they are available, the books are shipped to the address indicated. Again the customer is informed about this by e-mail. Finally, the order is archived the customer profile is updated.
Model the above workflow process using the notation of the book, i.e., a process definition (i.e., WF-net including triggers). Note that in this case the resource classification is trivial so it can be skipped. Make sure that your process definition is sound, i.e., it is always possible to end properly (token in final place) and there are no dead parts/dangling tokens. Test the process model using some typical scenarios to make sure that it is always possible to terminate properly!
Customers are seen as resources and require trigger symbol.
Customers are seen as resources and require trigger symbol.
Customers are seen as resources and require trigger symbol.

ps.
Translate the above model into a Staffware model. Note that some of the patterns supported by EPCs are not supported by Staffware. If a direct translation is not possible, provide an indirect translation to get the same resulting behavior.
Solution assignment 2

Not part of exam anymore, but could be asked for YAWL, WF-nets with resets, etc.
Assignment 3 (1.5 points)

a) Construct a sound process model using the book notation that uses the following patterns:
   • the Milestone pattern (make sure it is a real milestone and not a complicated way of modeling something simple)
   • the Deferred choice pattern
   • the Structured synchronizing merge pattern

For each of these patterns, clearly indicate where it can be found in the model.

b) Using a suitable diagram, describe the lifecycle for a work item undergoing distribution and execution in workflow (i.e., identify the basic states and state transitions of a work item).

c) Explain the Piled execution pattern. How does it operate and why is it useful? Give an example of its use in real life.
Pattern description:

- The ability of the workflow system to initiate the next instance of a workflow task (perhaps in a different case) once the previous one has completed.
- Provides a means of optimising task execution by pipelining instances of the same task and allocating them to the same resource.
- The resource undertakes work items sequentially and once a work item is completed, if another work item of the same type is present in the work queue, it immediately commences work on it - in effect it attempts to work on piles of the same types of work items.
- The aim is to leverage of experience of a resource in performing a certain type of task.
- Example: correcting exams per assignment, i.e., correct all answers to question 1, etc.
a) The WF-net is not sound. Give the coverability graph and explain how to see the error.

b) Although the WF-net is not sound, the model has "sound runs", i.e., firing sequences starting in [p1] and ending in [p5]. Give a sound WF-net (potentially having reset and/or inhibitor arcs) that allows for exactly the behavior corresponding these sound runs (modulo bisimulation).
It shows that "infinite garbage" may be left in place p3.
Note that a reset arc between t5 and p3 also leads to a sound model, however, then there is the problem that the sound run t1 t2 t2 t4 t5 is possible while this was not a sound run in the original model.
Assignment 5 (1.5 points)

Consider the transition system below that needs to be converted into a bisimilar Petri net using the Theory of Regions.

a) Show whether the transition system is elementary or not.

b) Construct the minimal saturated Petri net corresponding to this transition system. (Also provide intermediate steps showing how the result was obtained.)
Regions (1/2)
Regions (2/2)
a)

GER(a) = \{i,1,4\}  GER(b) = \{i,0,2\}
GER(c) = \{i,1,3\}

Intersection

Pre(a) = \{i,1,2,4\}  \{i,1,2,4\}
Pre(b) = \{i,0,2\}  \{i,0,2\}
Pre(c) = \{i,0,1,3\}  \{i,0,1,3\}

proper subsets of \{i,1,2,4\} including GER(a):
\{i,1,4\} splitting b and c

proper subsets of \{i,0,1,3\} including GER(c):
\{i,1,3\} splitting a and b

Not elementary since GER(a) does not coincide with the intersection of all pre-regions. The same holds for GER(c).
\{i,1,4\} splitting b and c

\{i,1,3\} splitting a and b
GER(a) = \{i,1,4\}  \hspace{1cm} GER(b) = \{i,0,2\}
GER(c) = \{i,1,3\}

Intersection

Pre(a) = \{i,1,2,4\}  \hspace{1cm} \{i,1,2,4\}
Pre(b) = \{i,0,2\}  \hspace{1cm} \{i,0,2\}
Pre(c) = \{i,0,1,3\}  \hspace{1cm} \{i,0,1,3\}

proper subsets of \{i,1,2,4\} including GER(a):
\{i,1,4\} splitting b and c

proper subsets of \{i,0,1,3\} including GER(c):
\{i,1,3\} splitting a and b

Split the labels of a in:
a1) for transitions a entering \{i,1,3\},
a2) for transitions a not crossing \{i,1,3\},
a3) for transitions a exiting \{i,1,3\},
AND
Split the labels of b in:
b1) for transitions b entering \{i,1,4\},
b2) for transitions b not crossing \{i,1,4\},
b3) for transitions b exiting \{i,1,4\},
AND
Split the labels of c in:
c1) for transitions c entering \{i,1,4\},
c2) for transitions c not crossing \{i,1,4\},
c3) for transitions c exiting \{i,1,4\},
First, determine all minimal regions and generalized excitation regions.

For each generalized excitation region \( GER(e) \) that is not the intersection of the pre-regions of \( e \), find a proper subset \( * S \) of the intersection of the pre-regions of \( e \), such that \( GER(e) \) is a subset of \( S \) and \( S \) has the least number of events violating the region conditions.

If multiple proper subsets have the same number of events violating the region conditions, then choose one of the smallest sets.

*) The proper subsets of \( S \) are all subsets, not including \( S \).
But given any transition system you should be able to do this!

This transition system is Elementary

or

This transition system is Elementary
Examen Business Process Management Systems (2II55)
6-8-2008, 14:00 - 17:00
Consider the application process for a mortgage of some bank. A mortgage is a loan for buying e.g. a house. The bank has two departments relevant for this process: finance and customer relations.

The process starts by registering the request for a mortgage by an employee of the customer relations department. Then the process is checked for completeness by an employee of the same department. If the information is incomplete, the applicant is contacted by the same employee. If the applicant sends this information within two weeks, the additional information is added by an employee of the customer relations department. If not, the process ends because of insufficient information.
When all information required is present, two checks are conducted (in any order). The credit history is checked by an employee of the customer relations department and the financial data is checked by an employee of the finance department. Based on these two checks, a manager of the customer relations department makes a conditional decision. If the conditional decision is negative, the customer is notified by an employee and the process ends. If the conditional decision is positive, the interest rate is determined and the property is checked. Both tasks can be done in any order and need to be done by experts of the customer relations department.
After these two tasks, a final decision is made when a manager of the customer relations department has a meeting with the applicant. The final decision may be that no agreement can be reached and, hence, the process ends. It can also be that during the meeting there is consensus on the conditions and the offer is subsequently stored in the information system of the bank. Moreover, it may also be the case that no immediate consensus can be reached but that the customer does not decline the offer immediately. In the later case, the applicant has one week to indicate that she will take the offer anyway. If there is no response within a week, the process ends. If there is a response in one week, the offer is stored in the information system of the bank.
• Model the above workflow process using the notation of the book, i.e., a process definition (i.e., WF-net including triggers). Also give the resource classification and link it to the process definition. Make sure that your process definition is sound, i.e., it is always possible to end properly (token in final place) and there are no dead parts/dangling tokens.
Note that in the above solution the completeness is checked again after obtaining additional information and that this process may be repeated. This is assumed and not explicitly in the text. Of course it is also allowed to change the arrow from record additional info from c1 to c4.
Assignment 2 (2 points)

- Model the following process in terms of YAWL (only control-flow).
- The process starts with task A. After executing A four tasks (B1, B2, B3, and B4) need to be executed. B1, B2, B3, and B4 can be executed in any order or even in parallel. Each of these four tasks can succeed or fail. If one of them fails, all other tasks are cancelled and task C is executed after which the process ends. If all four tasks succeed, then three other tasks can be executed in any order or even in parallel: D1, D2, and D3. These tasks are optional, i.e., some of them may be skipped. After completing (or skipping) tasks D1, D2, and D3, the process ends by executing E multiple times. The number of times E needs to be executed depends on the data associated with the case. The required number of instances of E can be executed in any order.
- Make sure that your YAWL model is sound. Also try to use the advanced constructs of YAWL to realize a compact model.
Solution assignment 2
Typical errors:

- Condition c5 outside cancellation region: cancel can be executed multiple times resulting in an unsound process.
- No OR-split or OR-join is used to model the optional parallel flows involving D1, D2, and D3. (Note that we assume at least one of them to happen. If desired, an extra arc can be added to also allow for skipping D1, D2, and D3.)
- No YAWL start and end of process.
- Task E is modeled as some loop rather than a multiple instance task (cf. text: based on data, can be executed in any order, etc.).
Assignment 3 (1.5 points)

• Construct a sound process model using the Staffware notation that uses the following patterns:
  • the Exclusive choice pattern
  • the Multi-merge pattern
  • the Structured synchronizing merge pattern

• For each of these patterns, clearly indicate where it can be found in the model. Also describe each of the patterns in a few words
Pattern: Exclusive Choice
- The divergence of a branch into two or more branches. When the incoming branch is enabled, the thread of control is immediately passed to precisely one of the outgoing branches based on the outcome of a logical expression associated with the branch.
- Directly supported by Staffware (note yellow triangle) but with the limitation that the choice can only be binary.

Pattern: Structured Synchronizing Merge
- The convergence of two or more branches (which diverged earlier in the process at a uniquely identifiable point) into a single subsequent branch. The thread of control is passed to the subsequent branch when each active incoming branch has been enabled.
- Not supported by Staffware. However, in the example the main idea is implemented in an indirect manner (by skipping tasks and then synchronizing all branches)

Pattern: Multi-Merge
- The convergence of two or more branches into a single subsequent branch. Each enablement of an incoming branch results in the thread of control being passed to the subsequent branch.
- Not supported by Staffware. However, in the example the main idea is implemented in an indirect manner (by duplicating G such that G can be executed multiple times without getting canceled through Staffware's packman semantics).
• Typical errors:
  • "Multi-merge" is not the same as "Simple Merge"!
  • "Structured Synchronizing Merge" is not the same as "Synchronization"!
a) The WF-net is not sound. Give the coverability graph and explain how to see the error.

b) Although the WF-net is not sound, the model has "sound runs", i.e., firing sequences starting in \([p1]\) and ending in \([p6]\). Give a *sound* WF-net (potentially having reset and/or inhibitor arcs) that allows for exactly the behavior corresponding these sound runs.
Solution

coverability tree

[p1]  
  ↓  
[p2]  
  ↓  
[p2,p3\^[ω]]  
  ↓  
[p2,p3\^[ω],p4\^[ω]]  
  ↓  
[p5,p3\^[ω],p4\^[ω]]  
  ↓  
[p6,p3\^[ω],p4\^[ω]]  
  ↓  
[p5,p3\^[ω]]  
  ↓  
[p5,p3\^[ω]]  
  ↓  
[p5,p3\^[ω]]  
  ↓  
[p1]  
  ↓  
[p2]  
  ↓  
[p2,p3\^[ω]]  
  ↓  
[p2,p3\^[ω],p4\^[ω]]  
  ↓  
[p5,p3\^[ω],p4\^[ω]]  
  ↓  
[p6,p3\^[ω],p4\^[ω]]  
  ↓  
[p5,p3\^[ω],p4\^[ω]]

coverability graph

coverability graph by ProM

dead end

old
Typical errors:

- Still a deadlock is possible.
- A sound sequence is no longer possible, e.g., \( t_1, t_2, t_6, t_3, t_2, t_4, t_3, t_5 \).
- An unsound sequence in original model becomes possible in new model, e.g., \( t_1, t_2, t_2, t_4, t_5 \).
a) Show whether the transition system is elementary or not.

b) Construct the minimal saturated Petri net corresponding to this transition system. (Also provide intermediate steps showing how the result was obtained.)
GER(a) = \{i,1,4\}
GER(b) = \{i,0,2\}
GER(c) = \{0,1,3\}
GER(d) = \{3\}

Pre(a) = \{i,1,4\} \{i,1,4\}
Pre(b) = \{i,0,2\} \{i,0,2\}
Pre(c) = \{i,0,1,3\} \{i,0,1,3\}
Pre(d) = \{0,2,3,5\}, \{1,3,4,5\} \{3,5\}

intersections

proper subsets of \{i,0,1,3\} including GER(c):
\{0,1,3\} splitting a and b

proper subsets of \{3,5\} including GER(d):
\{3\} splitting a,b and c
Problem: intersection of preregions \{3,5\} is different from generalized excitation region \{3\}

\[ \text{GER}(d) = \{3\} \]

\[ \text{Pre}(d) = \{0,2,3,5\},\{1,3,4,5\} \]
GER(a) = \{i, 1, 4\}
GER(b) = \{i, 0, 2\}
GER(c) = \{0, 1, 3\}
GER(d) = \{3\}

intersection

Pre(a) = \{i, 1, 4\} \quad \{i, 1, 4\}
Pre(b) = \{i, 0, 2\} \quad \{i, 0, 2\}
Pre(c) = \{i, 0, 1, 3\} \quad \{i, 0, 1, 3\}
Pre(d) = \{0, 2, 3, 5\}, \{1, 3, 4, 5\} \quad \{3\}

proper subsets of \{i, 0, 1, 3\} including GER(c):
\{0, 1, 3\} splitting a and b

proper subsets of \{3, 5\} including GER(d):
\{3\} splitting a,b and c

Split the labels of a in:
a1) for transitions a entering \{0, 1, 3\},
a2) for transitions a not crossing \{0, 1, 3\},
a3) for transitions a exiting \{0, 1, 3\},

AND

Split the labels of b in:
b1) for transitions b entering \{0, 1, 3\},
b2) for transitions b not crossing \{0, 1, 3\},
b3) for transitions b exiting \{0, 1, 3\},
For all events holds that the intersection of the pre-regions equals the generalized excitation region.

Each pair of states can be separated by a region.

This transition system is Elementary
Good Luck!
on Friday 19-4-2013 (9:00-12:00)