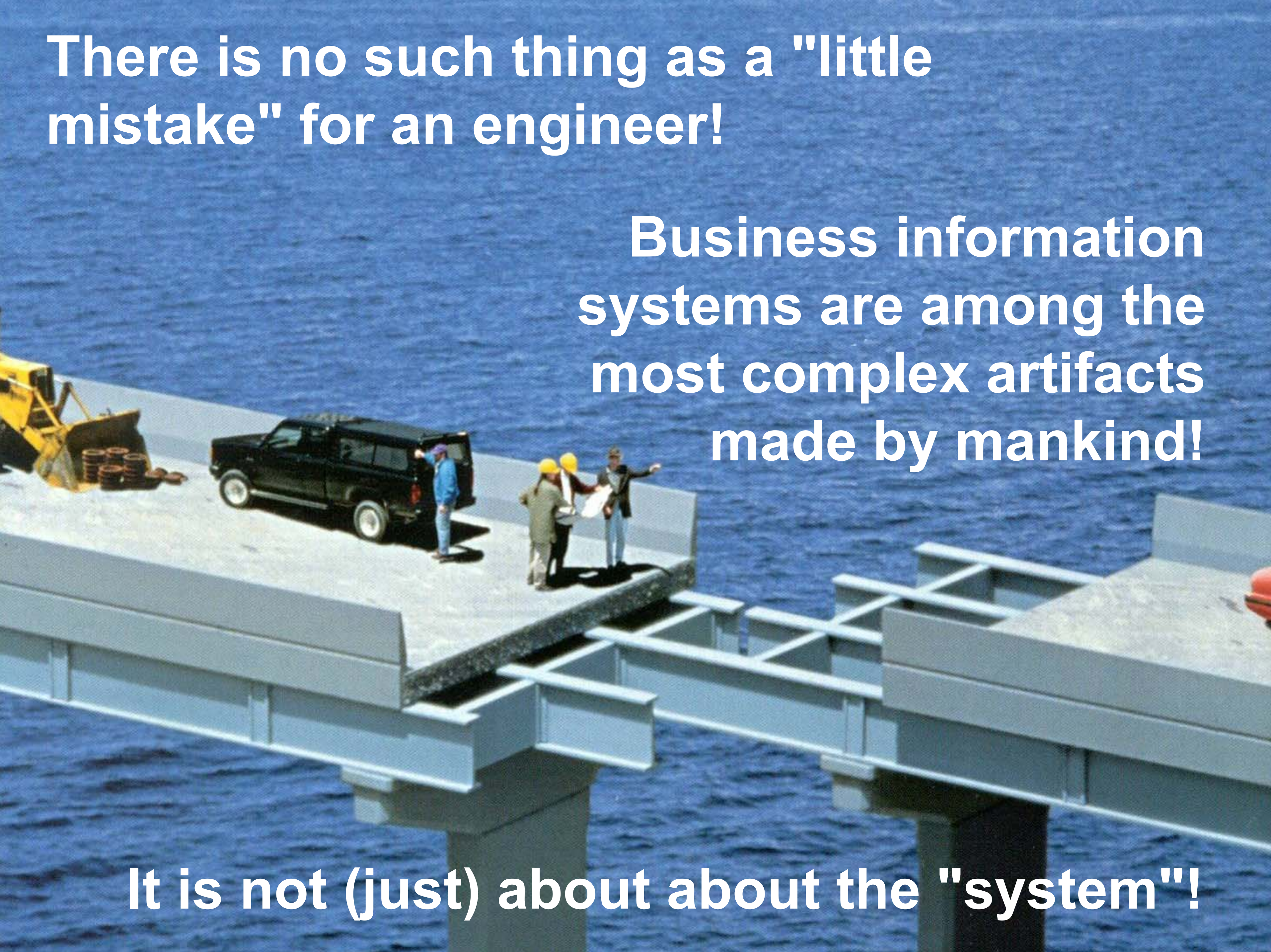
Process Mining and other BPM Challenges for the Graph Transformation Community

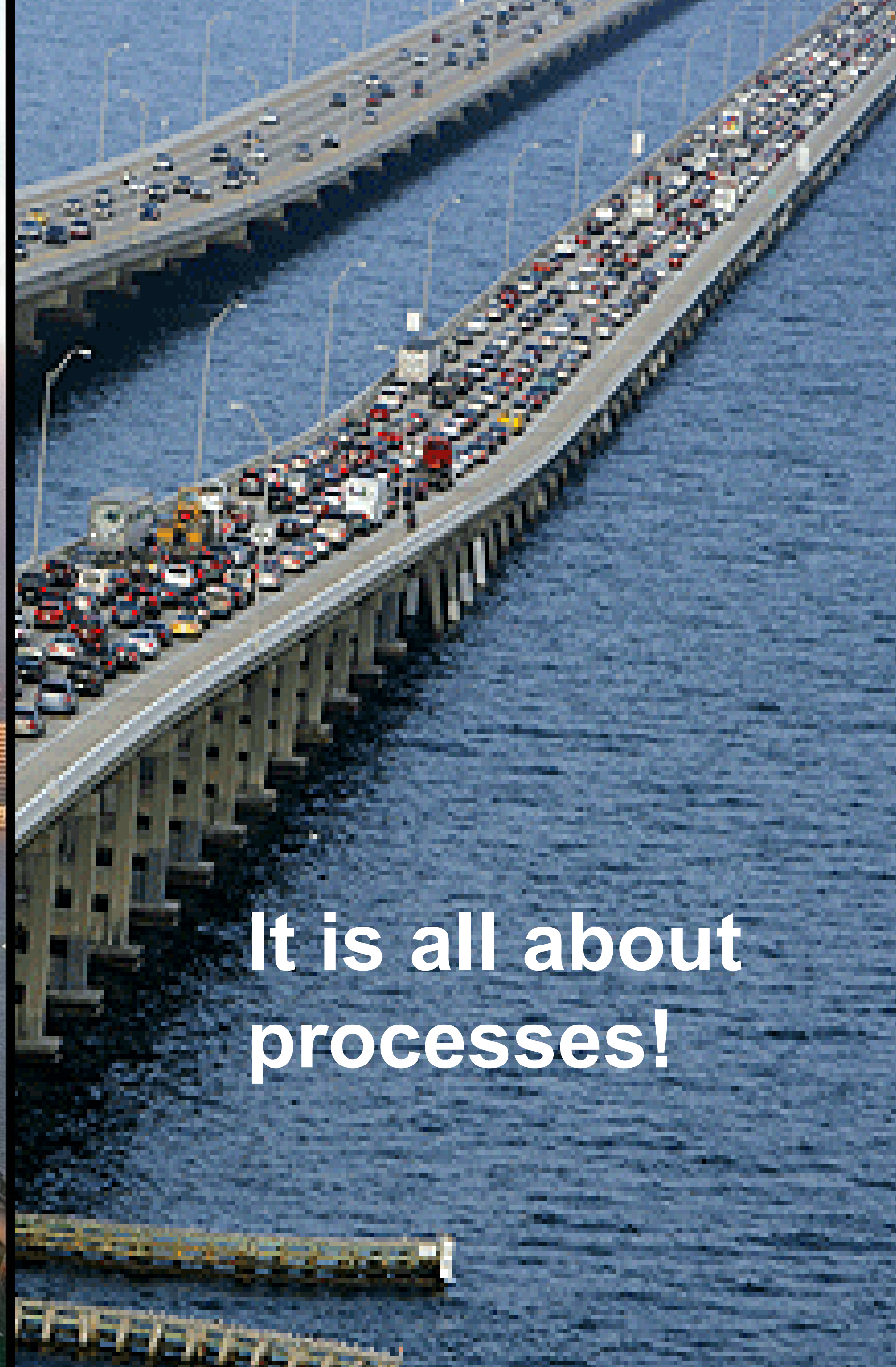
prof.dr.ir. Wil van der Aalst

4th International Conference on Graph Transformation
Leicester (United Kingdom), September 7 - 13, 2008

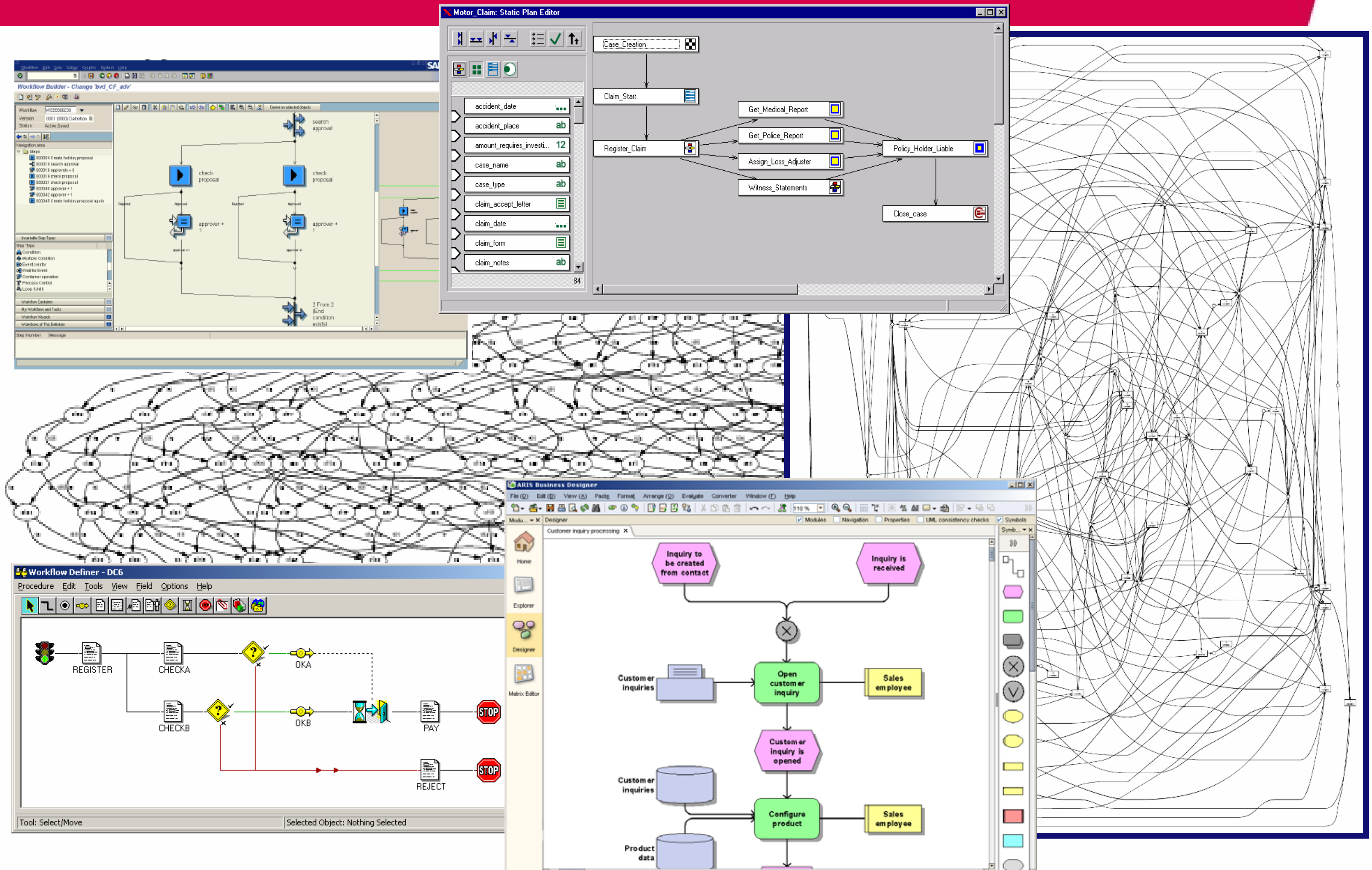
TU/e Technische Universiteit Eindhoven University of Technology

Where innovation starts

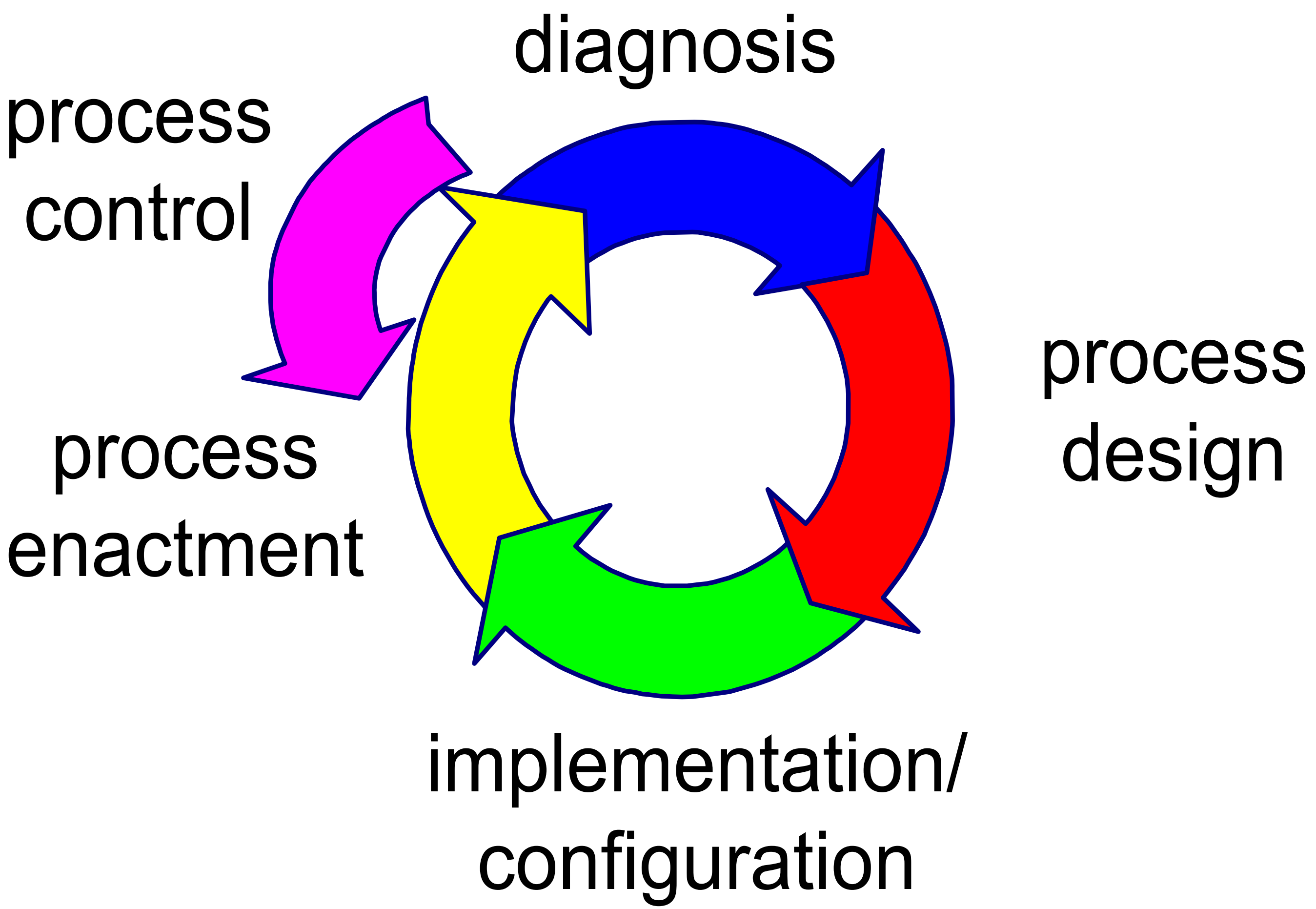

There is no such thing as a "little mistake" for an engineer!
Business information systems are among the most complex artifacts made by mankind!
It is not (just) about about the "system"!


It is all about processes!

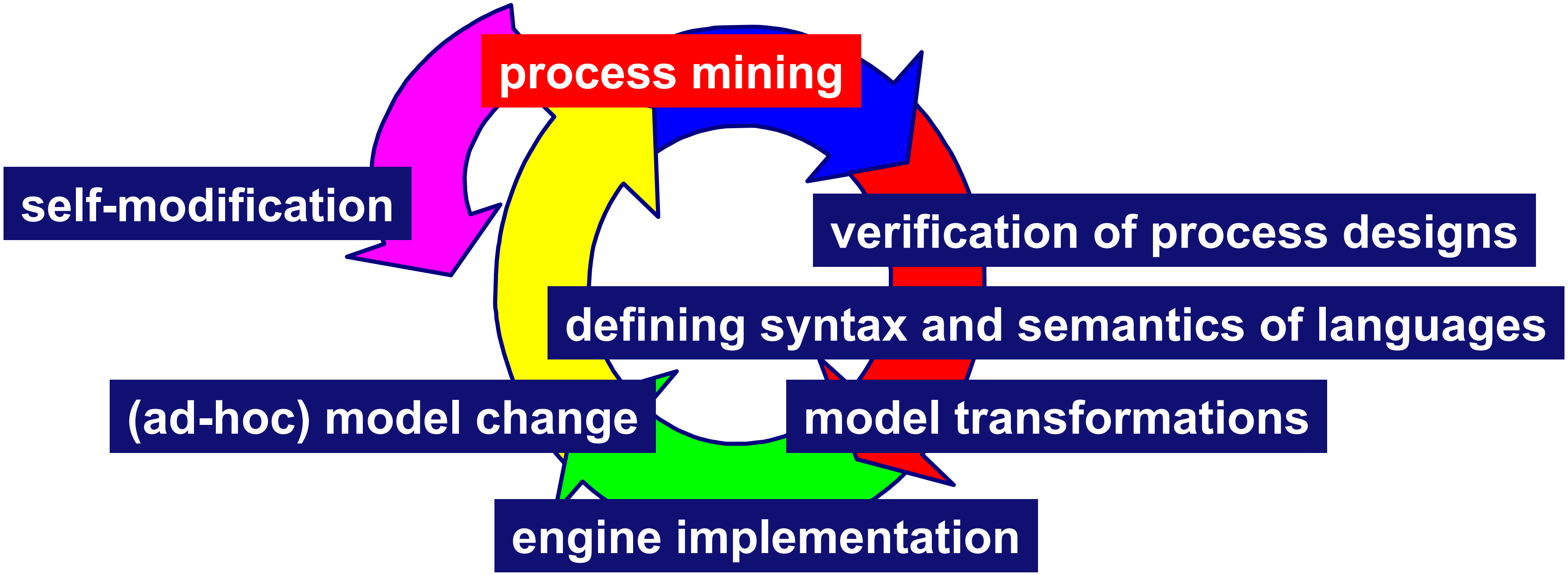
Graphs in Business Process Management



Business Process Management (BPM) lifecycle



Challenges for the Graph Transformation Community



Some personal observations

1. Disconnect between models and systems (unless models are directly used for enactment).
2. Models ≠ reality.
3. Simulation results ≠ real performance.
4. Too much focus on design time (cf. conformance, etc.).
5. Large scale verification is possible but people do not care (most of the time).

Overview of Process Mining

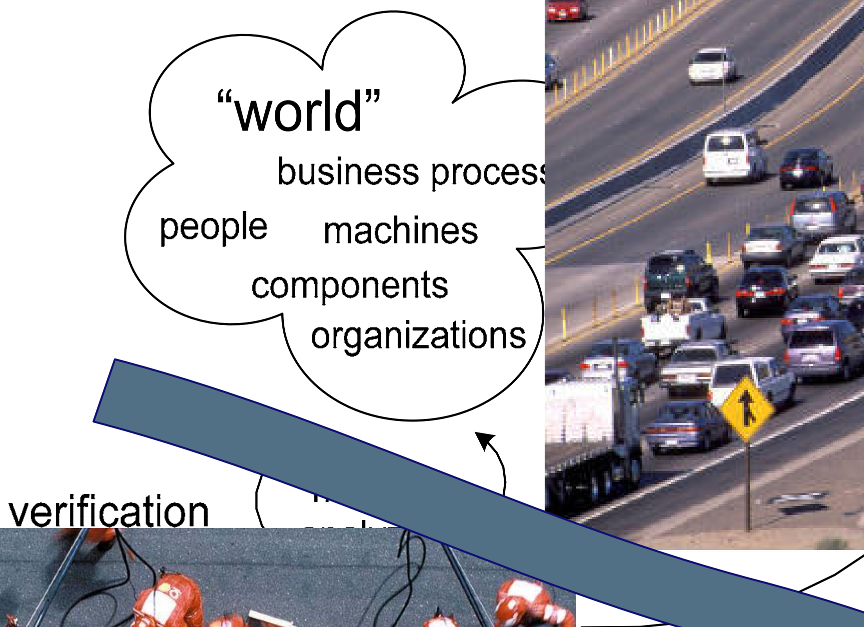
Technische Universiteit
Eindhoven
University of Technology

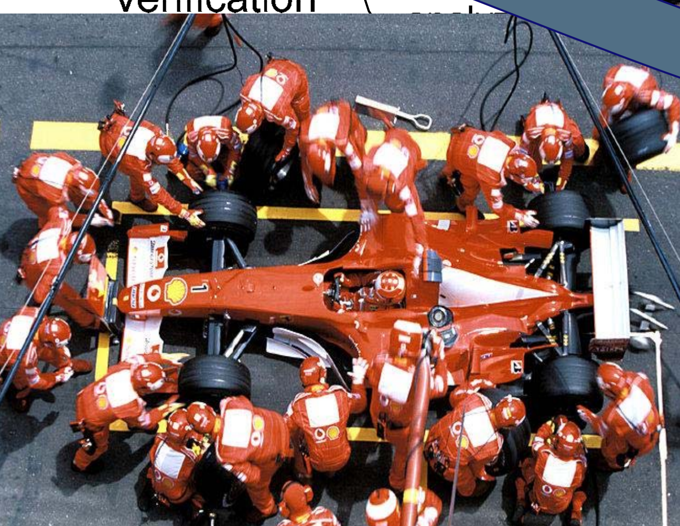
Where innovation starts

Thanks!

- Wil van der Aalst
- Peter van den Brand
- Boudewijn van Dongen
- Christian Günther
- Eric Verbeek
- Ana Karla Alves de Medeiros
- Anne Rozinat
- Minseok Song
- Ton Weijters
- Remco Dijkman
- Gianluigi Greco
- Antonella Guzzo
- Kristian Bisgaard Lassen
- Ronny Mans
- Jan Mendling
- Vladimir Rubin
- Kenny van Uden
- Irene Vanderfeesten
- Barbara Weber
- Lijie Wen
- Mercy Amiyo
- Carmen Bratosin
- Toon Calders
- Jorge Cardoso
- Ronald Crooy
- Florian Gottschalk
- Monique Jansen-Vullers
- Peter Khisa Wakholi
- Nicolas Knaak
- Sven Lambrechts
- Joyce Nakatumba
- Mariska Netjes
- Mykola Pechenizkiy
- Maja Pesic
- Hajo Reijers
- Stefanie Rinderle
- Domenico Saccà
- Helen Schonenberg
- Marc Voorhoeve
- Jianmin Wang
- Jan Martijn van der Werf
- Martin van Wingerden
- Jianhong Ye
- Huub de Beer
- Elena Casares
- Alina Chipaila
- Walid Gaaloul
- Martijn van Giessel
- Shaifali Gupta
- Thomas Hoffmann
- Peter Hornix
- René Kerstjens
- Ralf Kramer
- Wouter Kunst
- Laura Maruster
- Andriy Nikolov
- Adarsh Ramesh
- Jo Theunissen
- ...

Role of models

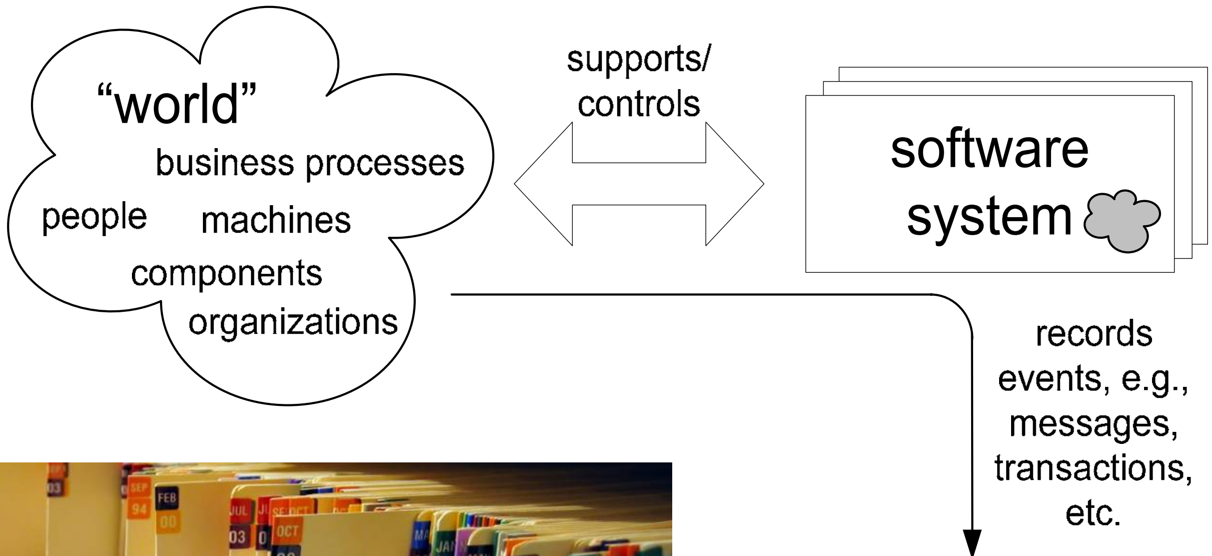


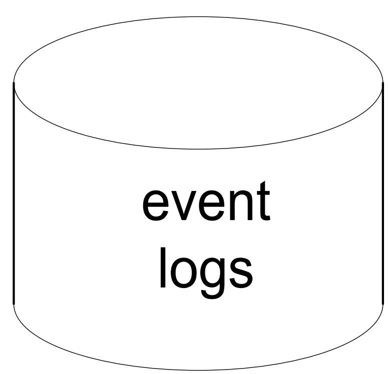


real world

powerpoint reality

Event logs are a reflection of reality





Examples:

IBM

staffware. THE POWER OF PROCESS

Pallas Athena PROCESS MANAGEMENT

FileNet An IBM Company

océ

SAP

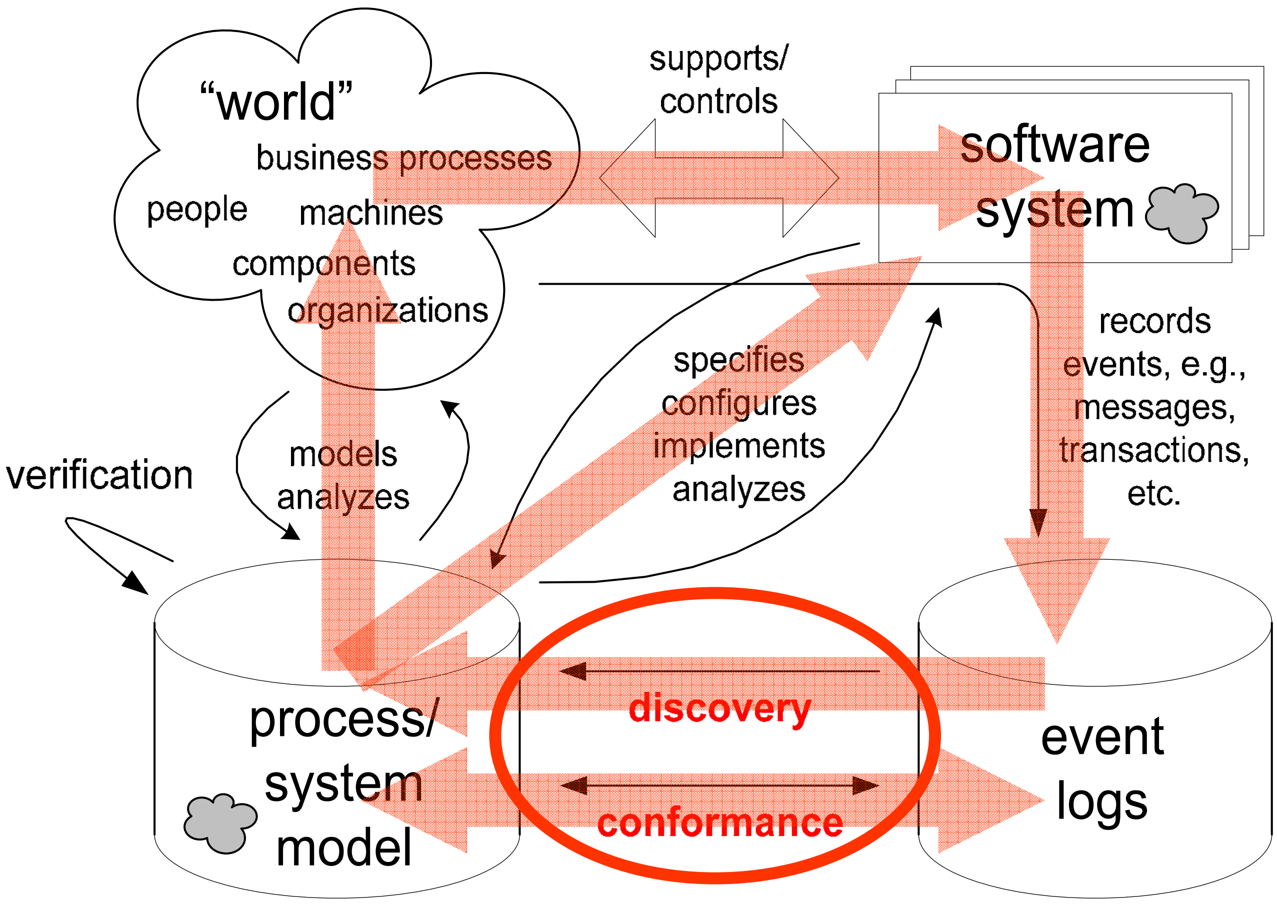
PHILIPS sense and simplicity

ORACLE

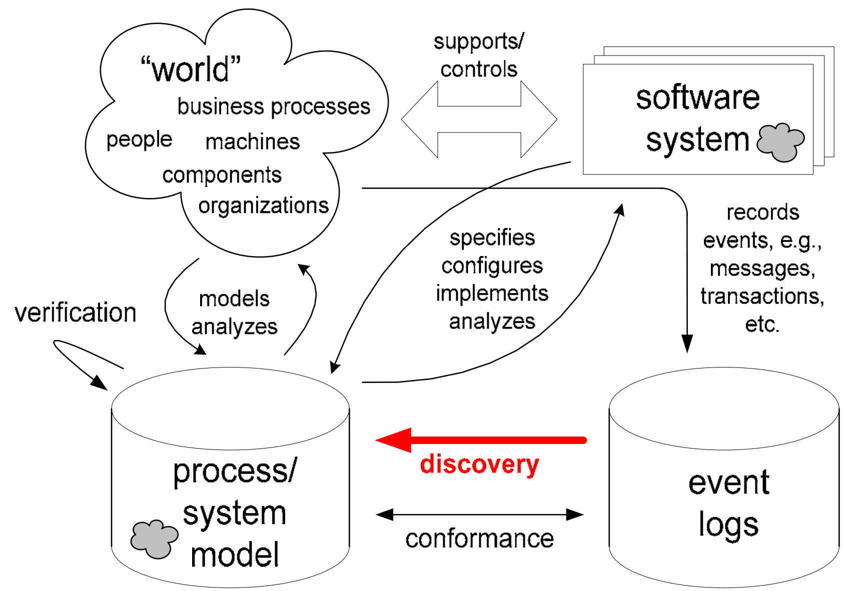
ASML

WebSphere software

Process mining: Linking events to models

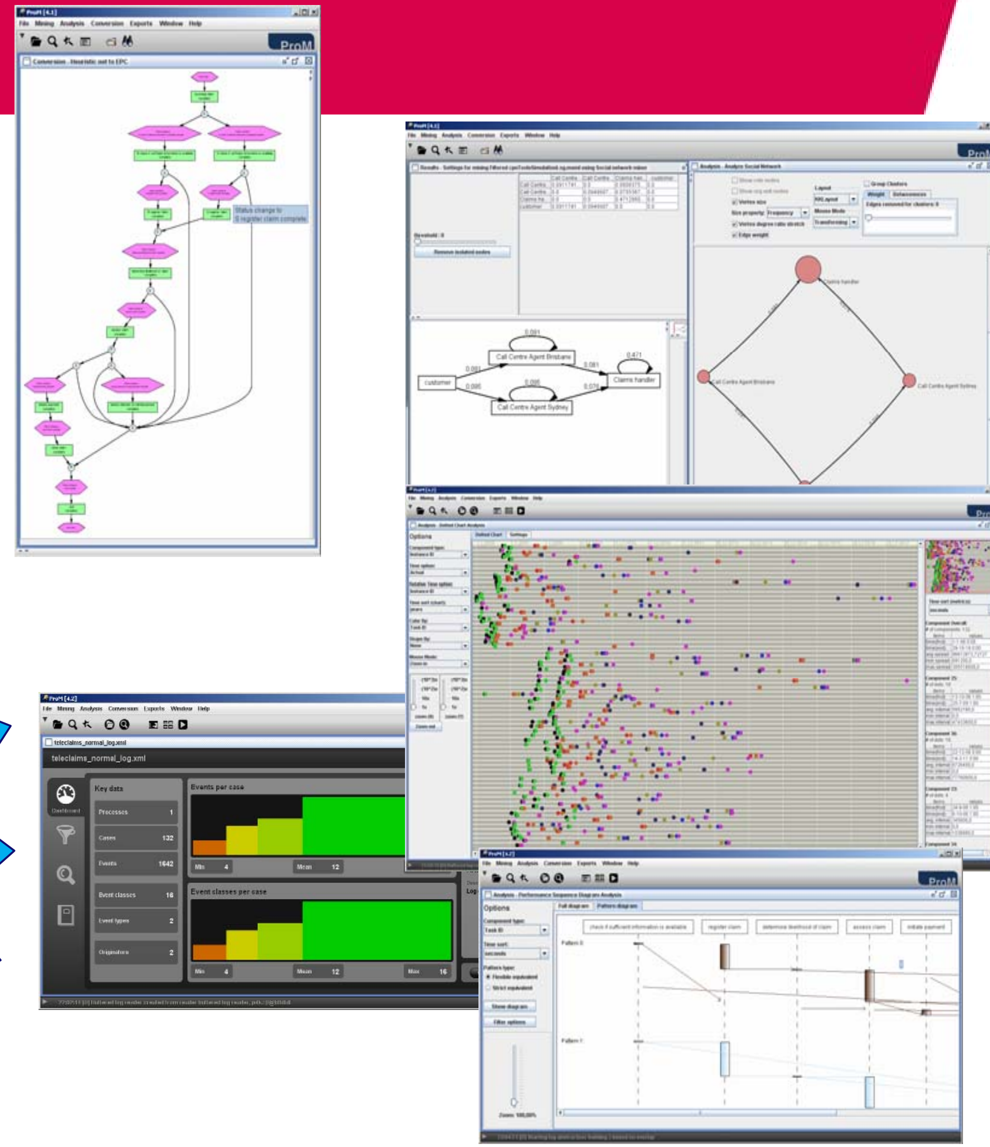


Discovery



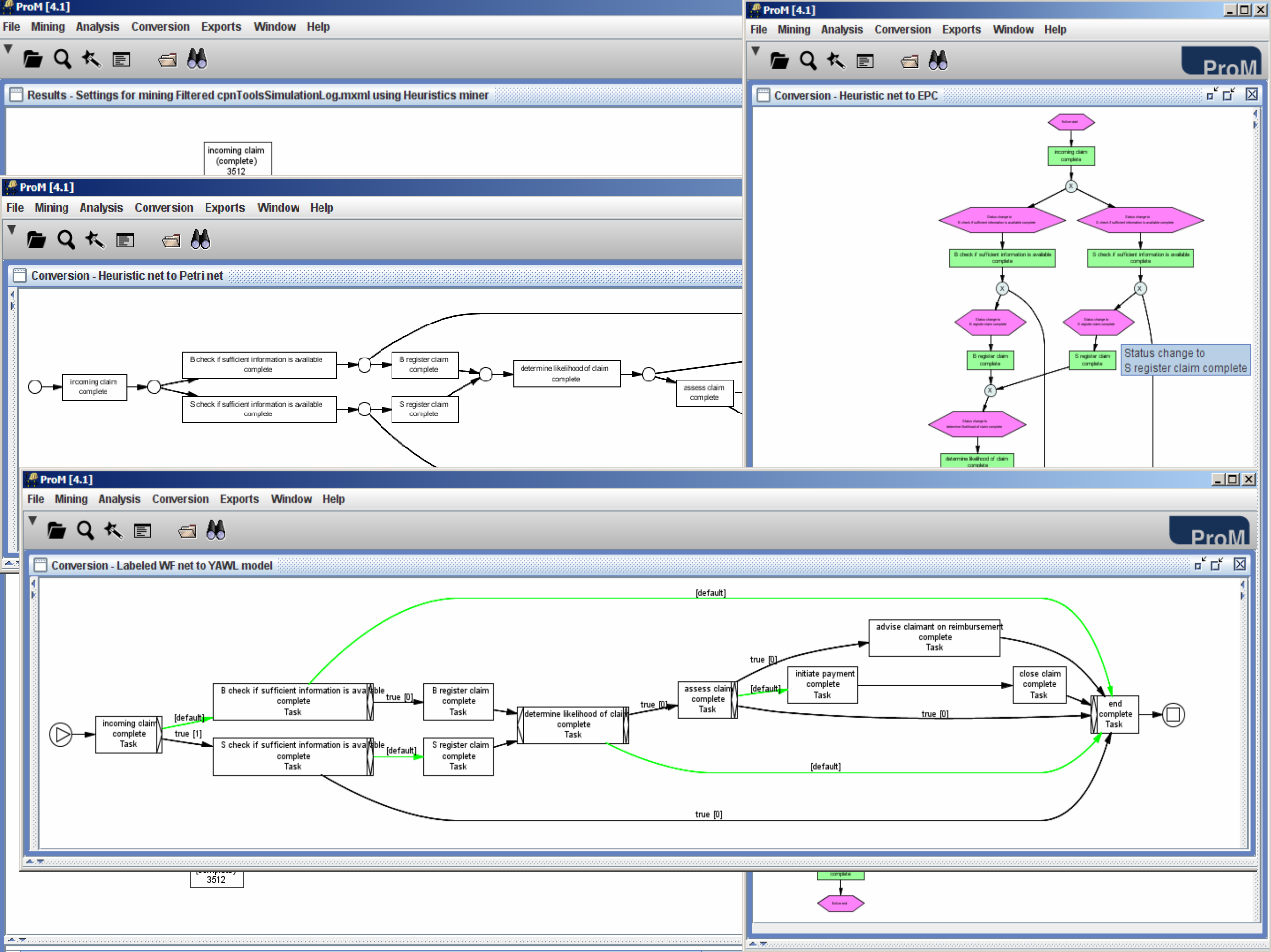
"world"
business processes
people
machines
components
organizations
supports/
controls
software
system
records
events, e.g.,
messages,
transactions,
etc.
specifies
configures
implements
analyzes
models
analyzes
verification
process/
system
model
discovery
conformance
event
logs

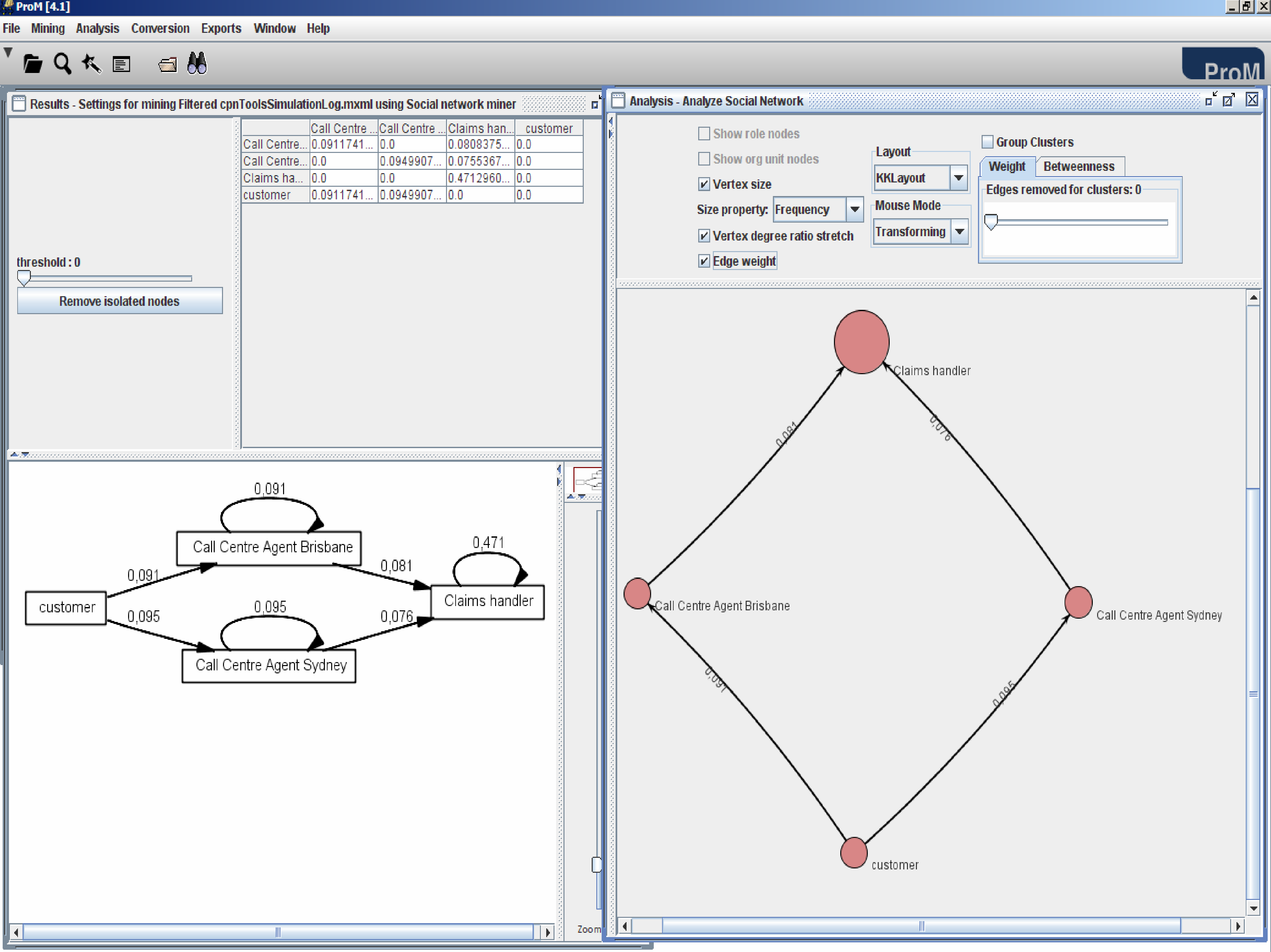
MXML Log
- **instances:** 3512
- **audit trail entries:** 46138



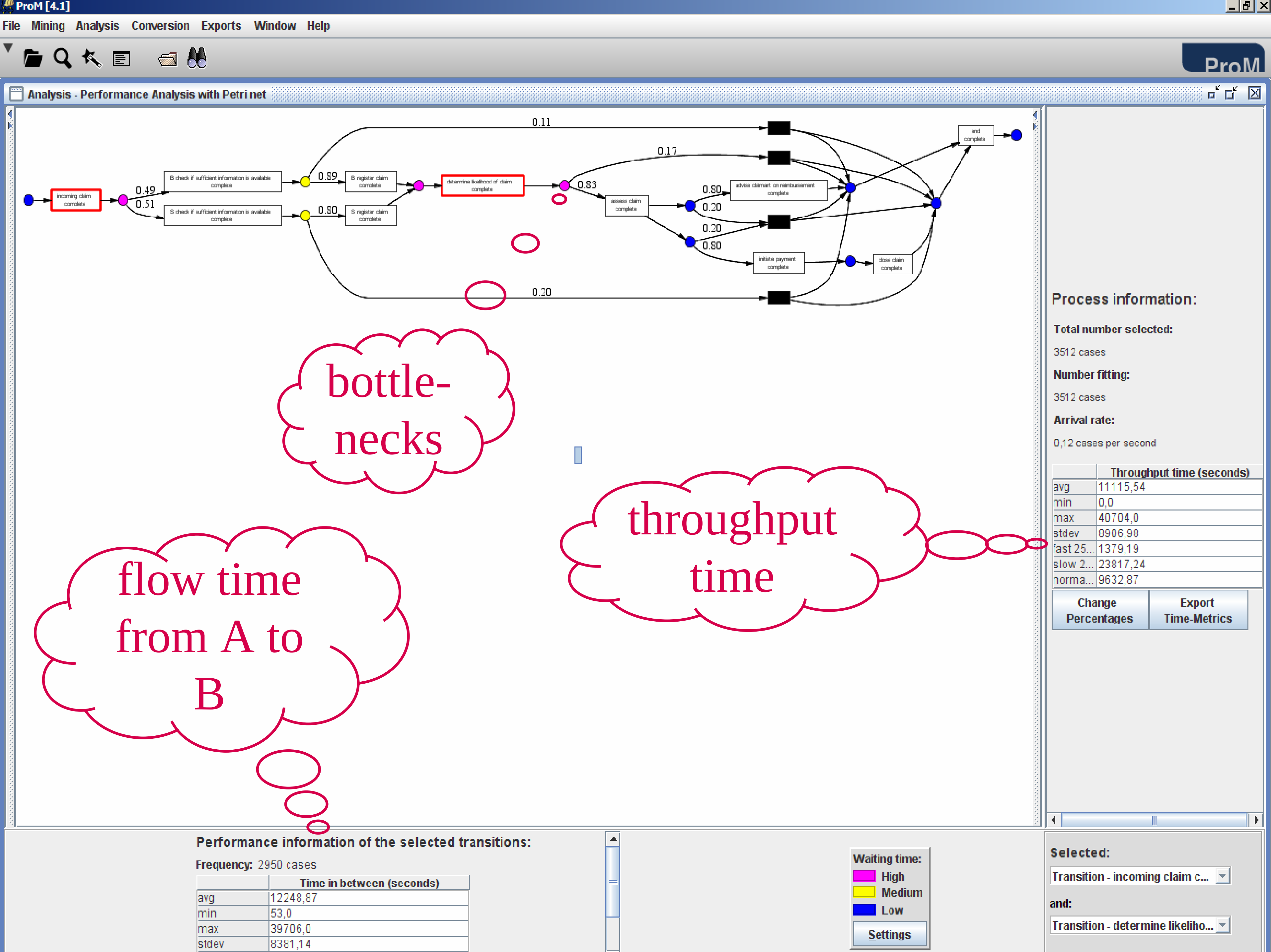
ProM supports +40 types of model discovery!


ProM [4.2]
File Mining Analysis Conversion Exports Window Help
ProM
Analysis - Dotted Chart Analysis
Dotted Chart
Settings
Options
Component type:
Instance ID
Time option:
Relative(Time)
Relative Time option:
Instance ID
Time sort (chart):
years
Color By:
Task ID
Shape By:
None
Mouse Mode:
Zoom in
(10^3)x
(10^2)x
10x
1x
zoom (X)
zoom (Y)
Zoom out
0days:0:0:0
365days:0:0:0
730days:0:0:0
1095days:0:0:0
1460days:0:0:0
1825days:0:0:0
2190days:0:0:0
2555days:0:0:0
2920days:0:0:0
3285days:0:0:0
3650days:0:0:0
4015days:0:0:0
49 | initiate payment | Claims handler | Tue Jun 25 01:00:00 CEST 2013 | complete | Position:2.033856E8
49 | close claim | Claims handler | Tue Jun 25 01:00:00 CEST 2013 | start | Position:2.033856E8
49 | close claim | Claims handler | Wed Jul 03 01:00:00 CEST 2013 | complete | Position:2.040768E8
49 | end | Claims handler | Wed Jul 03 01:00:00 CEST 2013 | start | Position:2.040768E8
49 | end | Claims handler | Wed Jul 03 01:00:00 CEST 2013 | complete | Position:2.040768E8
Time sort (metrics):
seconds
Component Overall:
# of components: 132
items values
time(first) 0days:0:0:0
time(end) 4462days:0:0:0
avg spread 96612872,72727
min spread 691200,0
max spread 385516800,0
Component 35:
# of dots: 16
items values
time(first) 0days:0:0:0
time(end) 1016days:0:0:0
avg. interval 5852160,0
min interval 0,0
max interval 47433600,0
Component 36:
# of dots: 16
items values
time(first) 0days:0:0:0
time(end) 1515days:0:0:0
avg. interval 8726400,0
min interval 0,0
max interval 77760000,0
Component 33:
# of dots: 4
items values
time(first) 0days:0:0:0
time(end) 12days:0:0:0
avg. interval 345600,0
min interval 0,0
max interval 1036800,0
Component 34:
# of dots: 16
items values
time(first) 0days:0:0:0
time(end) 2482days:0:0:0
avg. interval 14296320,0
min interval 0,0
max interval 146361600,0
Component 39:
# of dots: 16
items values
time(first) 0days:0:0:0
time(end) 537days:0:0:0
21:59:19 [D] Buffered log reader created from reader buffered log reader, pitk.: [l@b8dfdf

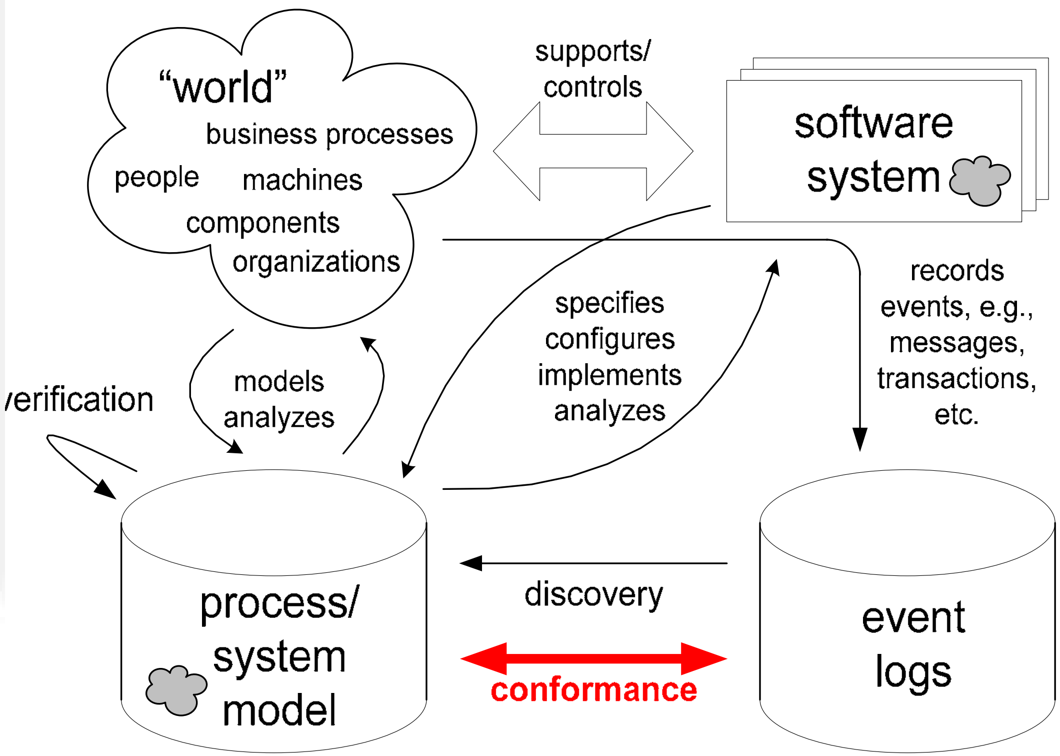

ProM [4.1]
File Mining Analysis Conversion Exports Window Help
Results - Settings for mining Filtered cpnToolsSimulationLog.mxml using Heuristics miner
incoming claim (complete) 3512
Conversion - Heuristic net to Petri net
incoming claim complete
B check if sufficient information is available complete
S check if sufficient information is available complete
B register claim complete
S register claim complete
determine likelihood of claim complete
assess claim complete
Conversion - Heuristic net to EPC
Status change to S register claim complete
Conversion - Labeled WF net to YAWL model
incoming claim complete Task
B check if sufficient information is available complete Task
S check if sufficient information is available complete Task
B register claim complete Task
S register claim complete Task
determine likelihood of claim complete Task
assess claim complete Task
advise claimant on reimbursement complete Task
initiate payment complete Task
close claim complete Task
end complete Task
[default]
true [0]
true [1]
ProM

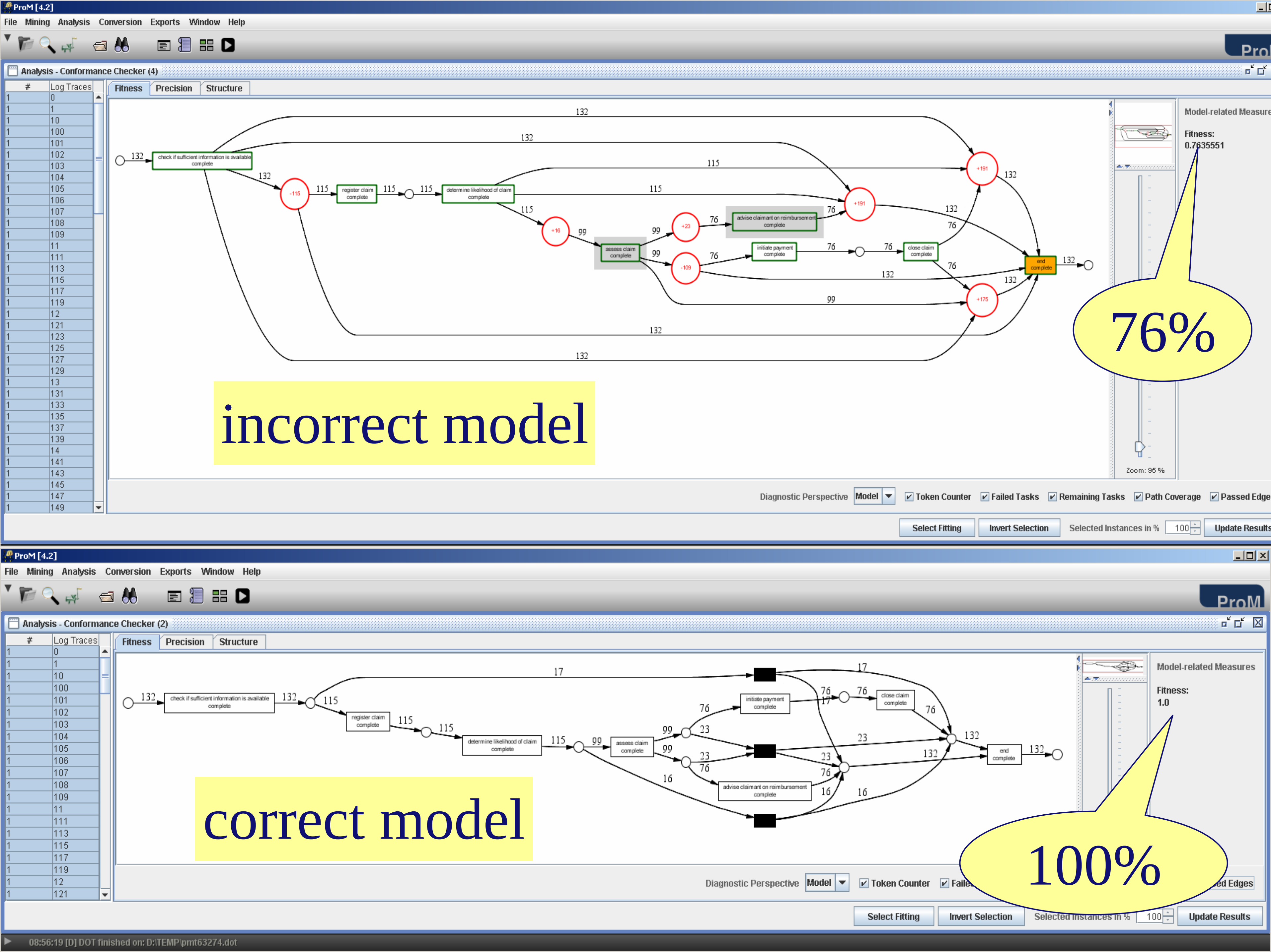


ProM [4.1]
File Mining Analysis Conversion Exports Window Help
ProM
Results - Settings for mining Filtered cpnToolsSimulationLog.mxml using Social network miner
Call Centre ... Call Centre ... Claims han... customer
Call Centre... 0.0911741... 0.0 0.0808375... 0.0
Call Centre... 0.0 0.0949907... 0.0755367... 0.0
Claims ha... 0.0 0.0 0.4712960... 0.0
customer 0.0911741... 0.0949907... 0.0 0.0
threshold : 0
Remove isolated nodes
0,091
Call Centre Agent Brisbane
0,471
0,091
0,081
customer
Claims handler
0,095
0,095
0,076
Call Centre Agent Sydney
Zoom
Analysis - Analyze Social Network
Show role nodes
Show org unit nodes
Vertex size
Size property: Frequency
Vertex degree ratio stretch
Edge weight
Layout
KKLayout
Mouse Mode
Transforming
Group Clusters
Weight
Betweenness
Edges removed for clusters: 0
Claims handler
0,091
0,076
Call Centre Agent Brisbane
Call Centre Agent Sydney
0,091
0,095
customer


ProM [4.1]
File Mining Analysis Conversion Exports Window Help
ProM
Analysis - Performance Analysis with Petri net
0.11
0.17
0.49
0.51
0.89
0.80
0.83
0.80
0.20
0.20
0.80
0.20
bottle-necks
throughput time
flow time from A to B
Process information:
Total number selected:
3512 cases
Number fitting:
3512 cases
Arrival rate:
0,12 cases per second
Throughput time (seconds)
avg 11115,54
min 0,0
max 40704,0
stdev 8906,98
fast 25... 1379,19
slow 2... 23817,24
norma... 9632,87
Change Percentages
Export Time-Metrics
Performance information of the selected transitions:
Frequency: 2950 cases
Time in between (seconds)
avg 12248,87
min 53,0
max 39706,0
stdev 8381,14
Waiting time:
High
Medium
Low
Settings
Selected:
Transition - incoming claim c...
and:
Transition - determine likeliho...

Conformance Checking





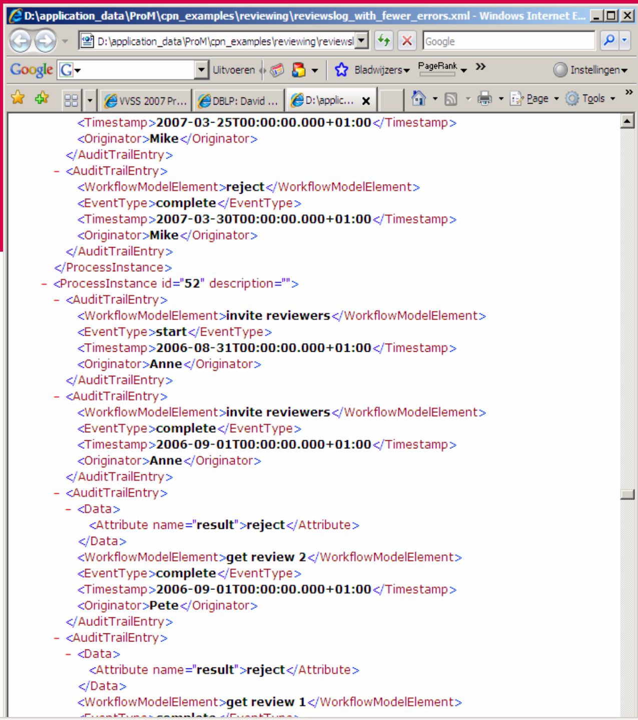
ProM [4.2]
Analysis - Conformance Checker (4)
Fitness
Precision
Structure
Model-related Measures
Fitness:
0.7635551
incorrect model
76%
Diagnostic Perspective
Token Counter
Failed Tasks
Remaining Tasks
Path Coverage
Passed Edges
Select Fitting
Invert Selection
Selected Instances in %
Update Results
ProM [4.2]
Analysis - Conformance Checker (2)
Fitness:
1.0
correct model
100%
08:56:19 [D] DOT finished on: D:\TEMP\pmt63274.dot


ProM [4.2]
Exposé ...select the frame you want to bring forward
Results - Fuzzy Miner on Selected EPC
Fuzzy Miner
Graph
Unary metrics
Binary metrics
Concurrency filter
Edge filter
Node filter
Significance cutoff
Results - Multi-phase Macro Plugin on Selected Petri net
Zoom: 50 %
ProM
+/- 250 plug-ins
Results - Alpha algorithm plugin on Raw reviewslog_with_fewer_errors.xml (unfiltered)
Zoom: 66 %
Edit log relations
reviewslog_with_fewer_errors.xml
Dashboard
Key data
Processes 1
Cases 100
Events 2297
Event classes 20
Event types 2
Originators 10
Events per case
Min 11
Mean 22
Max 50
Event classes per case
Min 11
Mean 14
Max 15
Log info
Source
CPN Tools
Source program
CPN Tools
Start date
2006-01-01 00:00:00
End date
2008-05-05 01:00:00
Description
Log extracted from CPN Tools
start analyzing this log

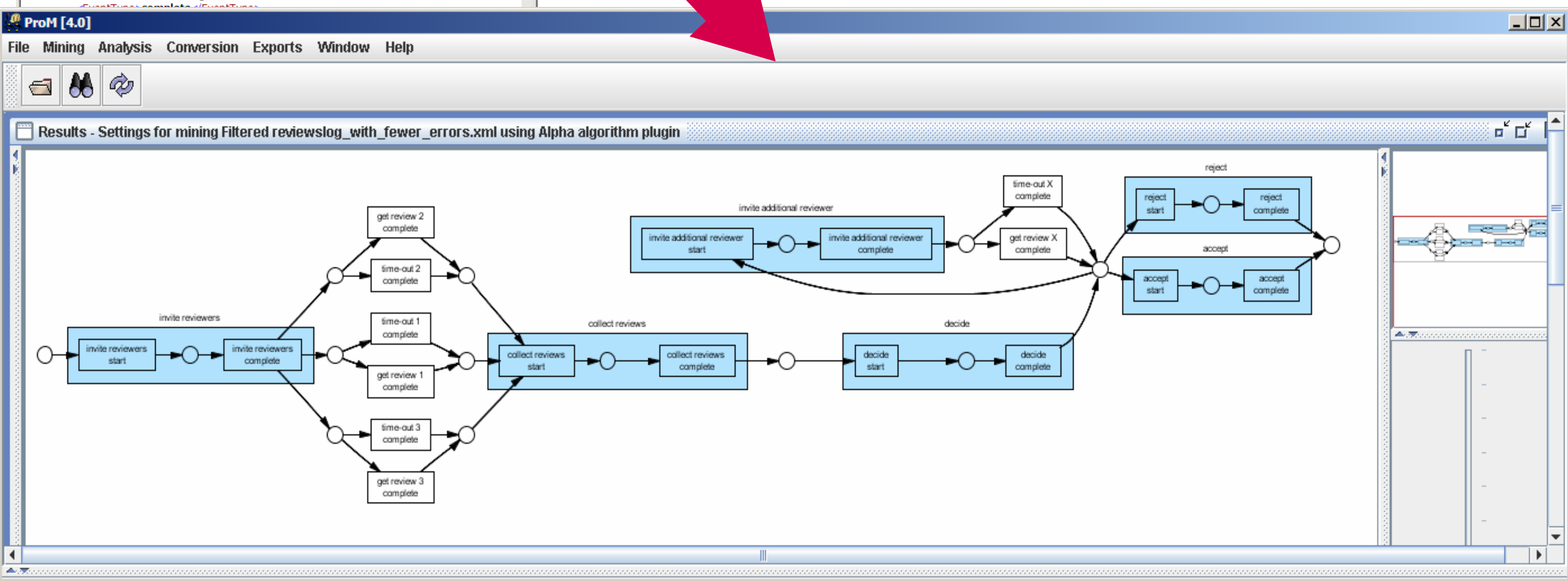
Process Discovery:
The Basics

Technische Universiteit
Eindhoven
University of Technology

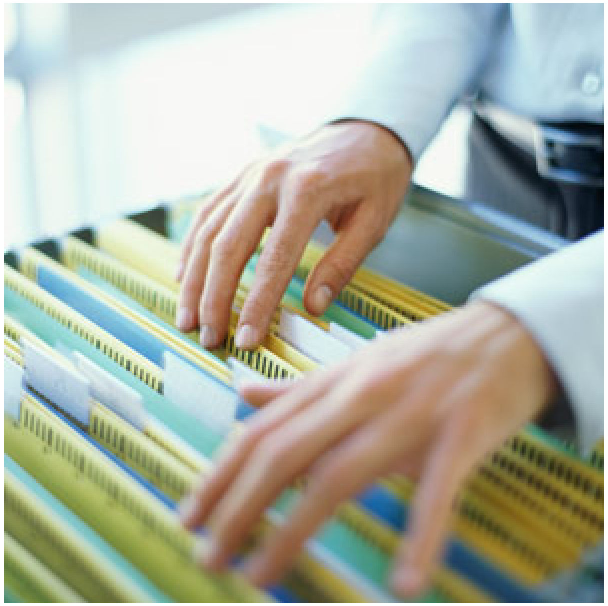
Where innovation starts



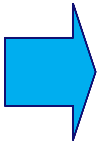
Alpha algorithm

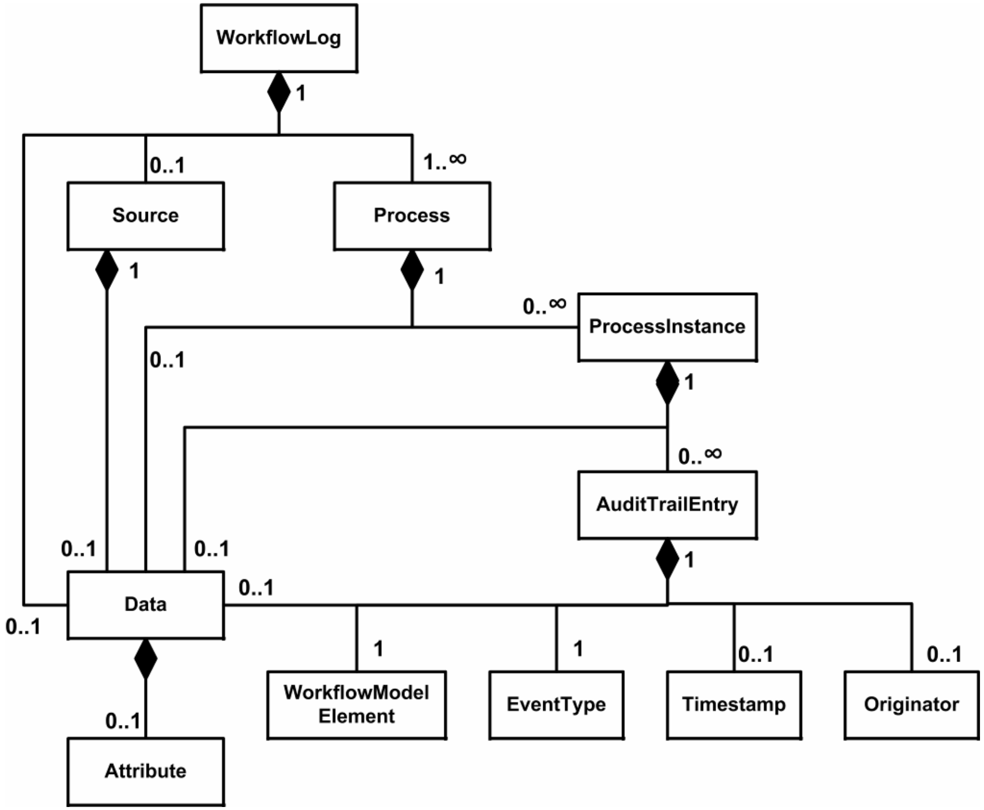


Starting point: An event log



event logs, audit trails, databases, message logs, etc.





unified event log (MXML)

Example log

- Minimal information in log: case id's and task id's.
- Additional information: event type, time, resources, and data.
- Sequences:
  - 1: ABCD
  - 2: ACBD
  - 3: ABCD
  - 4: ACBD
  - 5: EF
- So this log there are three possible sequences:
  - ABCD
  - ACBD
  - EF

```
case 1 : task A
case 2 : task A
case 3 : task A
case 3 : task B
case 1 : task B
case 1 : task C
case 2 : task C
case 4 : task A
case 2 : task B
case 2 : task D
case 5 : task E
case 4 : task C
case 1 : task D
case 3 : task C
case 3 : task D
case 4 : task B
case 5 : task F
case 4 : task D
```

>,→,||,# relations

- Direct succession: **x>y iff for some case x is directly followed by y.**
- Causality: **x→y iff x>y and not y>x.**
- Parallel: **x||y iff x>y and y>x**
- Choice: **x#y iff not x>y and not y>x.**

```
case 1 : task A
case 2 : task A
case 3 : task A
case 3 : task B
case 1 : task B
case 1 : task C
case 2 : task C
case 4 : task A
case 2 : task B
case 2 : task D
case 5 : task E
case 4 : task C
case 1 : task D
case 3 : task C
case 3 : task D
case 4 : task B
case 5 : task F
case 4 : task D
```

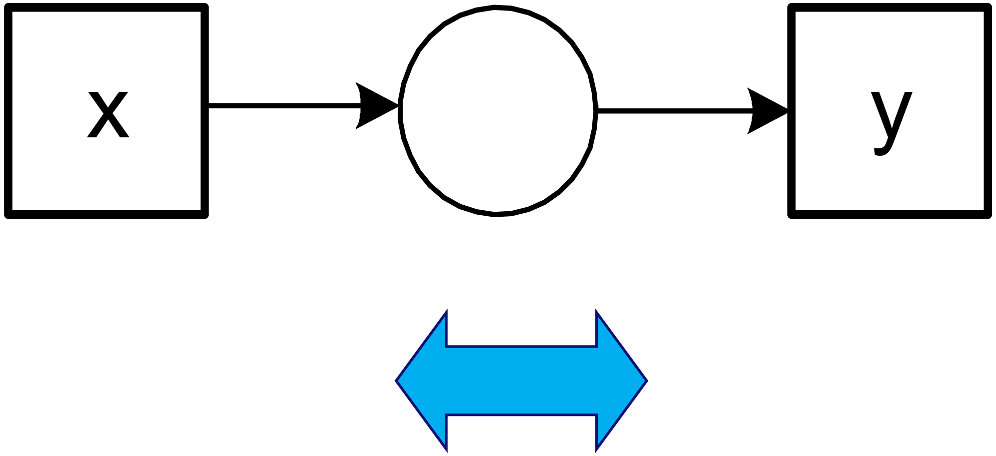
ABCD
ACBD
EF

A>B
A>C
B>C
B>D
C>B
C>D
E>F

A→B
A→C
B→D
C→D
E→F

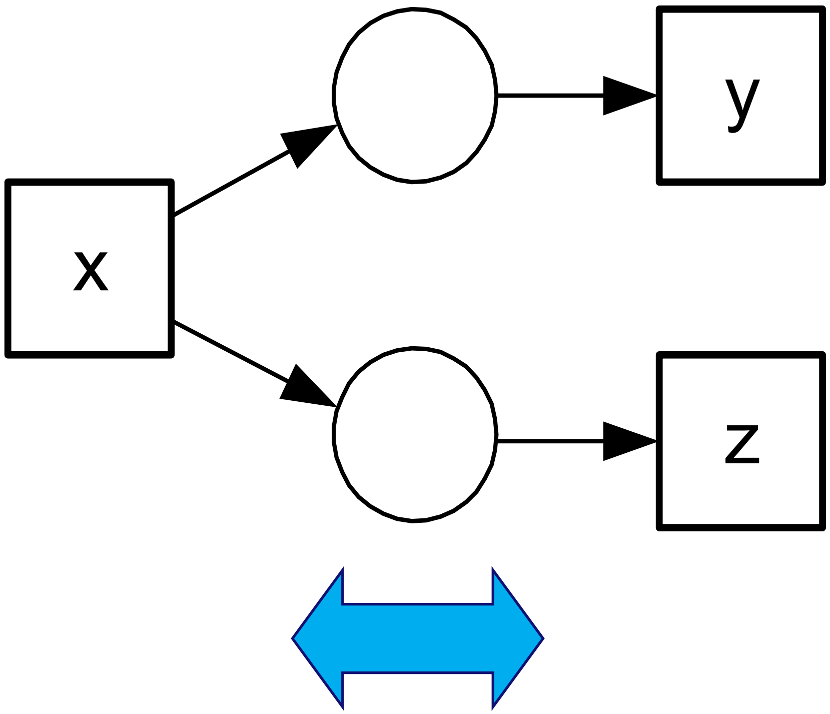
B||C
C||B

Basic idea (1)



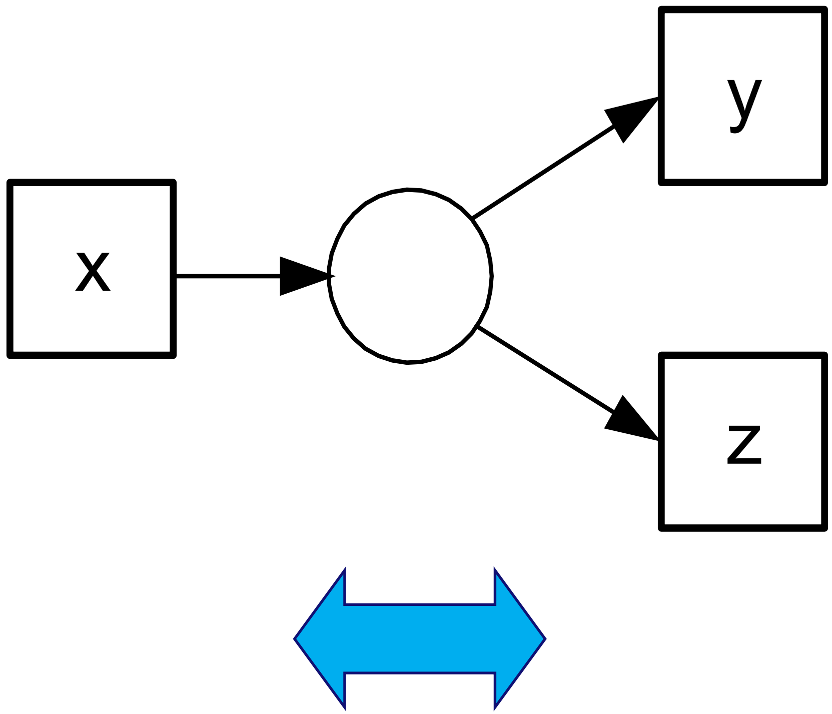
$$x \rightarrow y$$

Basic idea (2)



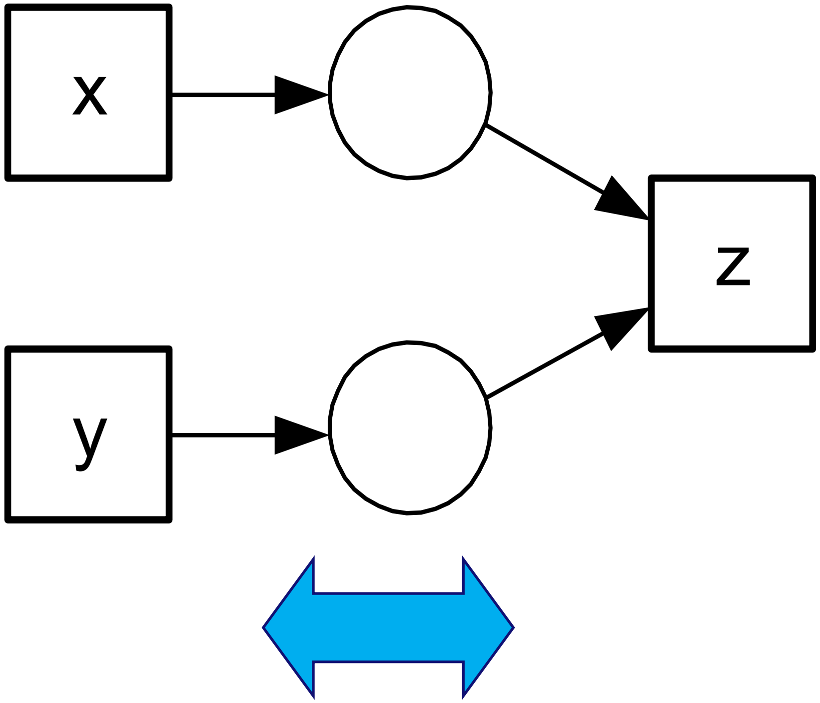
$x \rightarrow y$, $x \rightarrow z$, and $y || z$

Basic idea (3)



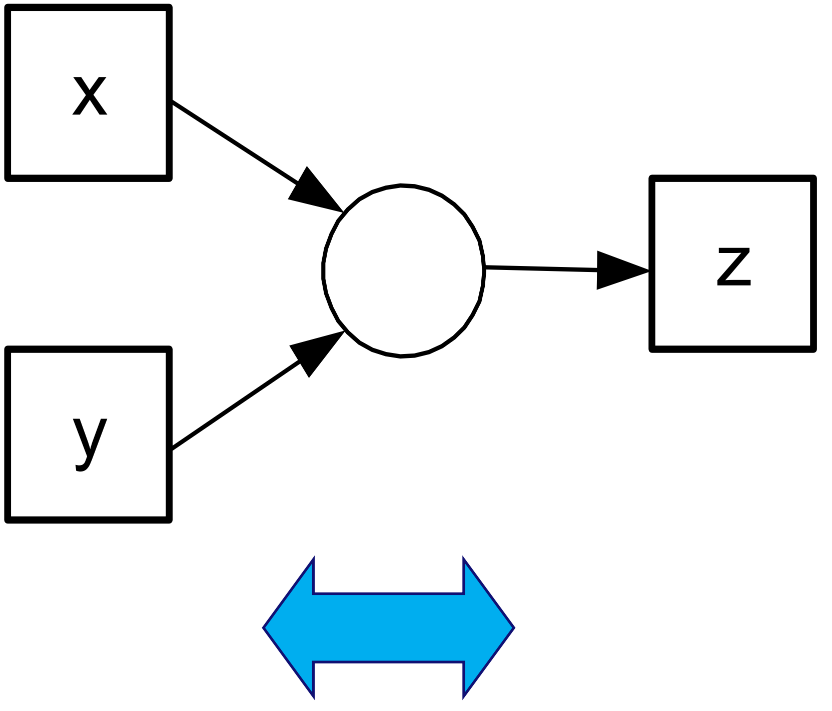
x→y, x→z, and y#z

Basic idea (4)



$x \rightarrow z$, $y \rightarrow z$, and $x||y$

Basic idea (5)



$x \to z$, $y \to z$, and x#y

It is not that simple: Basic alpha algorithm

Let W be a workflow log over T. $\alpha(W)$ is defined as follows.

1. $T_W = \{ t \in T \mid \exists_{\sigma \in W}\, t \in \sigma \}$,
2. $T_I = \{ t \in T \mid \exists_{\sigma \in W}\, t = \mathit{first}(\sigma) \}$,
3. $T_O = \{ t \in T \mid \exists_{\sigma \in W}\, t = \mathit{last}(\sigma) \}$,
4. $X_W = \{ (A,B) \mid A \subseteq T_W \wedge A \neq \emptyset \wedge B \subseteq T_W \wedge B \neq \emptyset \wedge$
   $\forall_{a \in A} \forall_{b \in B}\, a \rightarrow_W b \wedge \forall_{a1,a2 \in A}\, a_1 \#_W a_2 \wedge \forall_{b1,b2 \in B}\, b_1 \#_W b_2 \}$,
5. $Y_W = \{ (A,B) \in X \mid \forall_{(A',B') \in X} A \subseteq A' \wedge B \subseteq B' \Rightarrow (A,B) = (A',B') \}$,
6. $P_W = \{ p_{(A,B)} \mid (A,B) \in Y_W \} \cup \{i_W, o_W\}$,
7. $F_W = \{ (a, p_{(A,B)}) \mid (A,B) \in Y_W \wedge a \in A \} \cup \{ (p_{(A,B)}, b) \mid (A,B) \in$
   $Y_W \wedge b \in B \} \cup \{ (i_W, t) \mid t \in T_I \} \cup \{ (t, o_W) \mid t \in T_O \}$, and
8. $\alpha(W) = (P_W, T_W, F_W)$.

Example revisited

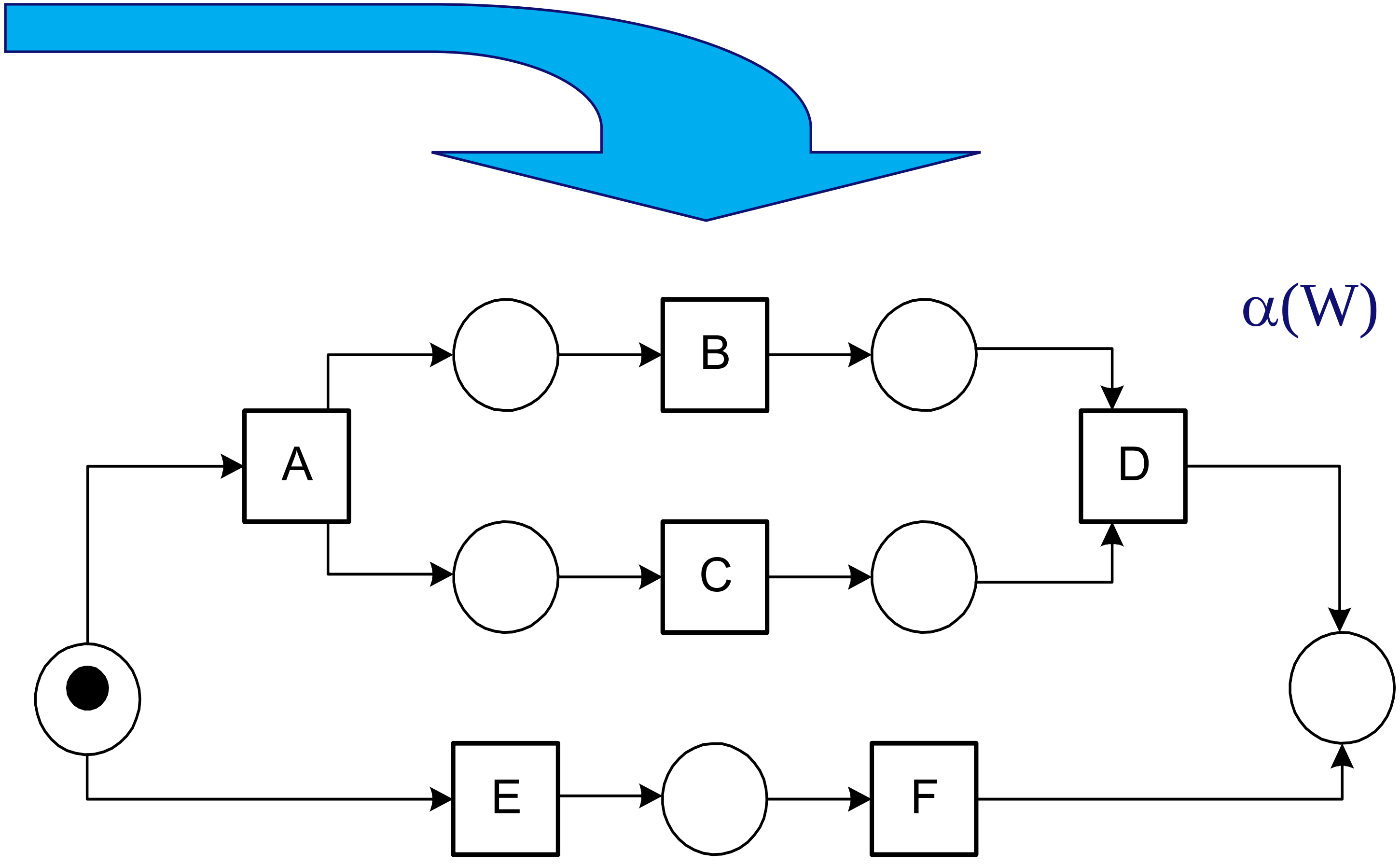
W:

```
case 1 : task A
case 2 : task A
case 3 : task A
case 3 : task B
case 1 : task B
case 1 : task C
case 2 : task C
case 4 : task A
case 2 : task B
case 2 : task D
case 5 : task E
case 4 : task C
case 1 : task D
case 3 : task C
case 3 : task D
case 4 : task B
case 5 : task F
case 4 : task D
```

A>B
A>C
B>C
B>D
C>B
C>D
E>F

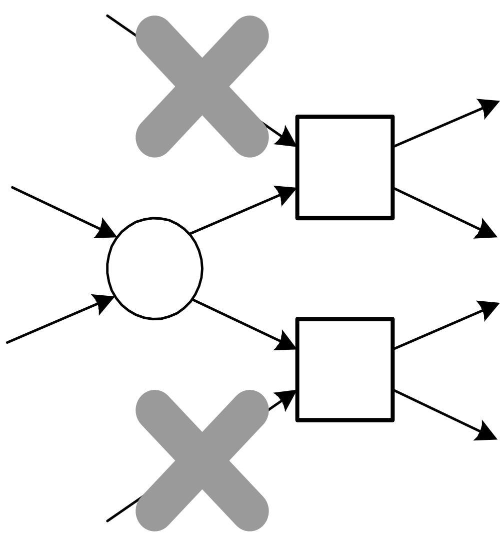
A→B
A→C
B→D
C→D
E→F

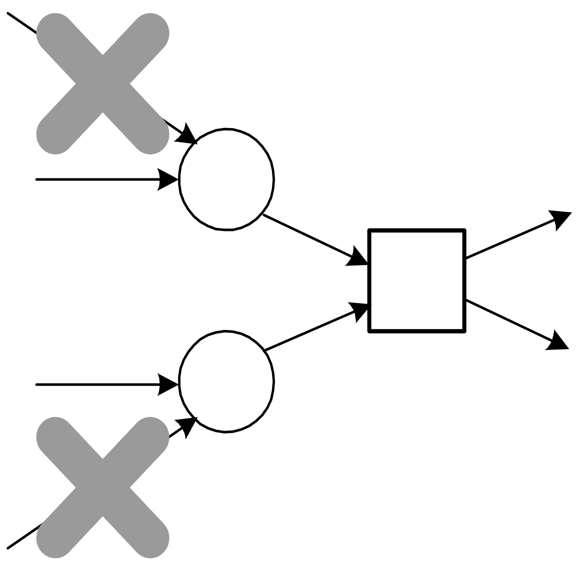
B||C
C||B



Properties of the Alpha algorithm

- **If log is complete with respect to relation >, it can be used to mine any SWF-net!**
- ***Structured Workflow Nets* (SWF-nets) have no implicit places and the following two constructs cannot be used:**

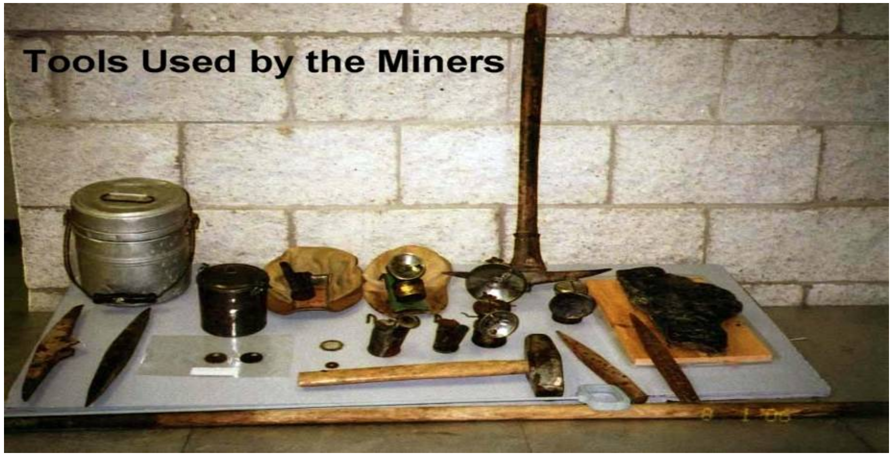




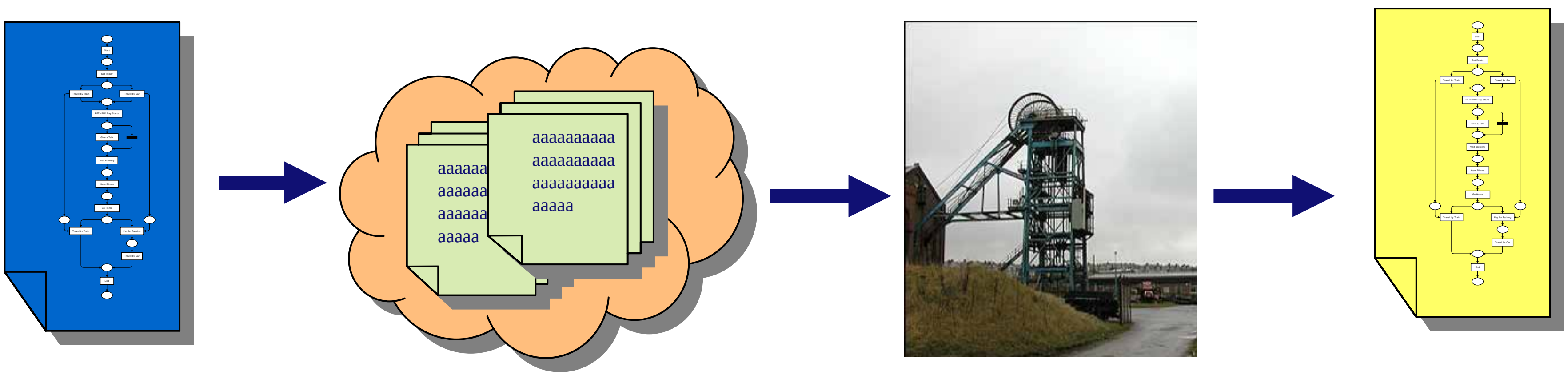
(Short loops require some refinement but not a problem.)

Alpha algorithm

- Mainly of theoretical interest!
- Too simple to be applicable to real-life logs.
- Does not address issues such as noise, etc.
- Should NOT be taken as a benchmark.
- However, the algorithm reveals:
  - basic process mining ideas and concepts in 8 lines,
  - theoretical limits of process mining.



Basic test for any mining algorithm: Rediscovery



Original Process → **Logs** → **Mining algortithm** → **Mined Process**

Can the mined process generate all the behavior in the log?

How close is the behavior of the mined process to the original one?

Controlled choices cannot be rediscovered (and in may cases this is good!)

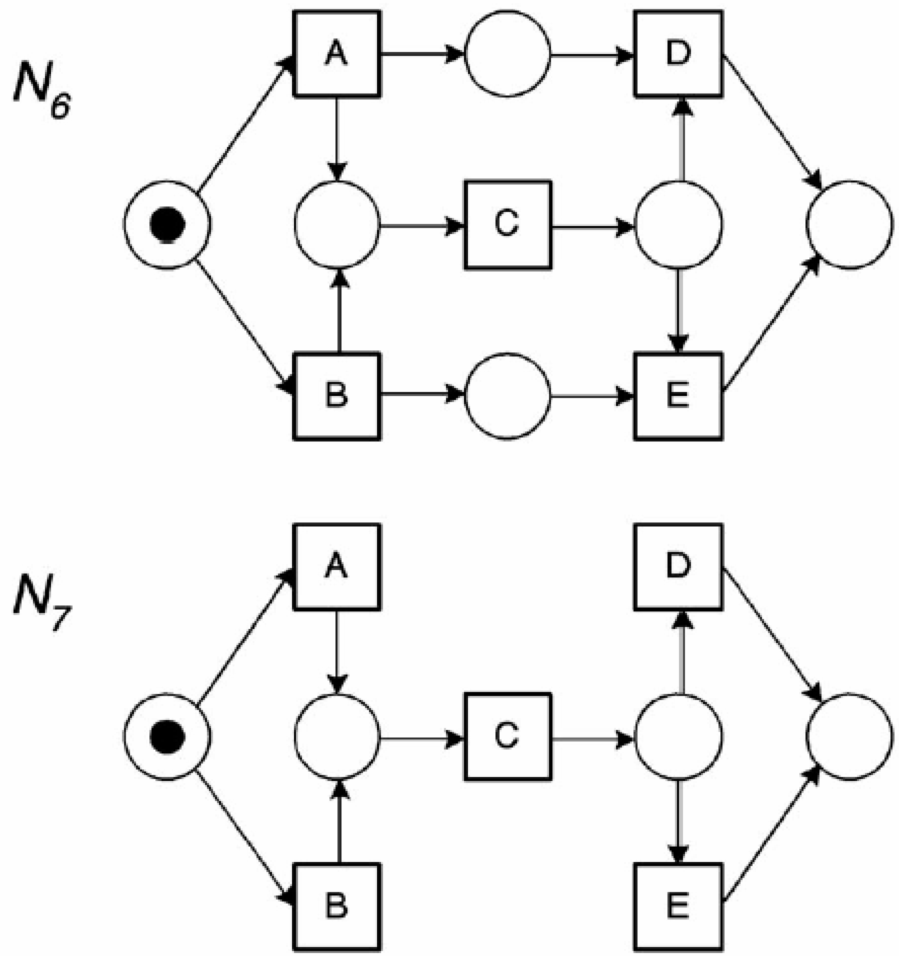


Fig. 7. The nonfree-choice WF-net N_6 cannot be rediscovered by the α algorithm.

Log only contains information about behavior and not structure

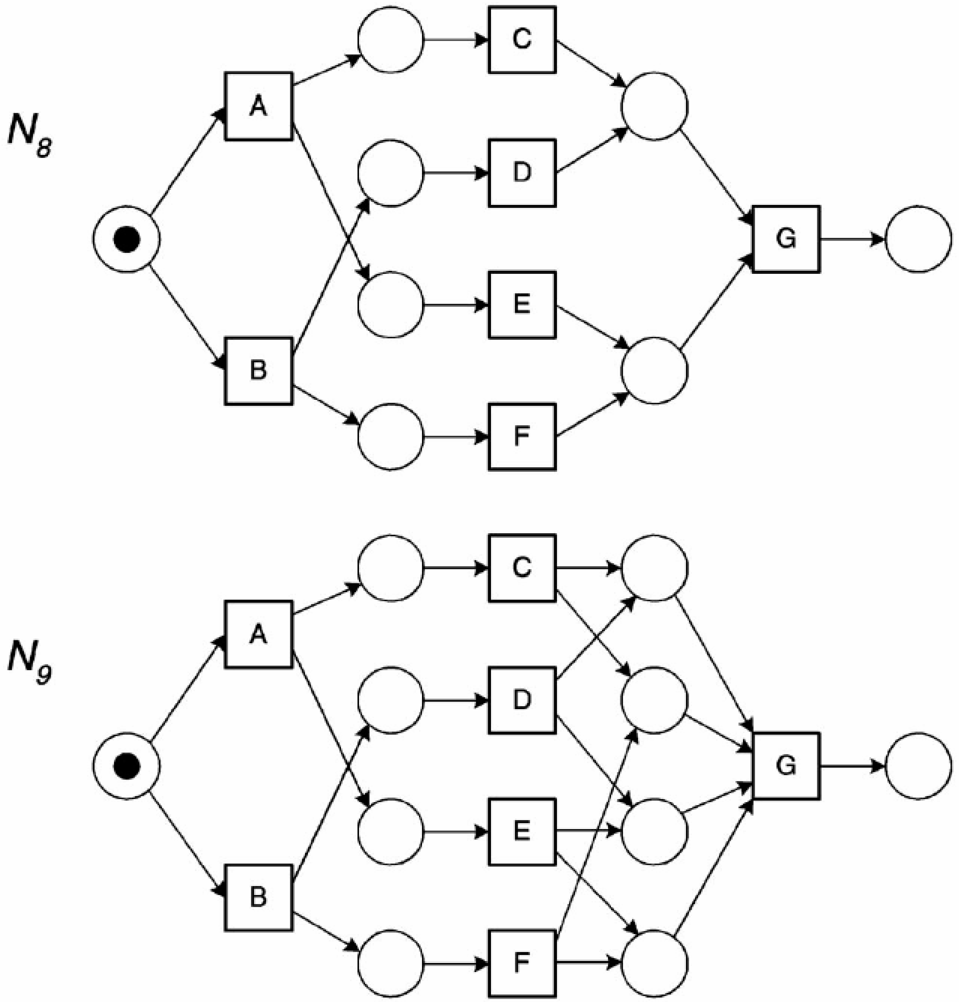


Fig. 8. WF-net N_8 cannot be rediscovered by the α algorithm. Nevertheless, α returns a WF-net which is behavioral equivalent.

Completeness notion may be too crude in some cases

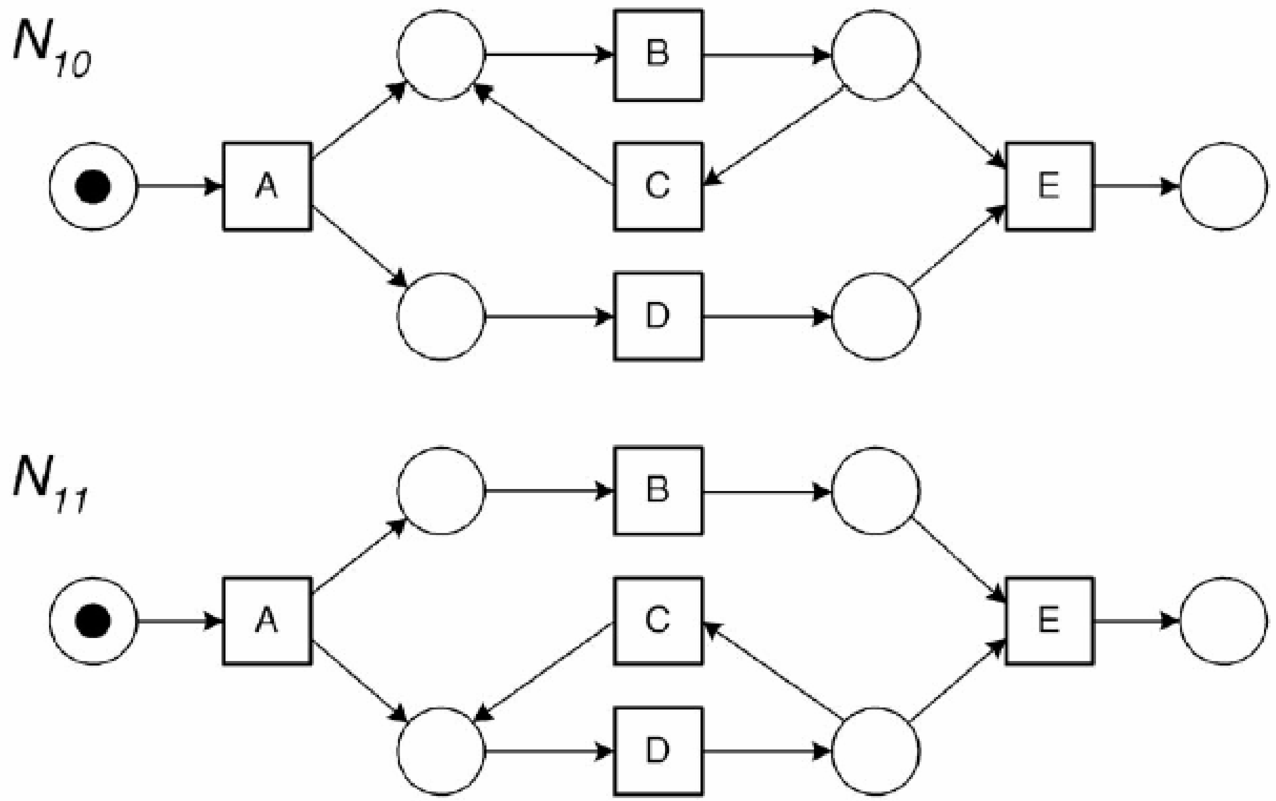


Fig. 9. Although both WF-nets are not behavioral equivalent they are identical with respect to $>$.

Another example of behaviorally equivalent SWF-nets

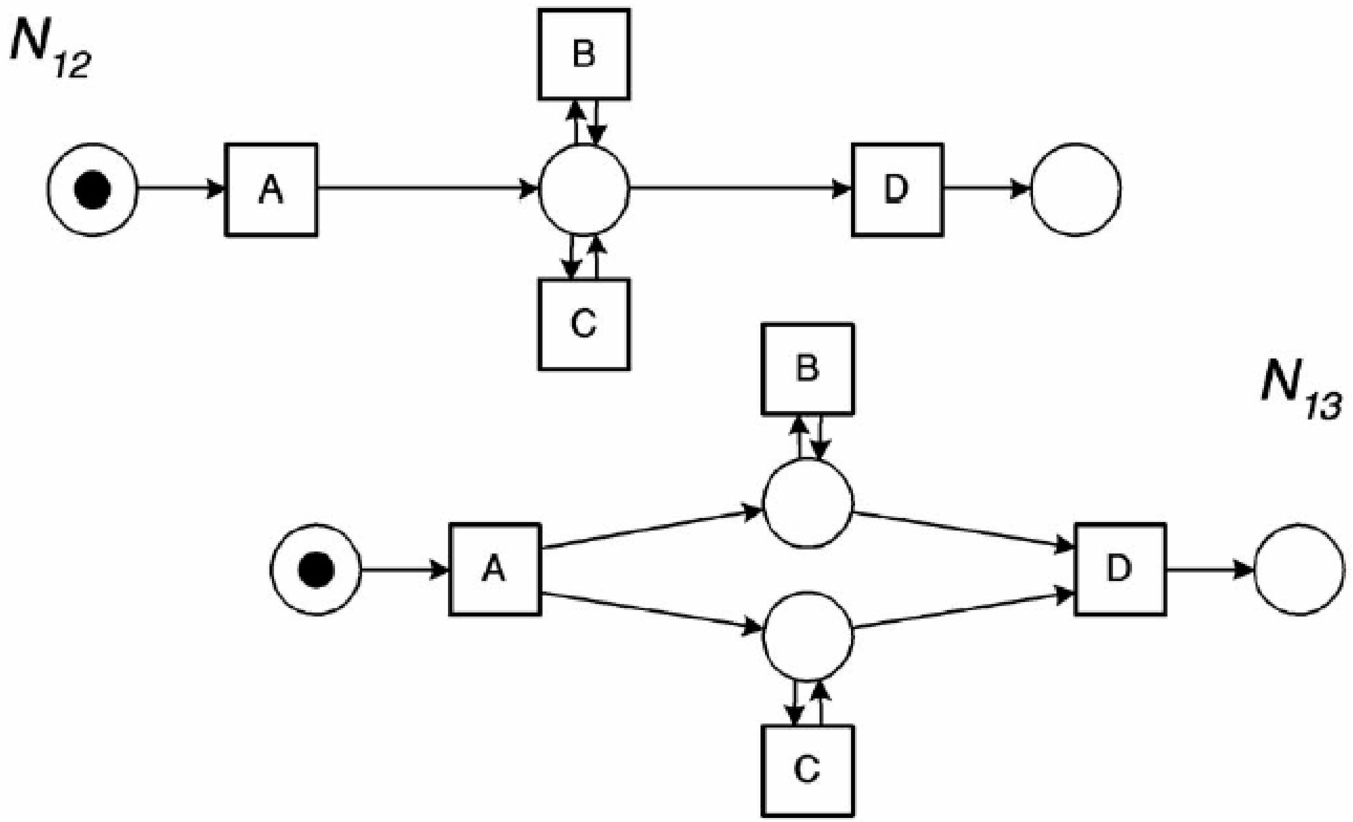
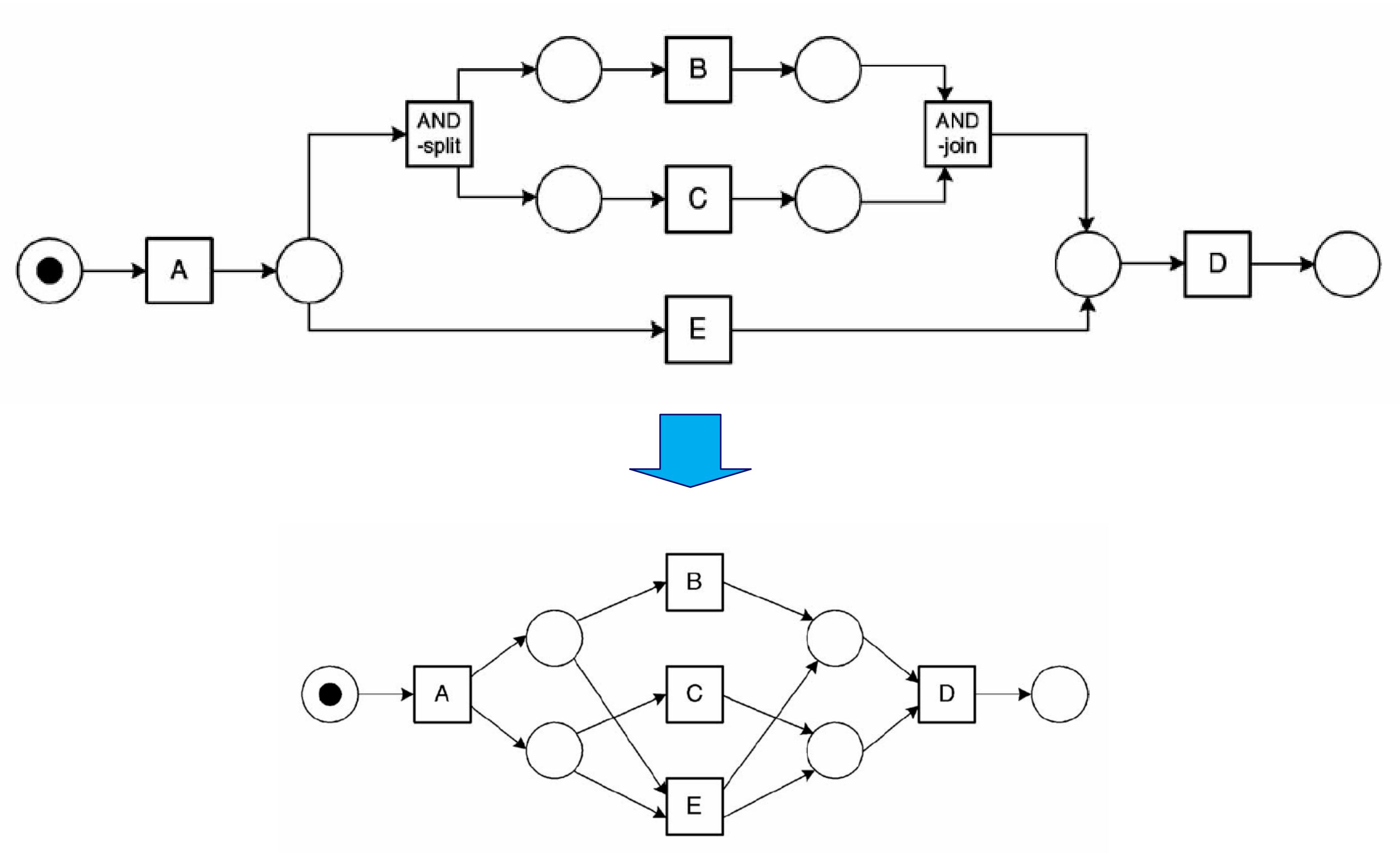
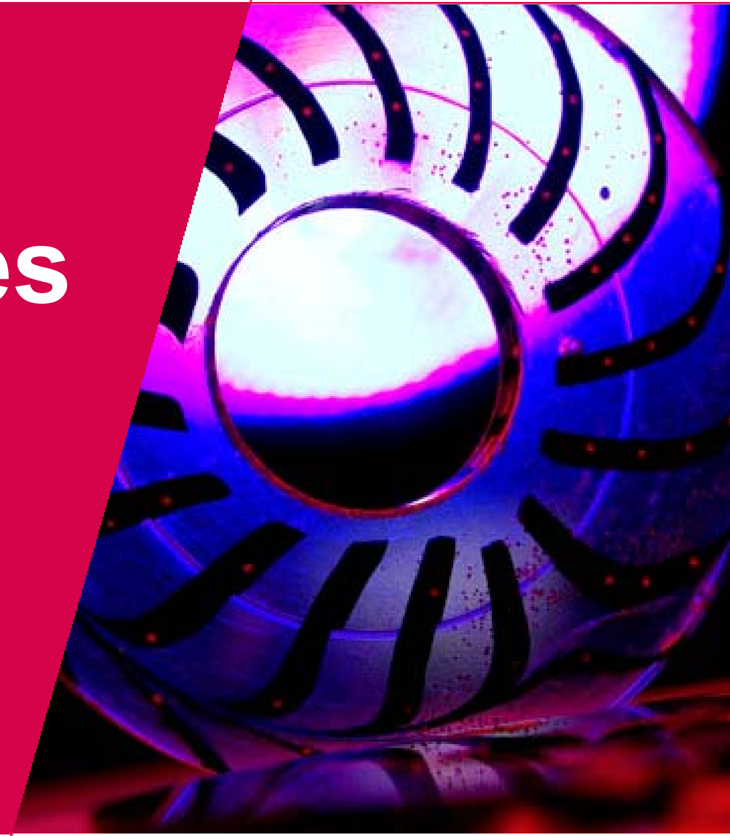


Fig. 10. Both SWF-nets are behavioral equivalent and, therefore, any algorithm will be unable to distinguish N_{12} from N_{13} (assuming a notion of completeness based on $>$).

Silent steps (and duplicate steps) cannot be discovered



Other Process Discovery Techniques



Technische Universiteit **Eindhoven** University of Technology

Where innovation starts

Overview of process discovery techniques

- **Classical techniques (e.g., learning state machines and the theory of regions): cannot handle concurrency and/or do not generalize (i.e., if it did not happen, it cannot happen).**
- **Algorithmic techniques**
  - **Alpha miner**
  - **Alpha+, Alpha++, Alpha#**
  - **Heuristic miner**
  - **Multi phase miner**
  - **...**
- **Genetic process mining**
- **Region-based process mining**
  - **State-based regions**
  - **Language based regions**

Multi-Phase Miner
(Boudewijn van Dongen et al.)

Two phases:

1) **Create a visual description of each instance, without choices and loops (cf. runs or occurrence nets).**
   - **Comprehensive representation**
   - **Ideal for performance analysis (cf. ARIS PPM)**
2) **Aggregate multiple instances to one process model.**
   - **Only causal relations between tasks are required**

Properties:

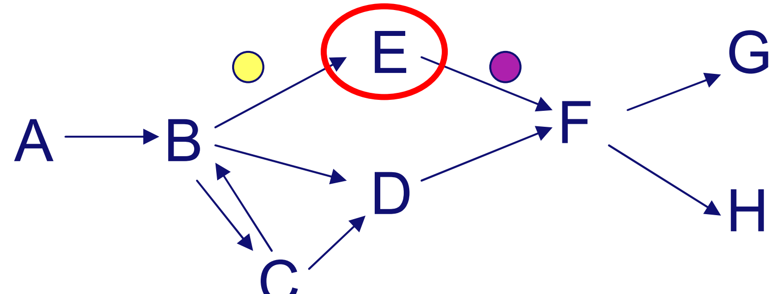
- **More robust and multi-lingual (cf. EPCs).**
- **Possibility of inspect instances**

Step 1: Create instance graphs

Log file:

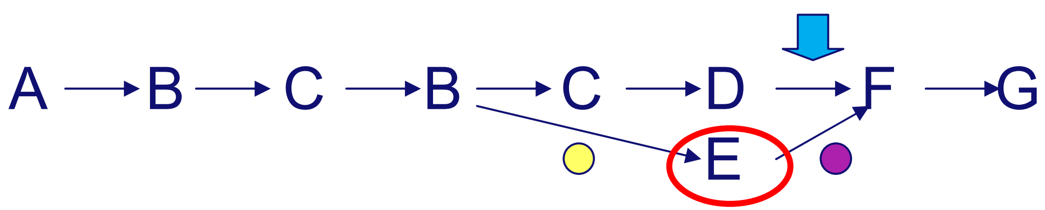
- **A, B, C, B, C, D, E, F, G**
- **A, B, E, D, F, G**
- **A, B, D, E, F, H**

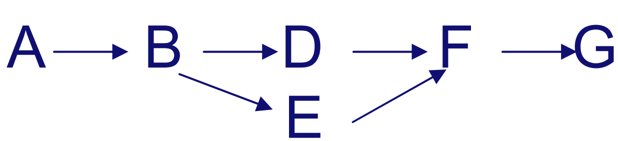
Causal relations:

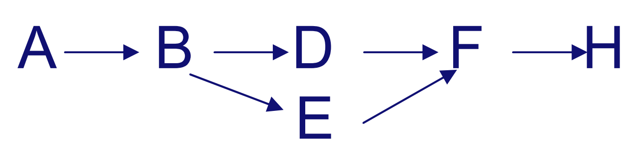


Similar to →in the alpha algorithm but refined further.

For each entry in every instance, find the closest causal predecessor and successor, and build instance graphs

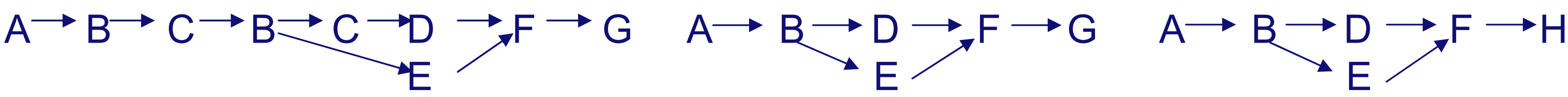




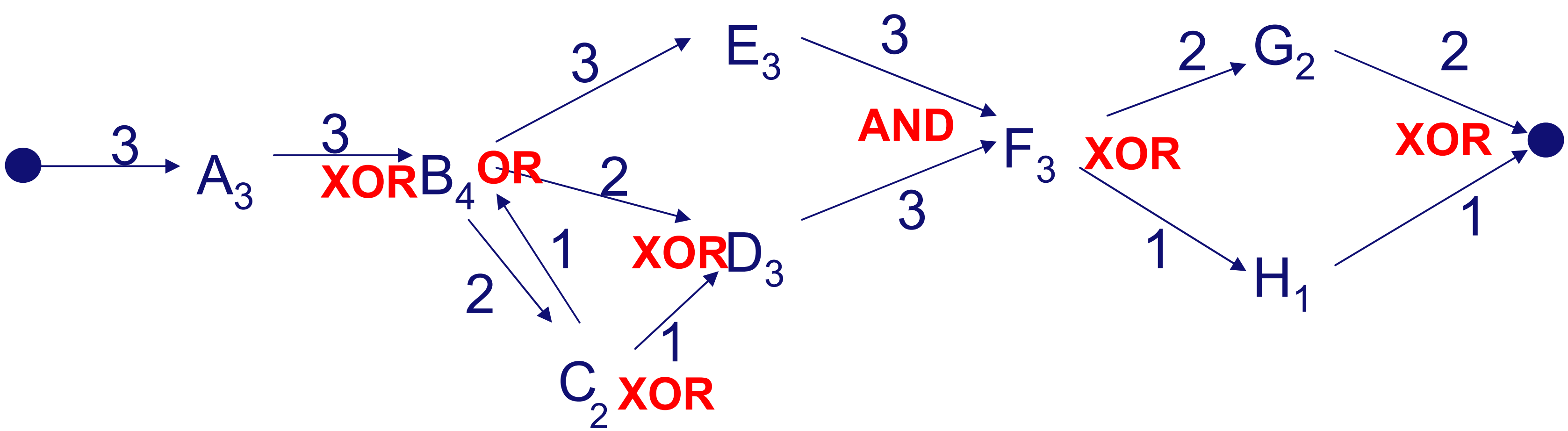


Step 2: Aggregate instance graphs

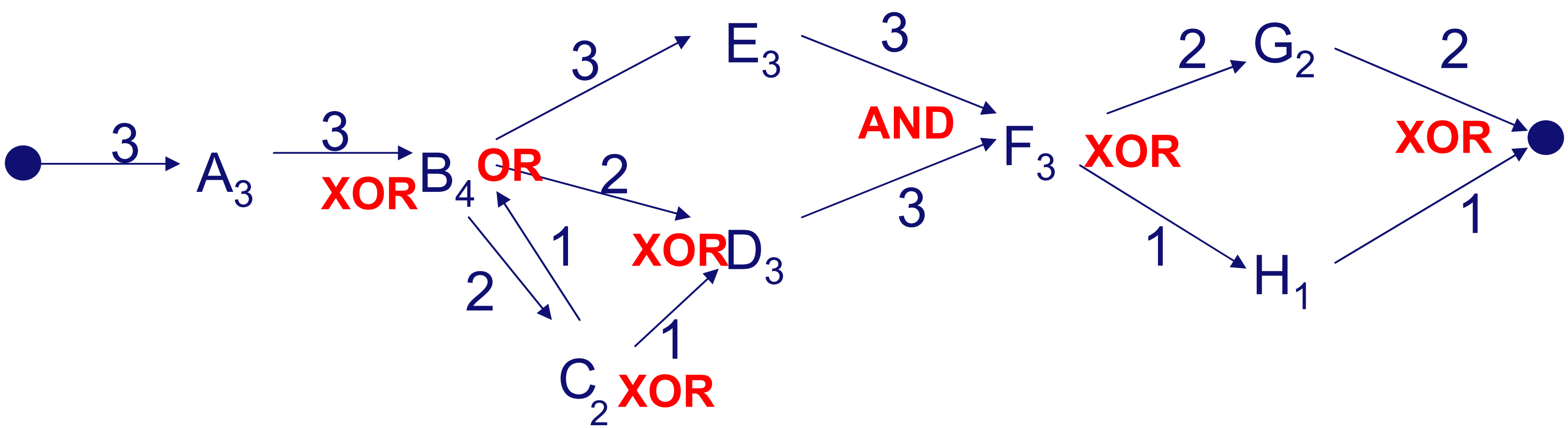
Three instance graphs:

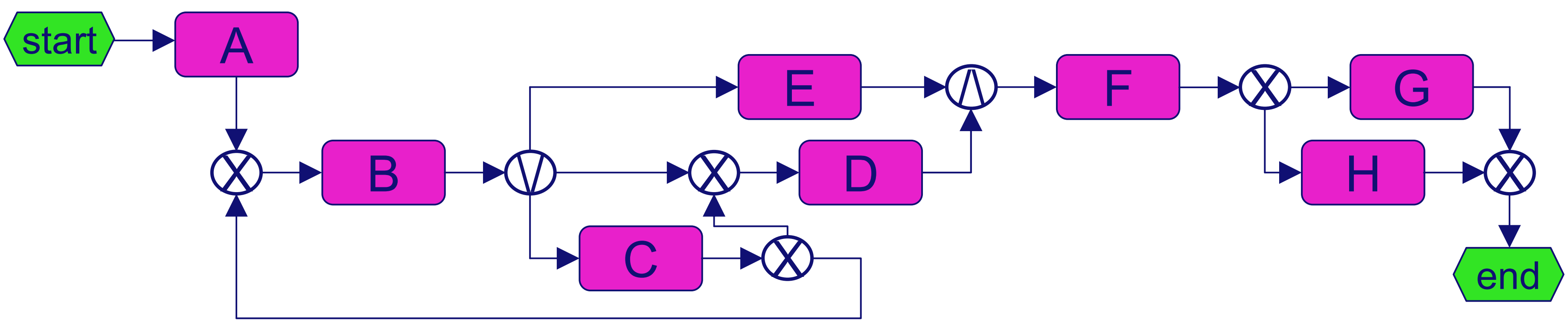


Aggregated instances:



Representation as an EPC



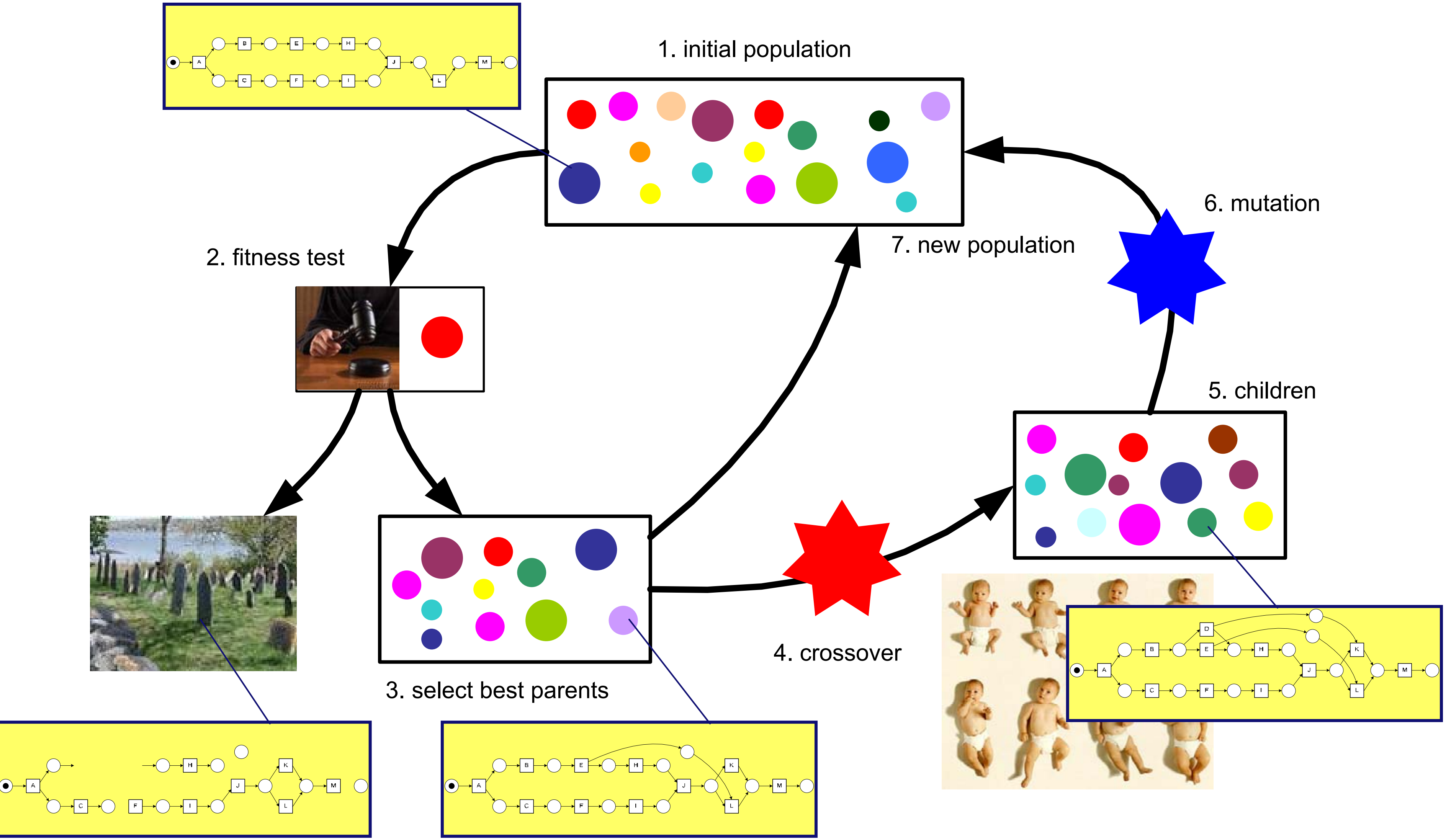


Any notation with OR-splits and OR-joins can be used, e.g., YAWL, BPMN, etc.

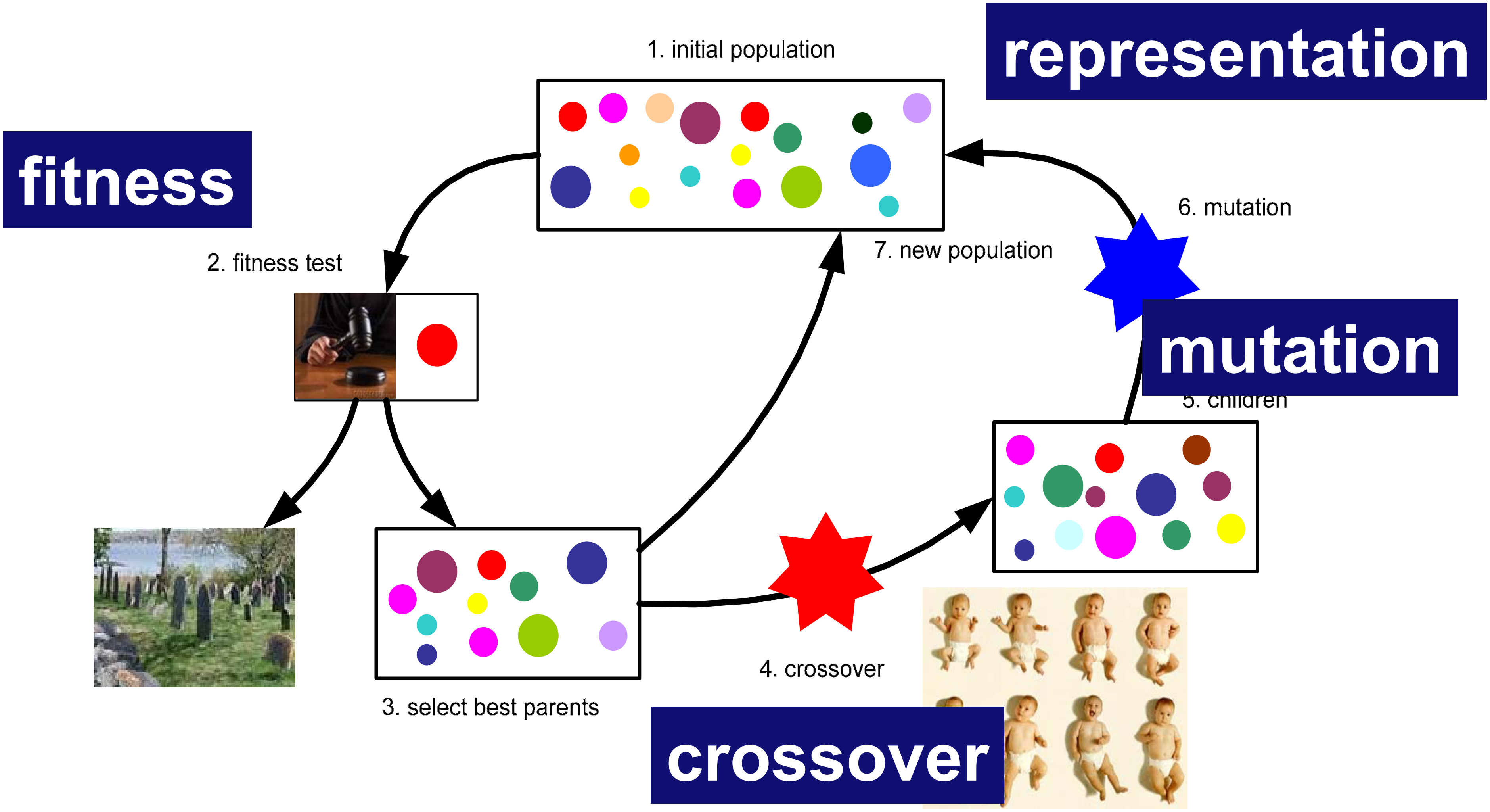
Properties of Multi-phase miner

- Always a fitness of 1, i.e., all traces in the logs may be reproduced (both potentially also many more).
- Very robust and fast, but tends to overgeneralize.
- Any subset of traces produces a meaningful result (event a single instance) that can be used for visualization purposes.
- No special provisions for noise or infrequent behavior.

Genetic Mining
(Ana Karla Alves de Medeiros et al.)

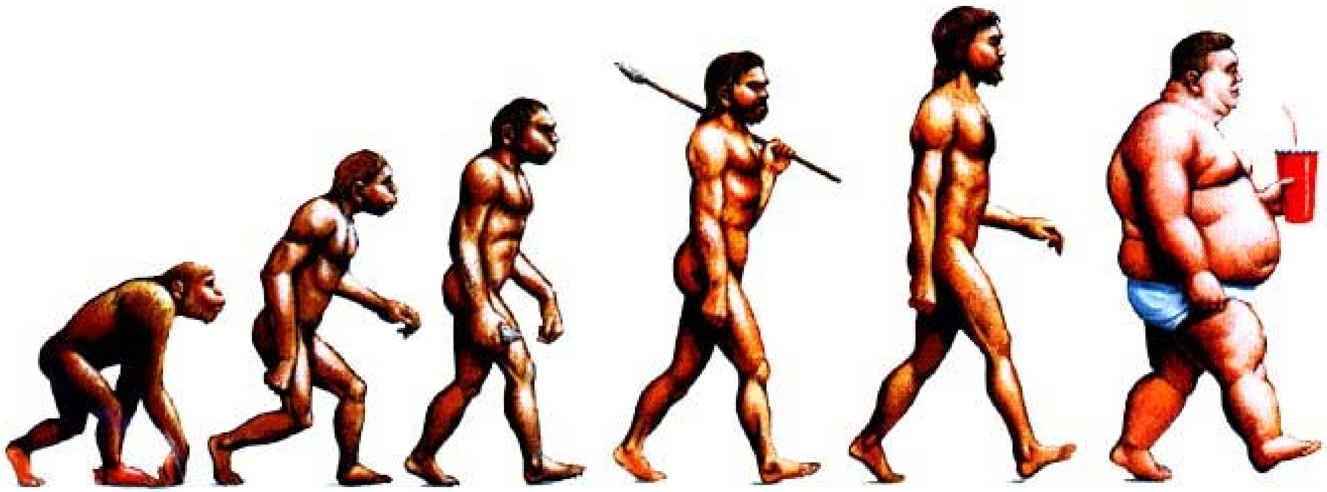


Design choices



Properties of Genetic Mining

- Requires a lot of computing power.
- Can deal with noise, infrequent behavior, duplicate tasks, invisible tasks, etc.
- Allows for incremental improvement and combinations with other approaches (heuristics post-optimization, etc.).

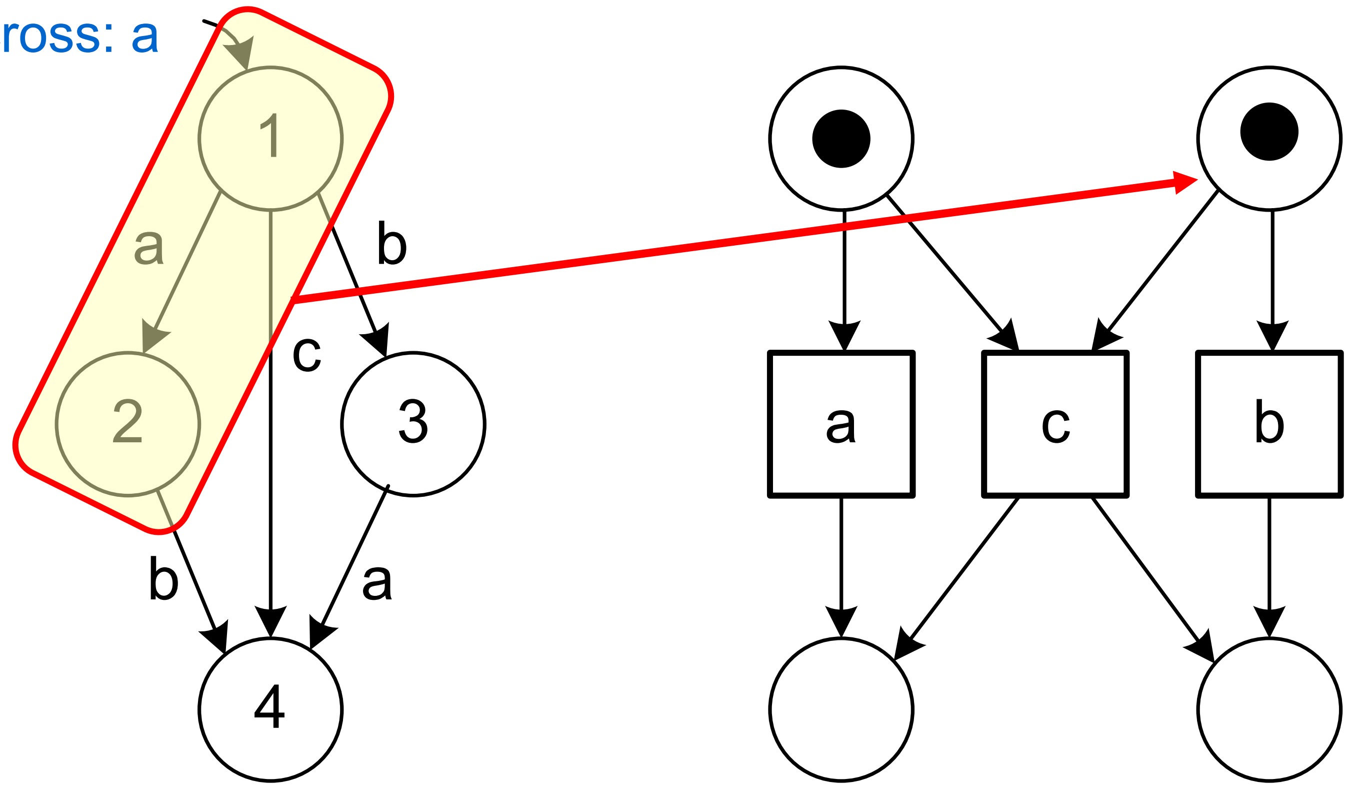


Region-Based Approaches

- Classical synthesis approaches to translate "behavior" into "models":
  - State-based regions
  - Language-based regions
- Synthesis ≠ process mining!
- Common issues:
  - Translating logs into transition systems (for state-based regions).
  - Overfitting.
  - Performance of algorithms and complexity of result.

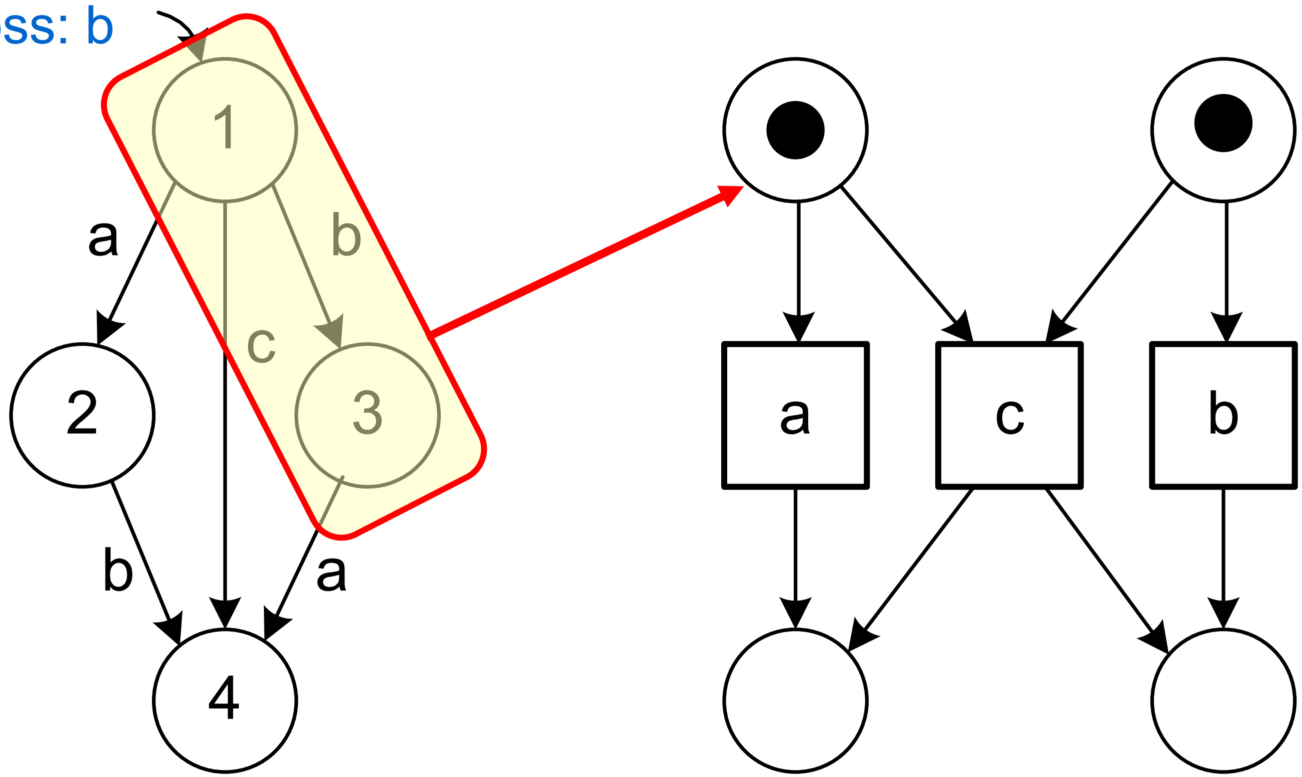
State-based regions

enter: -
exit: b,c
do not cross: a



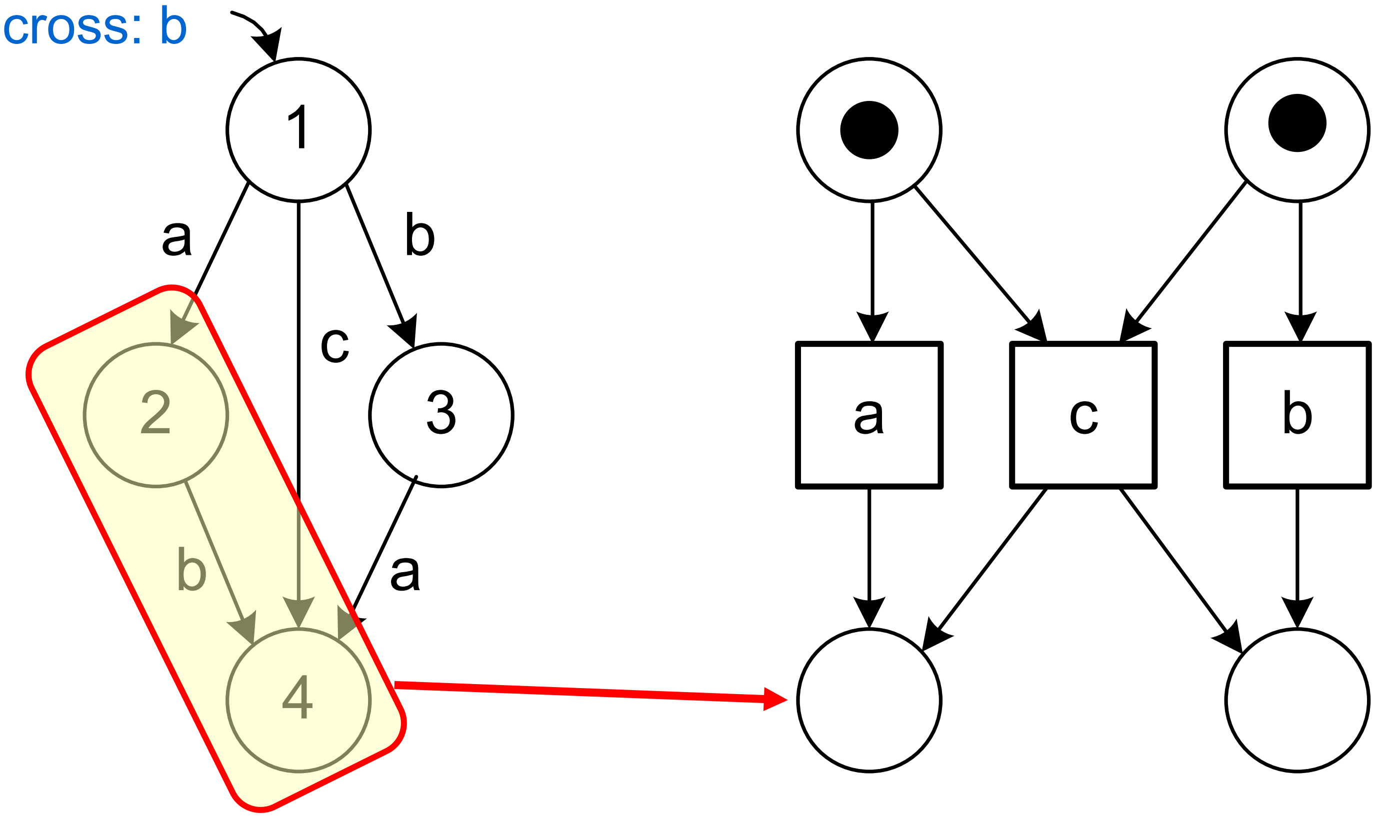
Second region

enter: -
exit: a,c
do not cross: b



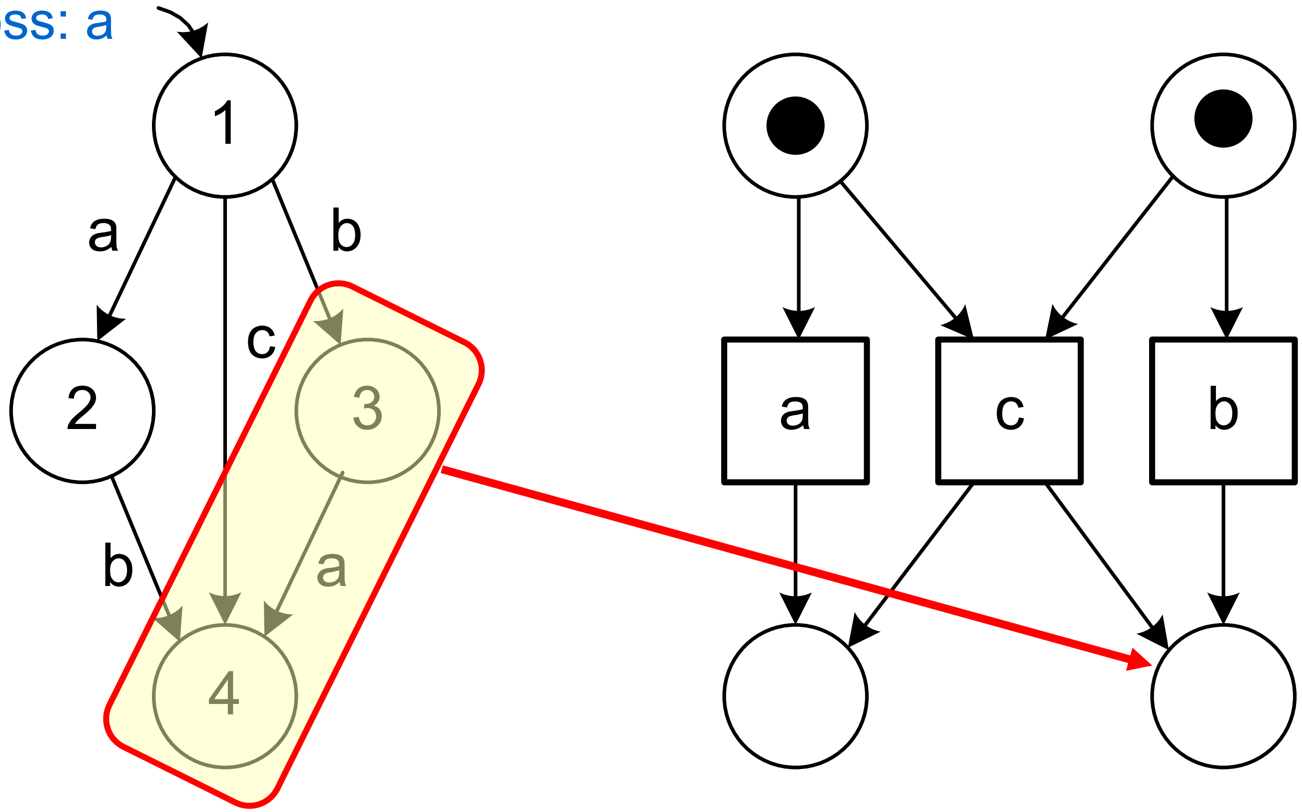
Third region

enter: a,c
exit: -
do not cross: b

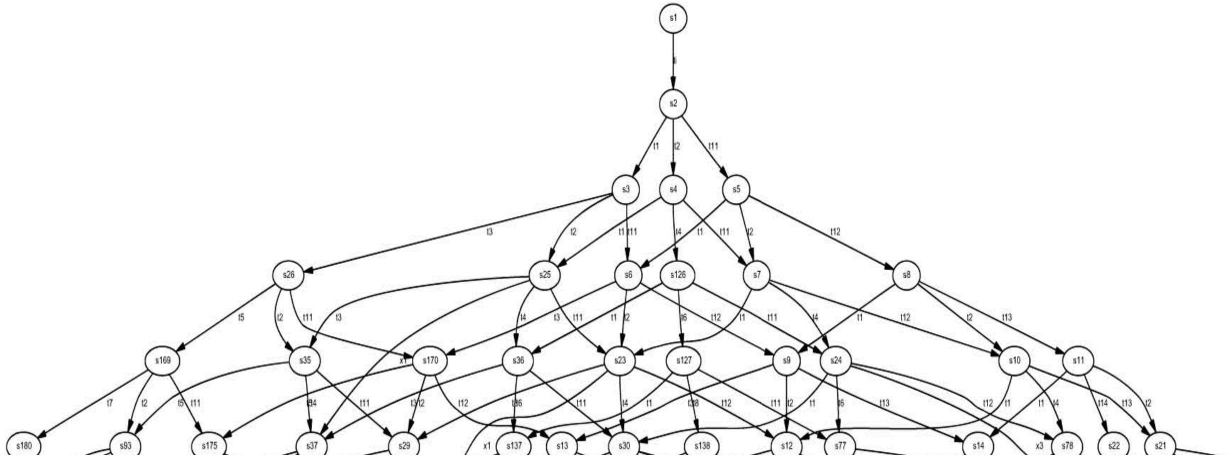


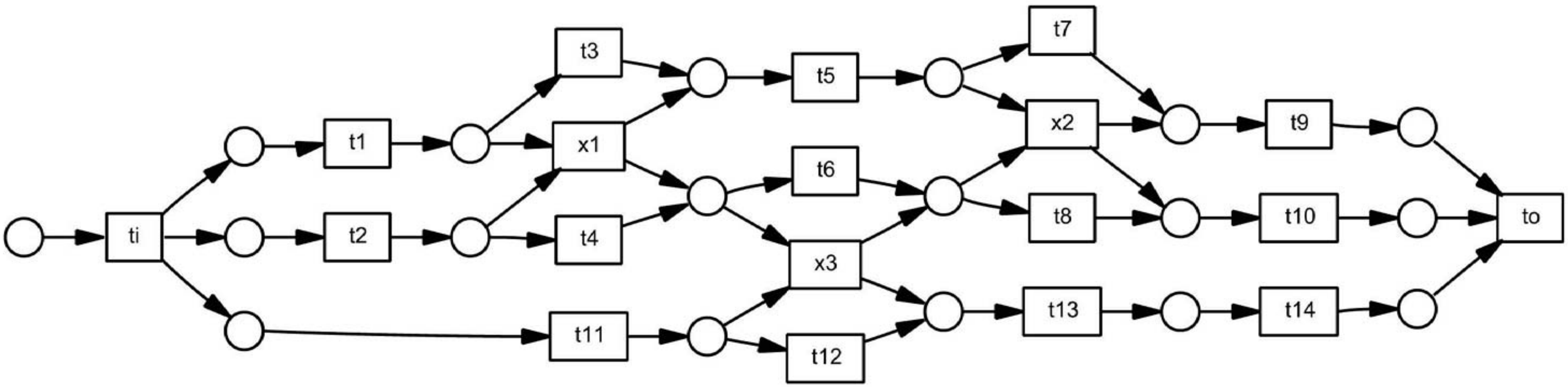
Fourth region

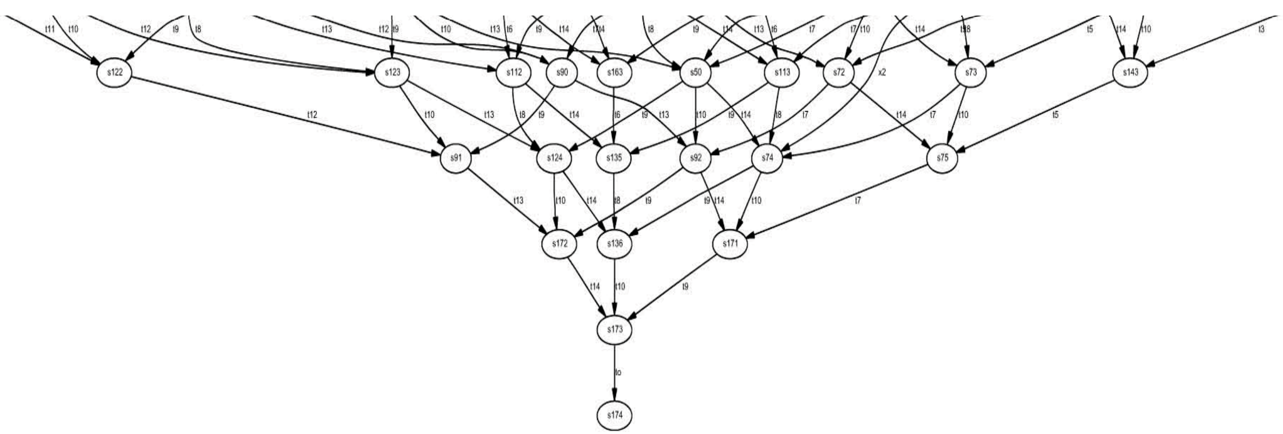
enter: b,c
exit: -
do not cross: a

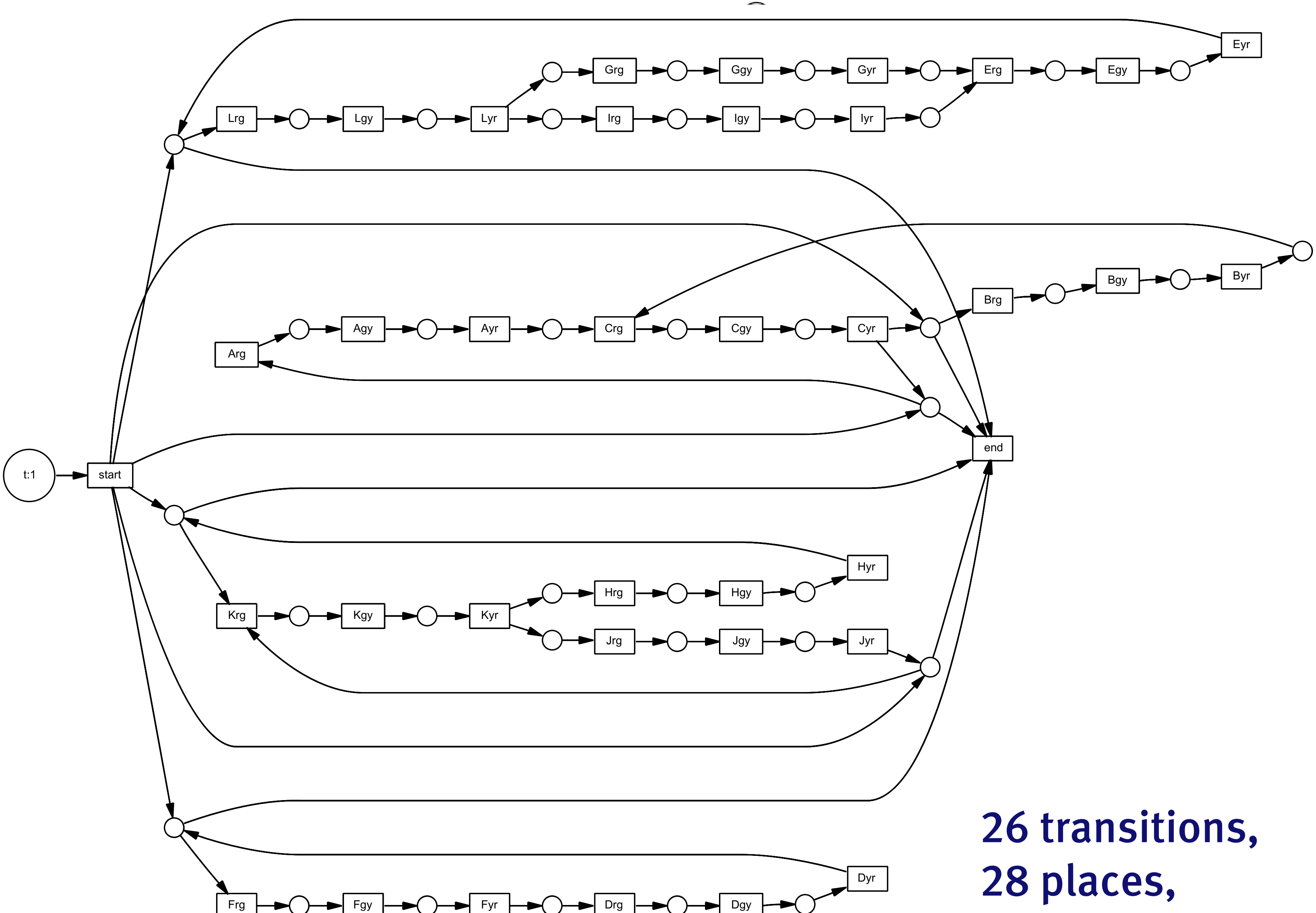


With some extensions (see work Cortadella et al.) any transition system can be converted into a bisimilar Petri net.




ti
t1
t2
t3
t4
x1
t11
t5
t6
x3
t12
t7
x2
t8
t13
t9
t10
t14
to

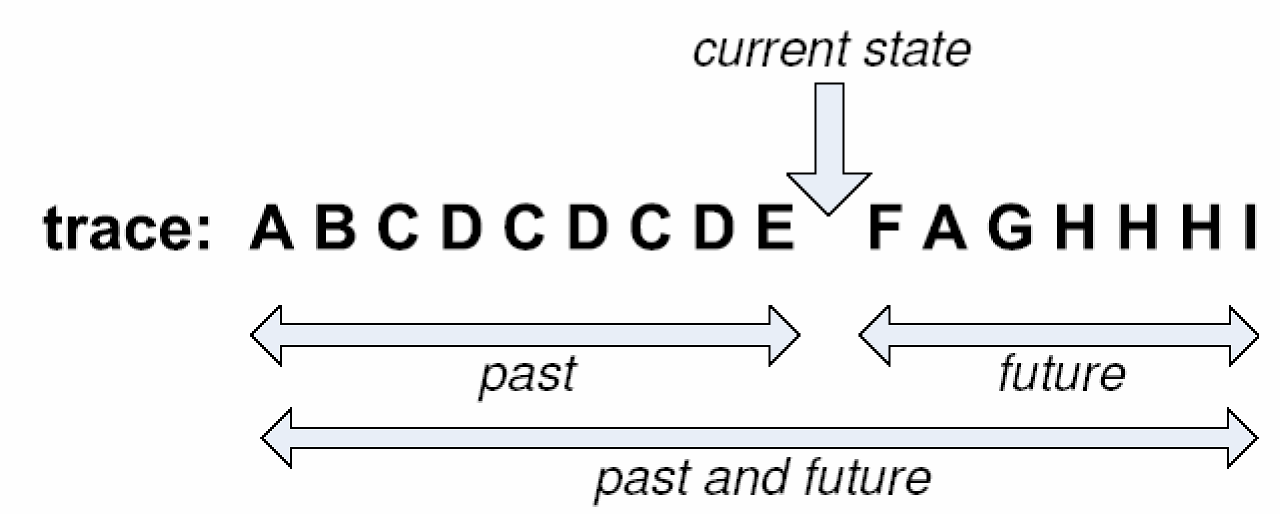




26 transitions,
28 places,
1 token

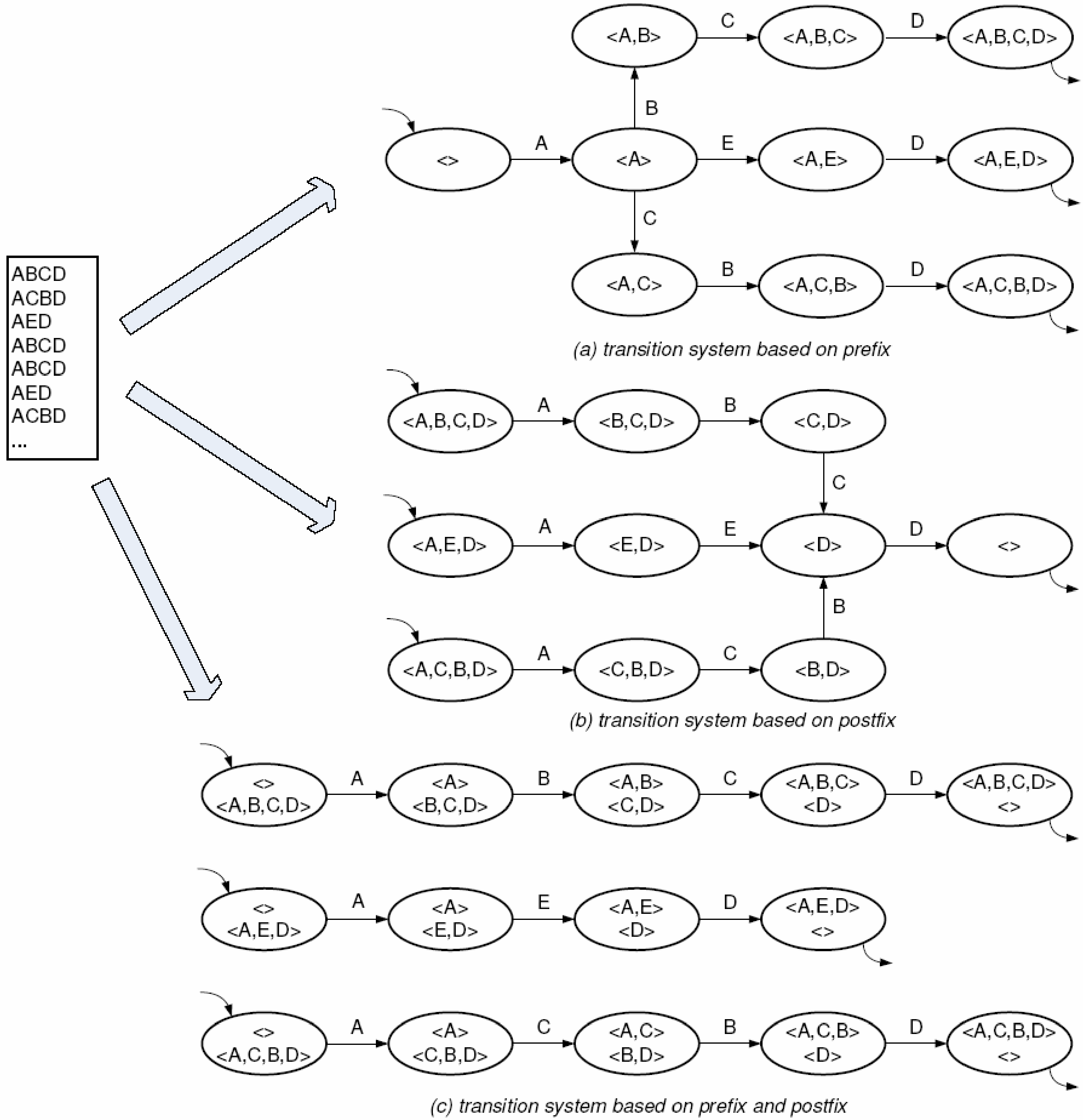
From event logs to transition systems

- **How to determine the current state?**
- **Determine scope:**



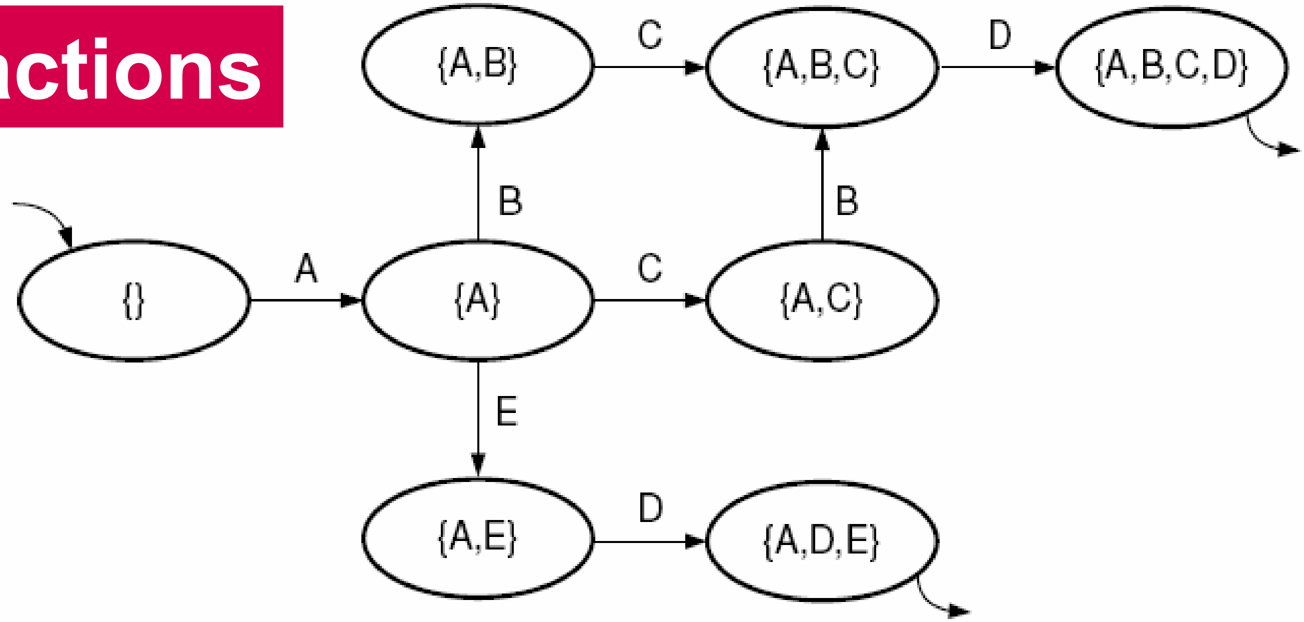
- **Determine abstraction:**
  - *sequence*, i.e., the order of activities is recorded in the state,
  - *multi-set of activities*, i.e., the number of times each activity is executed ignoring their order, or
  - *set of activities*, i.e., the mere presence of activities.

Example

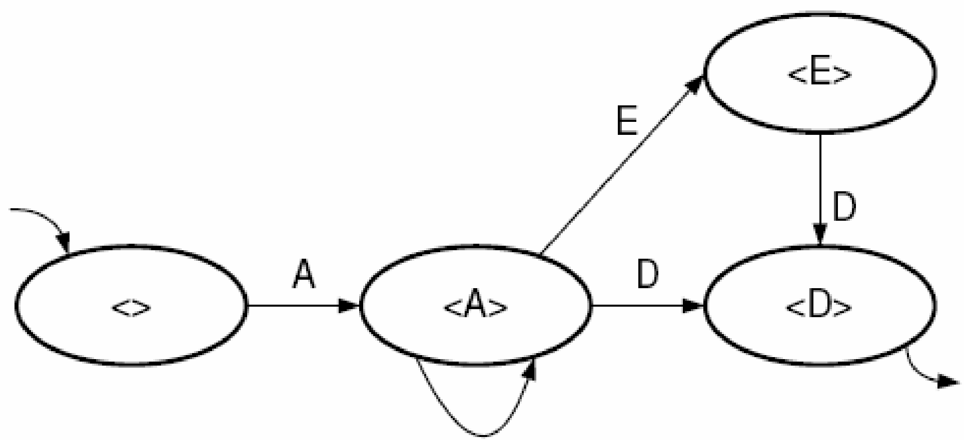

ABCD
ACBD
AED
ABCD
ABCD
AED
ACBD
...
<>
A
<A>
B
<A,B>
C
<A,B,C>
D
<A,B,C,D>
E
<A,E>
D
<A,E,D>
C
<A,C>
B
<A,C,B>
D
<A,C,B,D>
(a) transition system based on prefix
<A,B,C,D>
A
<B,C,D>
B
<C,D>
C
<A,E,D>
A
<E,D>
E
<D>
D
<>
<A,C,B,D>
A
<C,B,D>
C
<B,D>
B
(b) transition system based on postfix
<> <A,B,C,D>
A
<A> <B,C,D>
B
<A,B> <C,D>
C
<A,B,C> <D>
D
<A,B,C,D> <>
<> <A,E,D>
A
<A> <E,D>
E
<A,E> <D>
D
<A,E,D> <>
<> <A,C,B,D>
A
<A> <C,B,D>
C
<A,C> <B,D>
B
<A,C,B> <D>
D
<A,C,B,D> <>
(c) transition system based on prefix and postfix

Other abstractions

ABCD
ACBD
AED
ABCD
ABCD
AED
ACBD
...

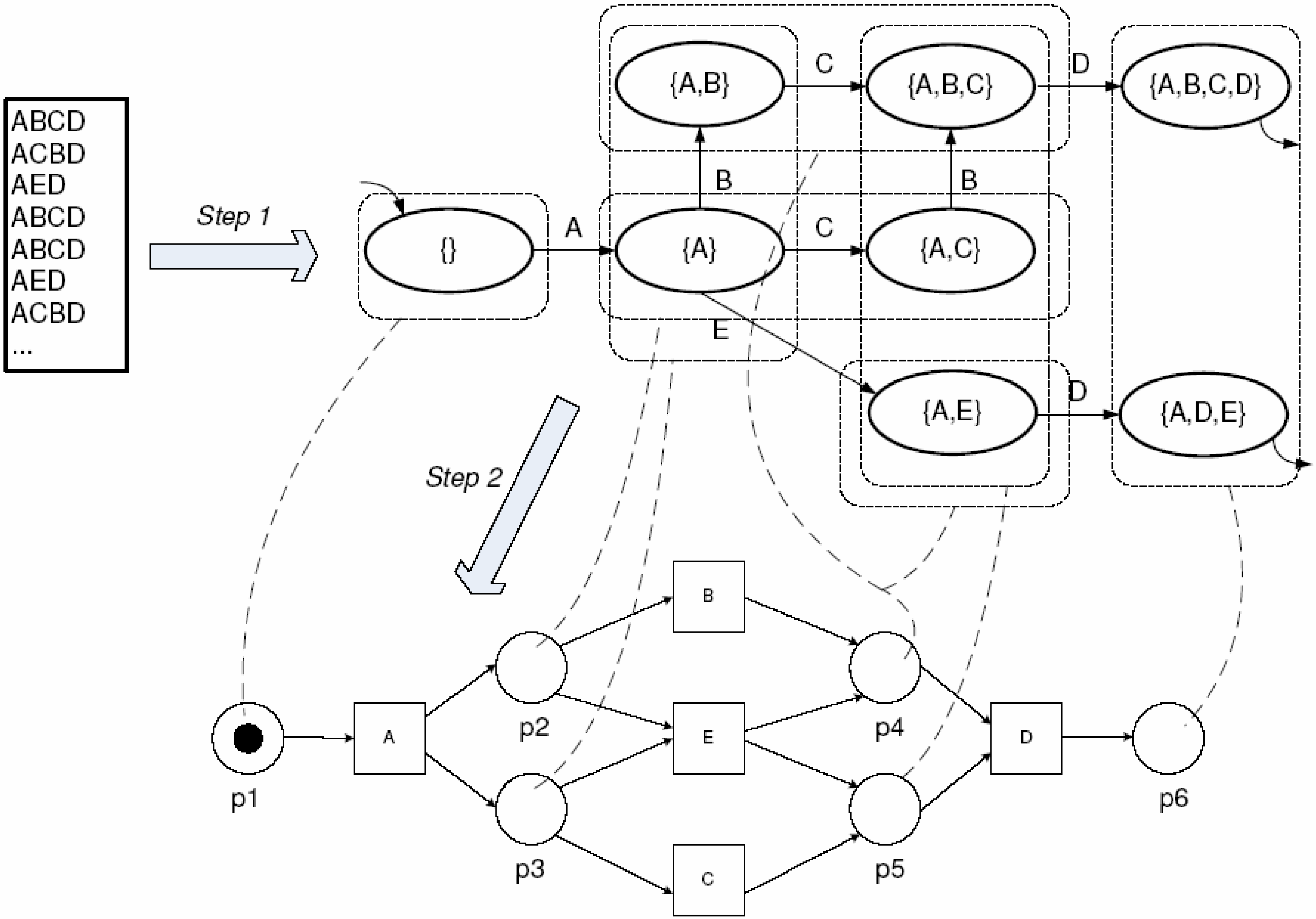


(a) transition system based on sets



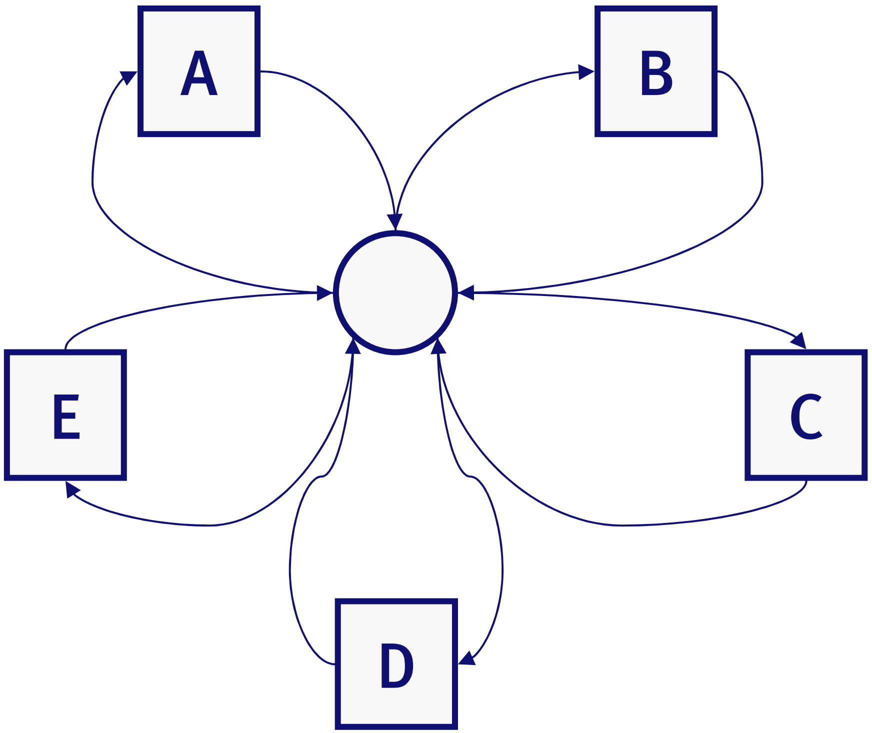
(b) transition system abstracting from B and C

Combination: A two step approach based on controlled abstraction and regions



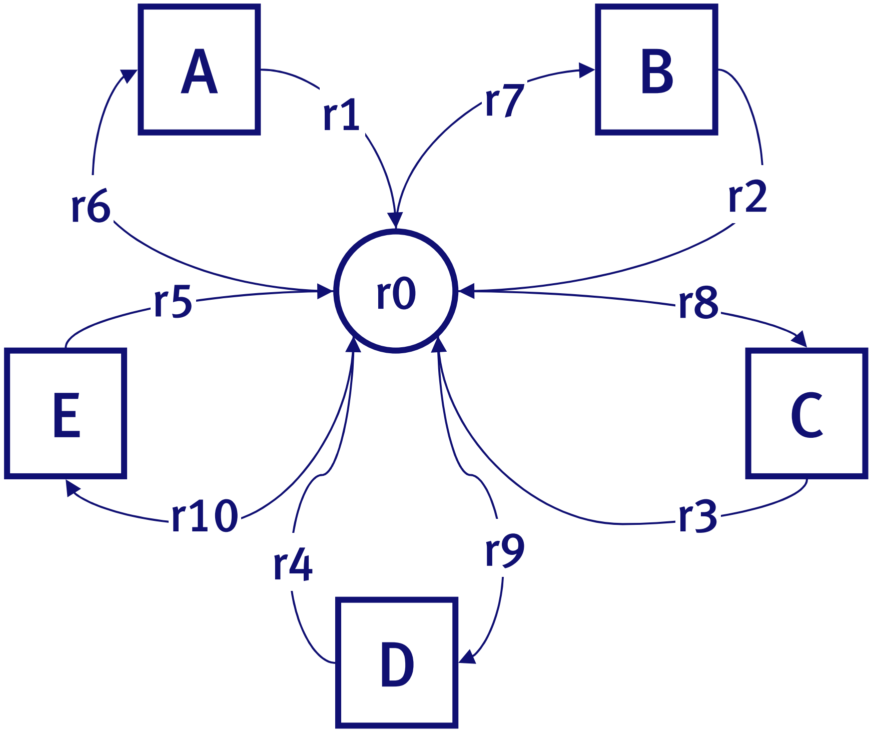
Language-Based Regions
(Van Dongen, Van der Werf, Lorenz, Desel, et al.)

- **Consider the following language:**
  abbe, acde, adce
- **Which prefix-closed is:**
  a, ab, abb, abbe,
  ac, acd, acde,
  ad, adc, adce
- **Idea: What are all the places I can add without making these prefixes impossible?**



Translated into a linear programming problem
A.r >= 0

a	r0−r6	>= 0
ab	r0+r1−r6−r7	>= 0
abb	r0+r1+r2−r6−2 r7	>= 0
abbe	r0+r1+2 r2−r6−2 r7−r10	>= 0
ac	r0+r1−r6−r8	>= 0
acd	r0+r1+r3−r6−r8−r9	>= 0
acde	r0+r1+r3+r4−r6−r8−r9−r10	>= 0
ad	r0+r1−r6−r9	>= 0
adc	r0+r1+r4−r6−r9−r8	>= 0
adce	r0+r1+r4+r3−r6−r9−r8−r10	>= 0

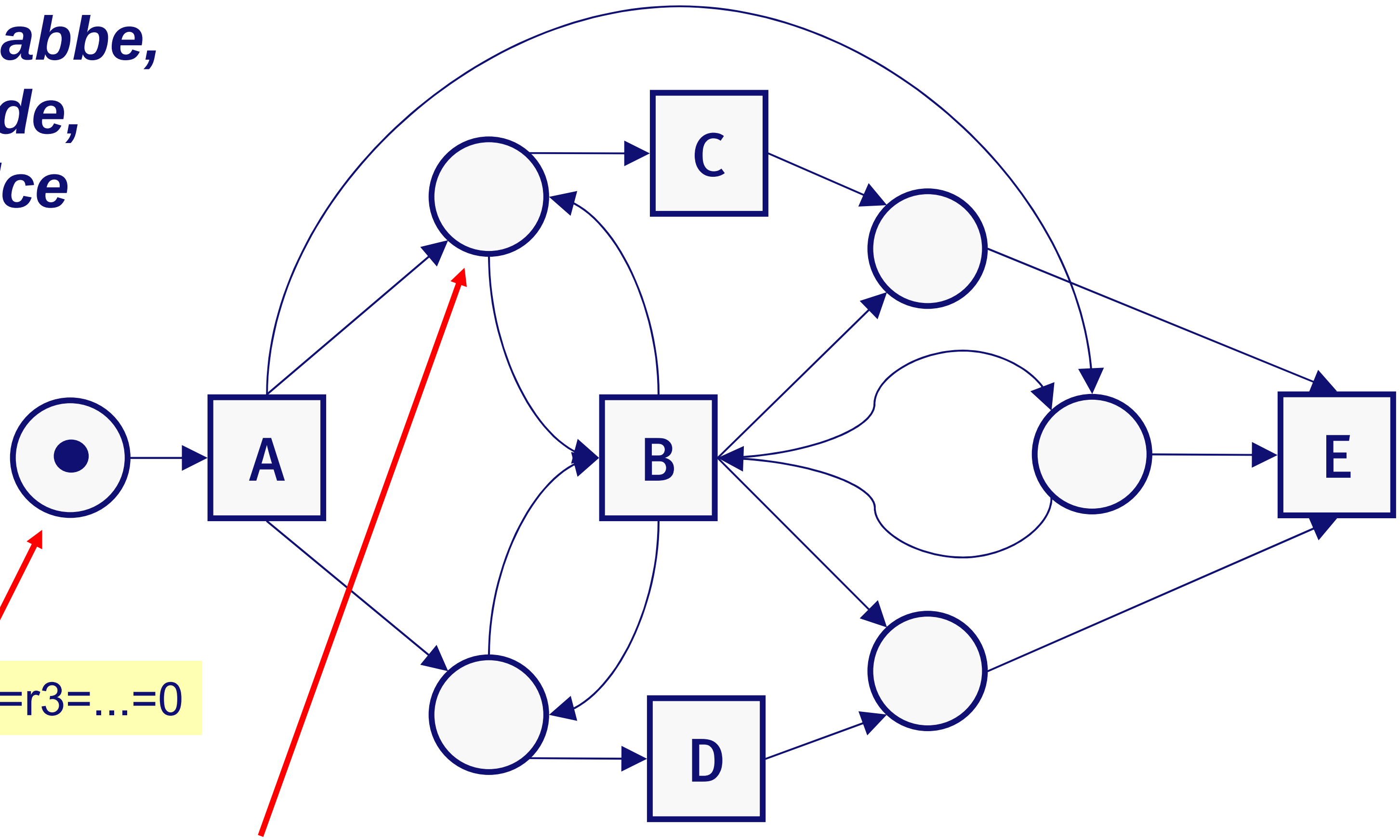


r0=r6=1 and r1=r2=r3=...=0 is an example solution and hence a possible place.

r1=r2=r8=r7=1 and r0=r3=r4=...=0 is an example solution and hence a possible place.

Result of Integer Linear Programming
(multiple formulations possible)

a, ab, abb, abbe,
ac, acd, acde,
ad, adc, adce



r0=r6=1 and r1=r2=r3=...=0

r1=r2=r8=r7=1 and r0=r3=r4=...=0

Customizable and tunable

- There are infinitely many places, but the selection of places to be added can be controlled.
- The ILP formulation can be used to search for subclasses (marked graph, state machine, free-choice, etc.) or to avoid showing "complex" places.
- The ILP formulation can be used to take frequencies into account.

Summary

- Alpha miner
- Multi phase miner
- Genetic process mining
- State-based region mining
- Language based region mining

Many more:

- Fuzzy miner
- Heuristics miner
- Alpha+, Alpha++, Alpha #, etc.
- ...

Balancing Between Overfitting and Underfitting

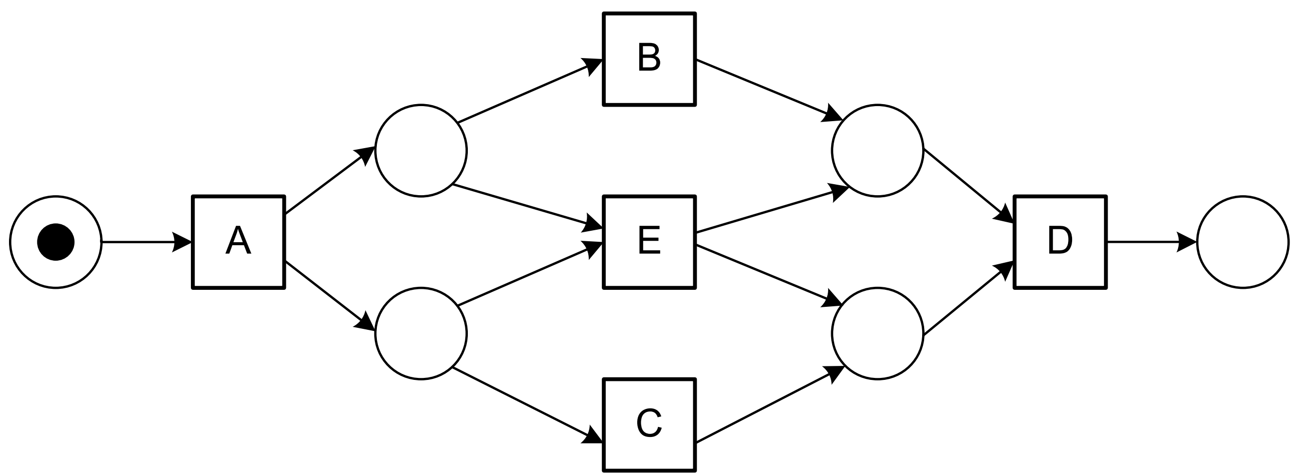
TU/e Technische Universiteit Eindhoven University of Technology

Where innovation starts

Challenge: Balancing Between Underfitting and Overfitting

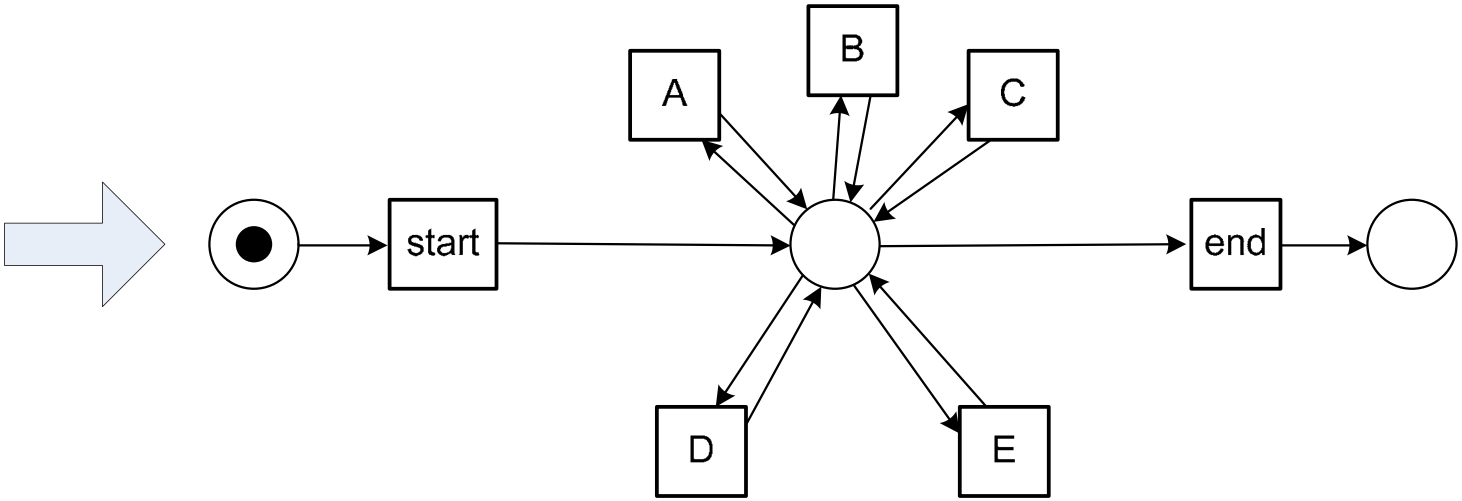
The essence

ABCD
ACBD
AED
ABCD
ABCD
AED
ACBD
...

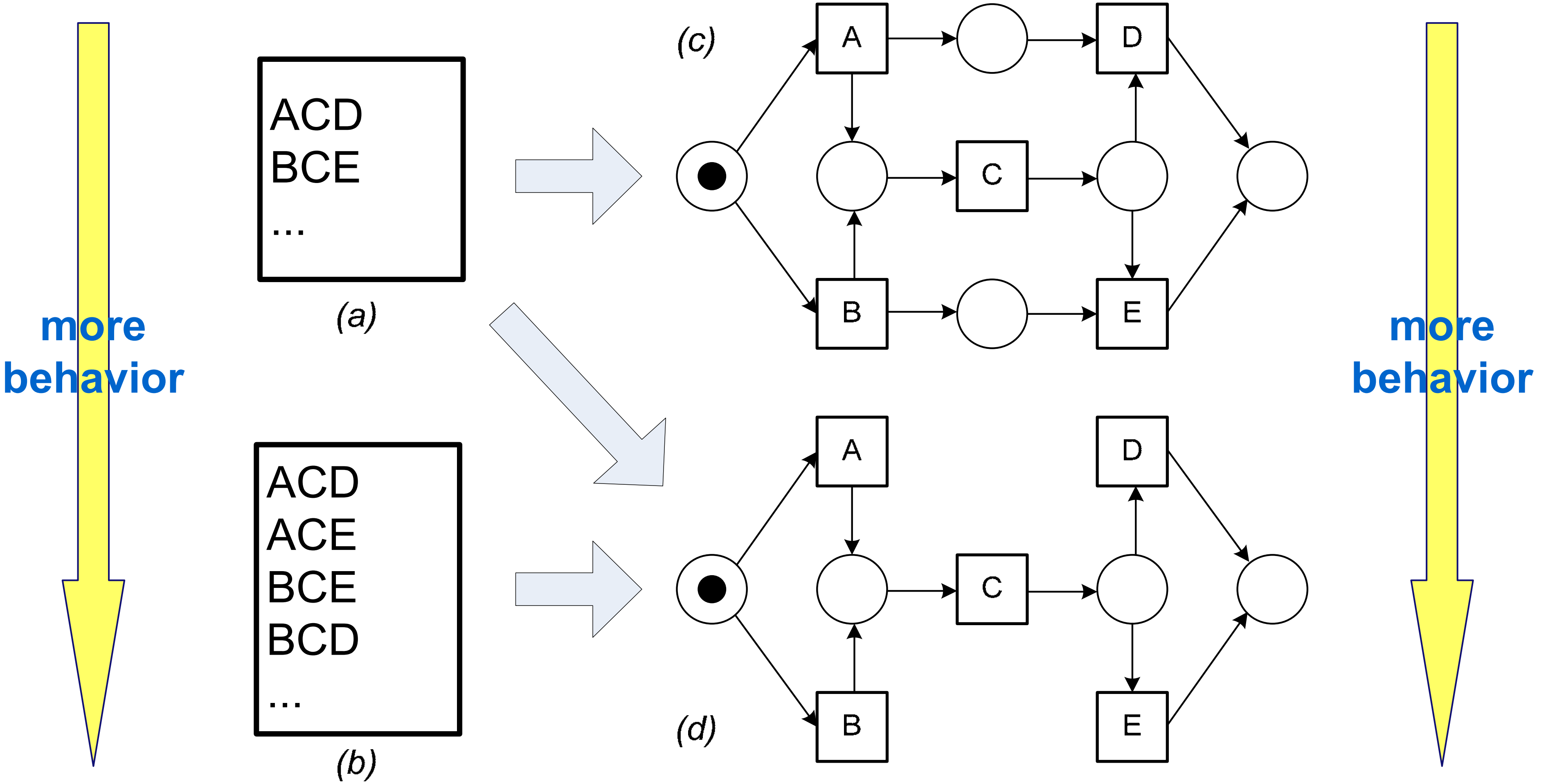


But ...

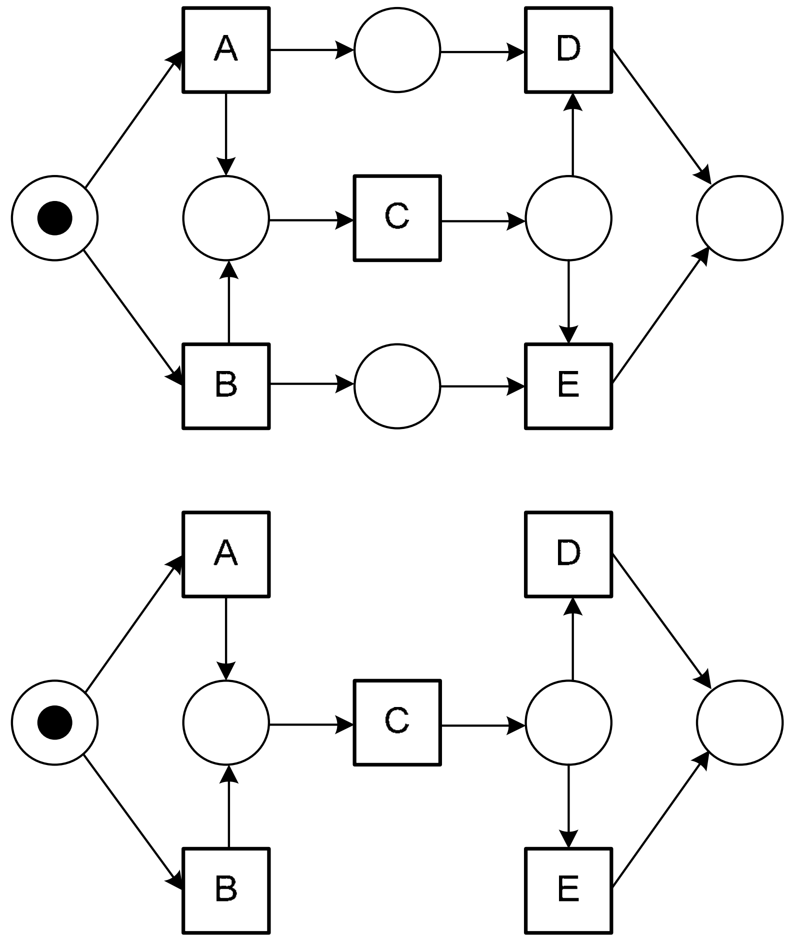
Any log containg activities A, B, C, D, and E.



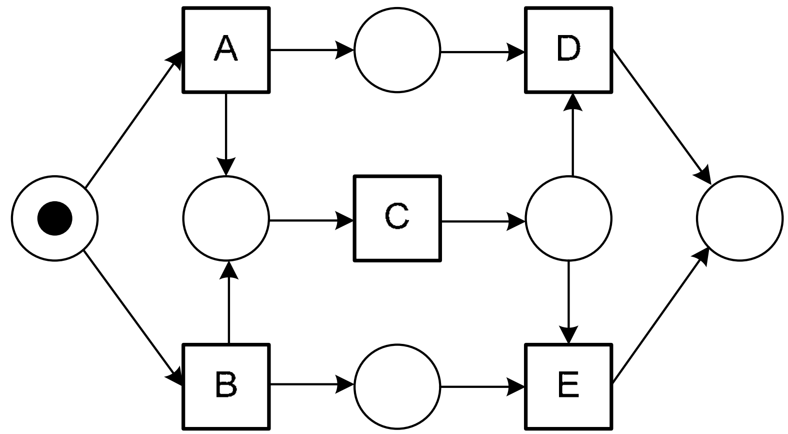
Finding a balance

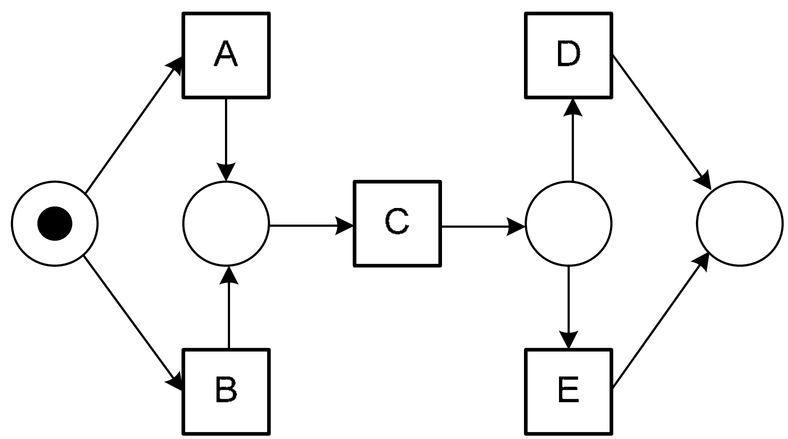


ACD	99
ACE	0
BCE	85
BCD	0

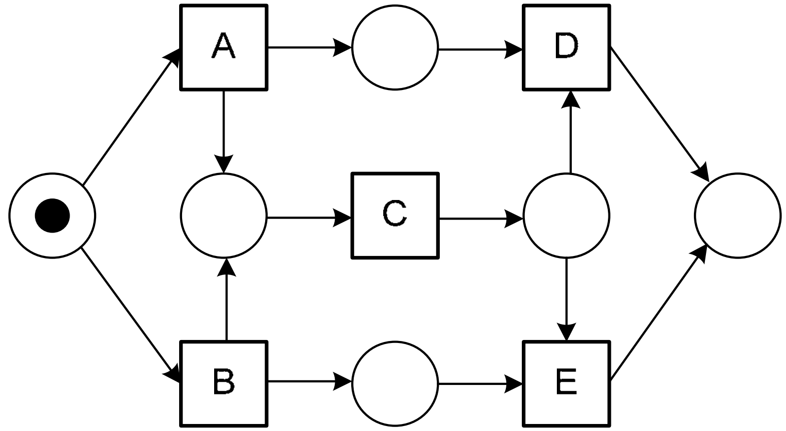


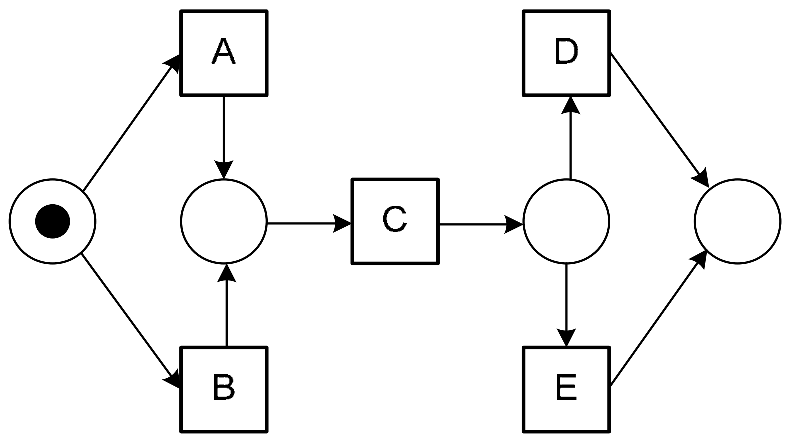
ACD	**99**
ACE	**88**
BCE	**85**
BCD	**78**





ACD	99
ACE	2
BCE	85
BCD	3

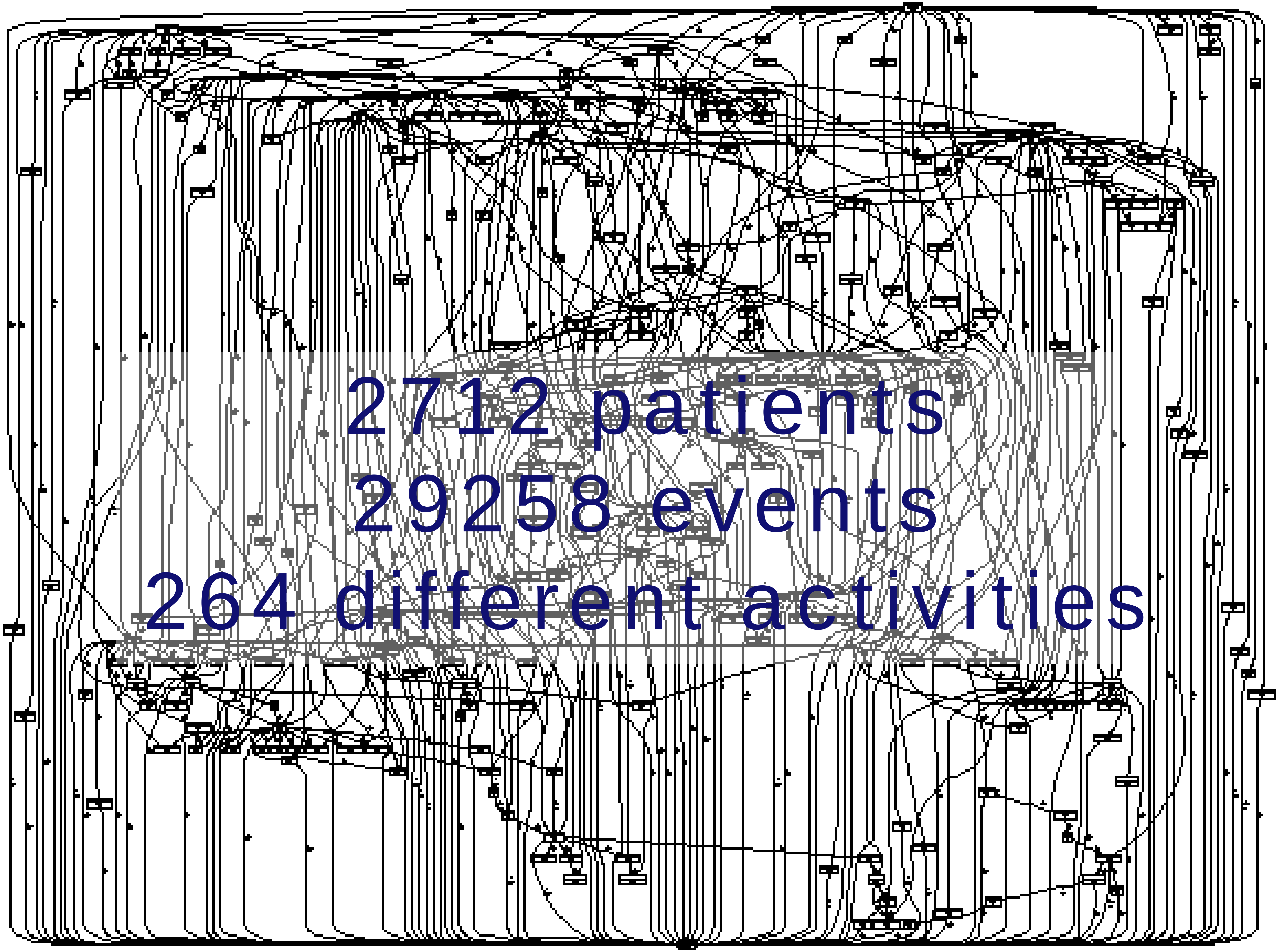


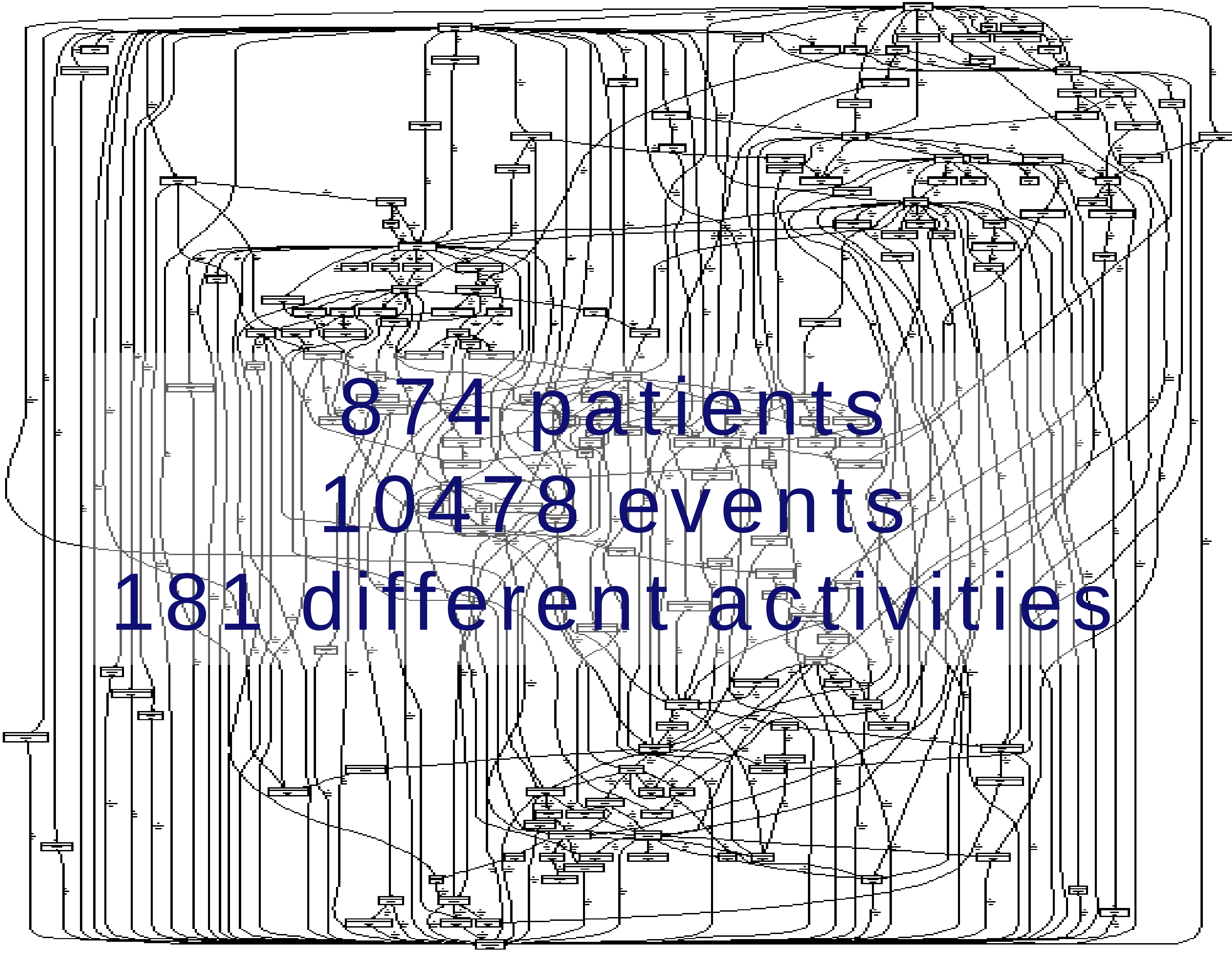


Important observations

- Frequencies matter!
- Adding a place equals restricting behavior!
- "The model" does not exist!

Relevance

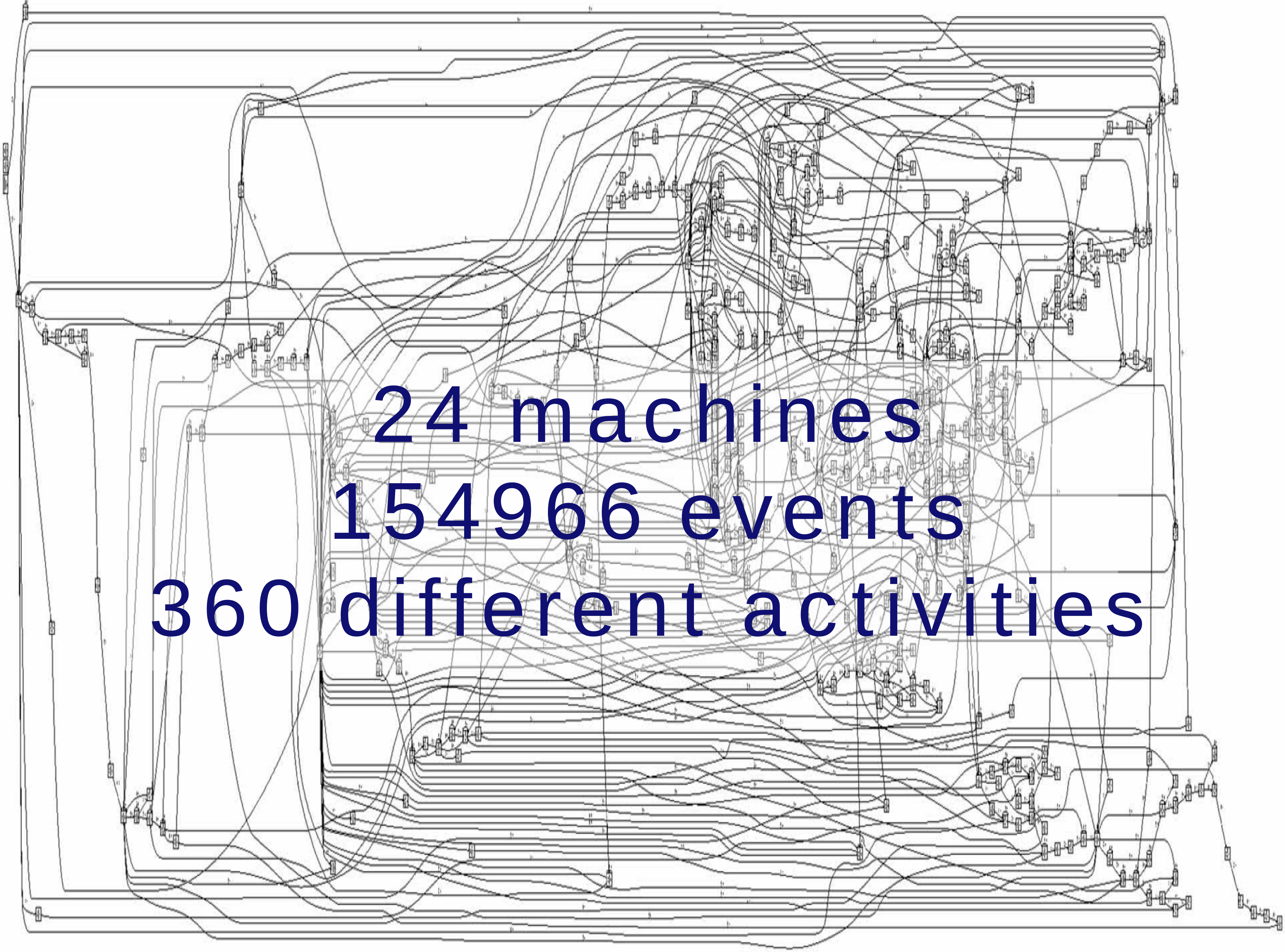

2712 patients
29258 events
264 different activities



874 patients

10478 events

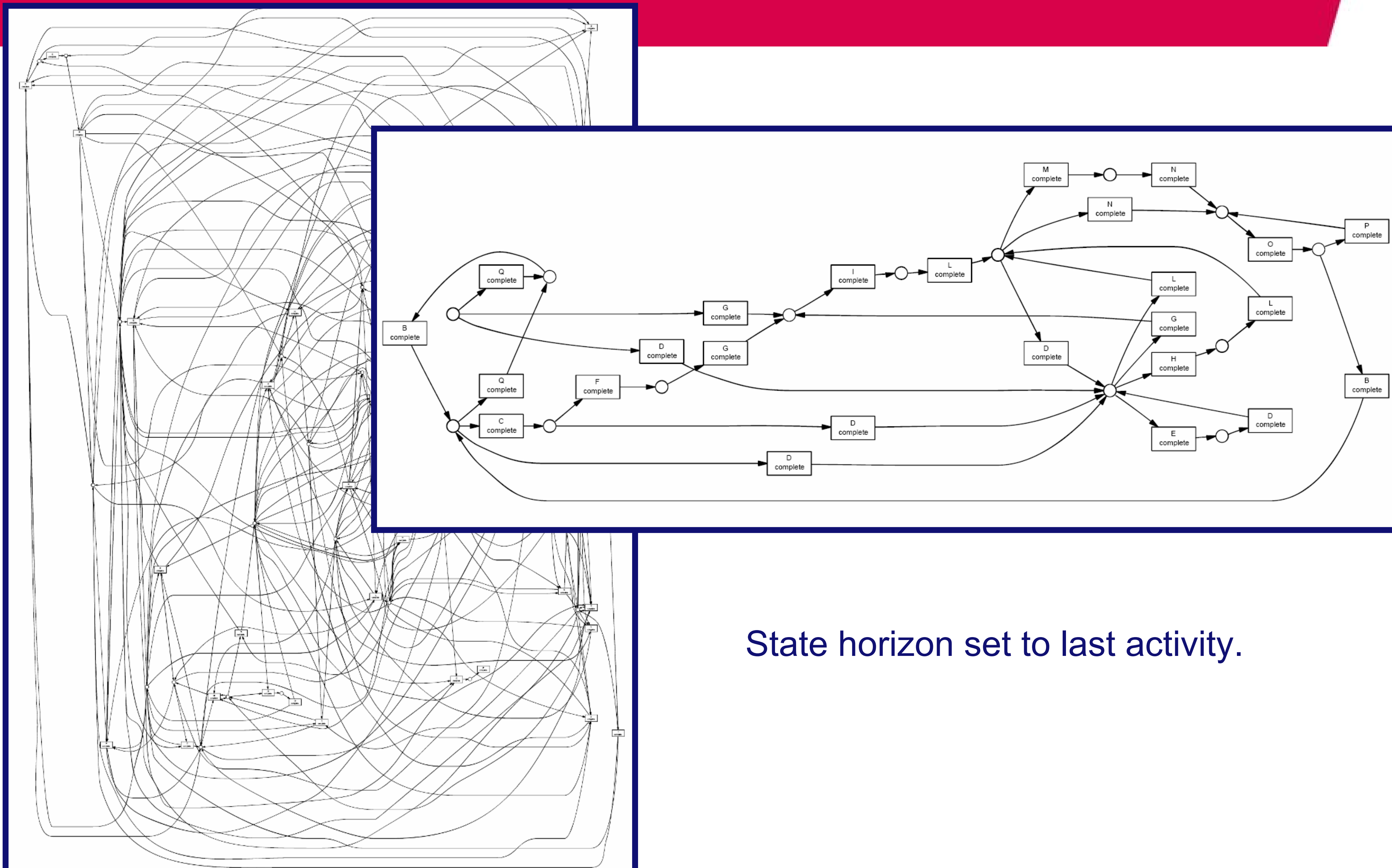
181 different activities


24 machines
154966 events
360 different activities

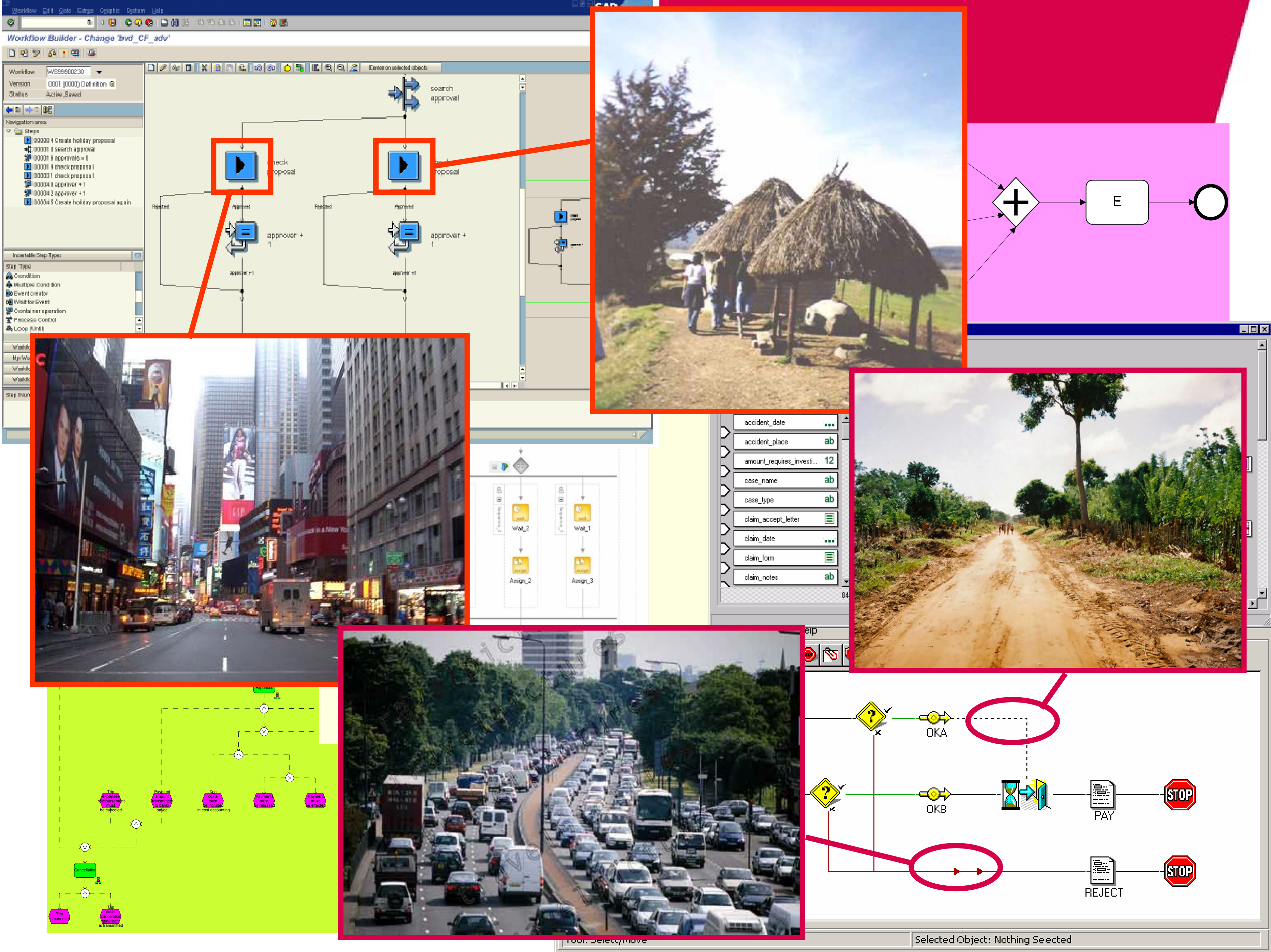
Problems

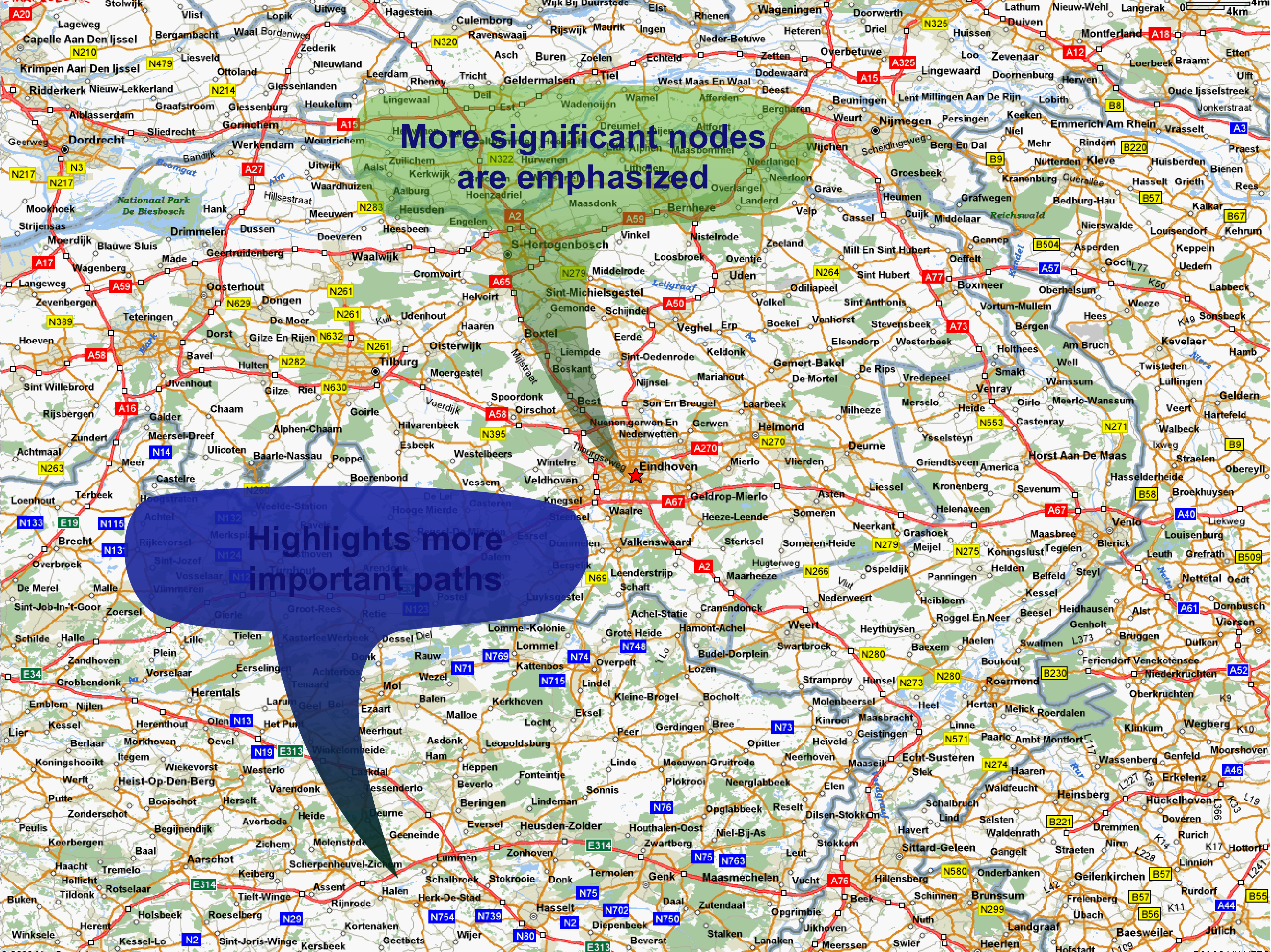
- Representational bias (i.e., generalization is driven by representation rather than log or preferences).
- Inability of dealing with or detecting noise.
- Wrong abstraction level.
- Limitation of current process modeling (visualization) techniques.

Example: Heusden



State horizon set to last activity.

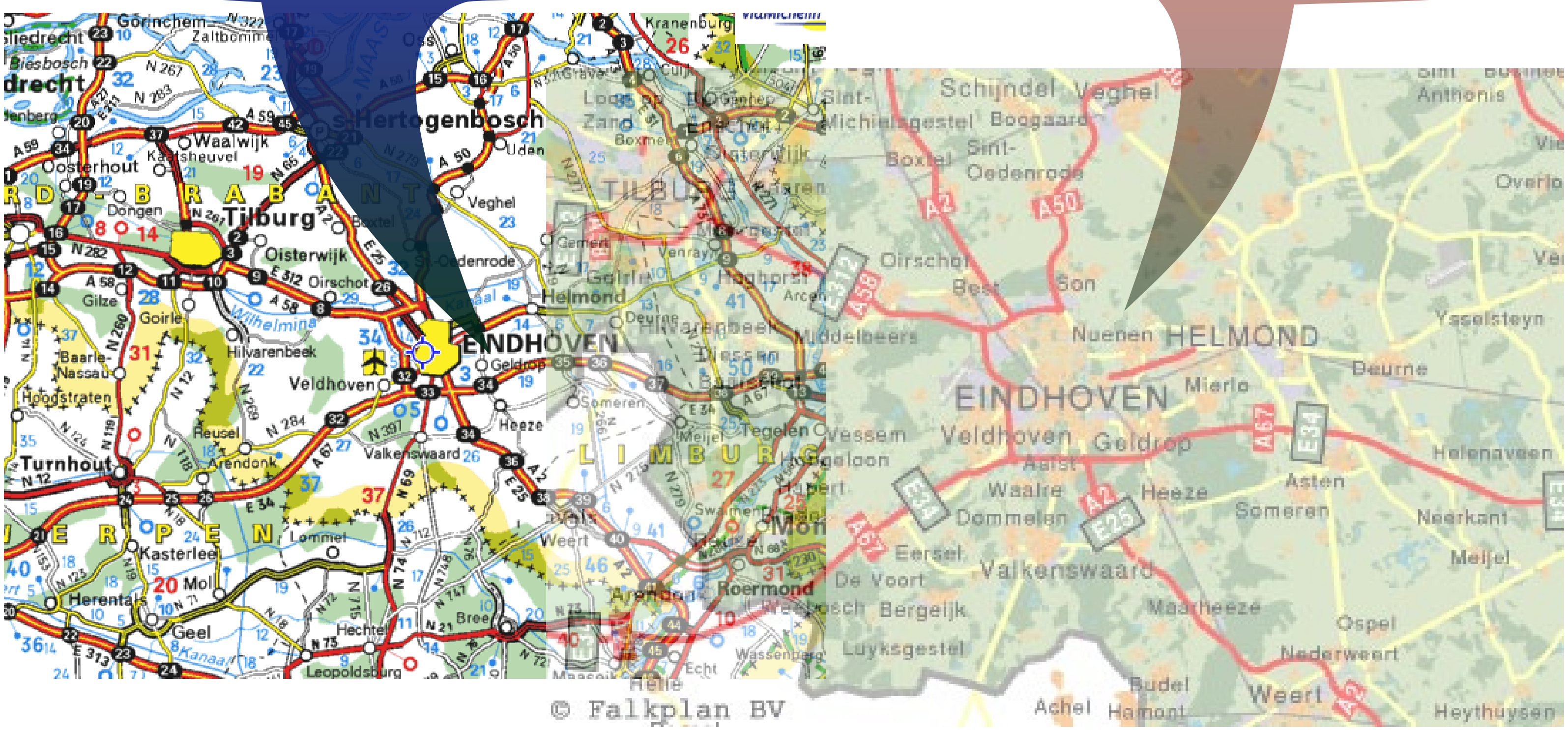

Workflow Builder - Change 'bvd_CF_adv'
search approval
check proposal
approver + 1
E
accident_date
accident_place
amount_requires_investi...
case_name
case_type
claim_accept_letter
claim_date
claim_form
claim_notes
Wait_2
Wait_1
Assign_2
Assign_3
OKA
OKB
PAY
REJECT
STOP
STOP
Selected Object: Nothing Selected


More significant nodes are emphasized
Highlights more important paths

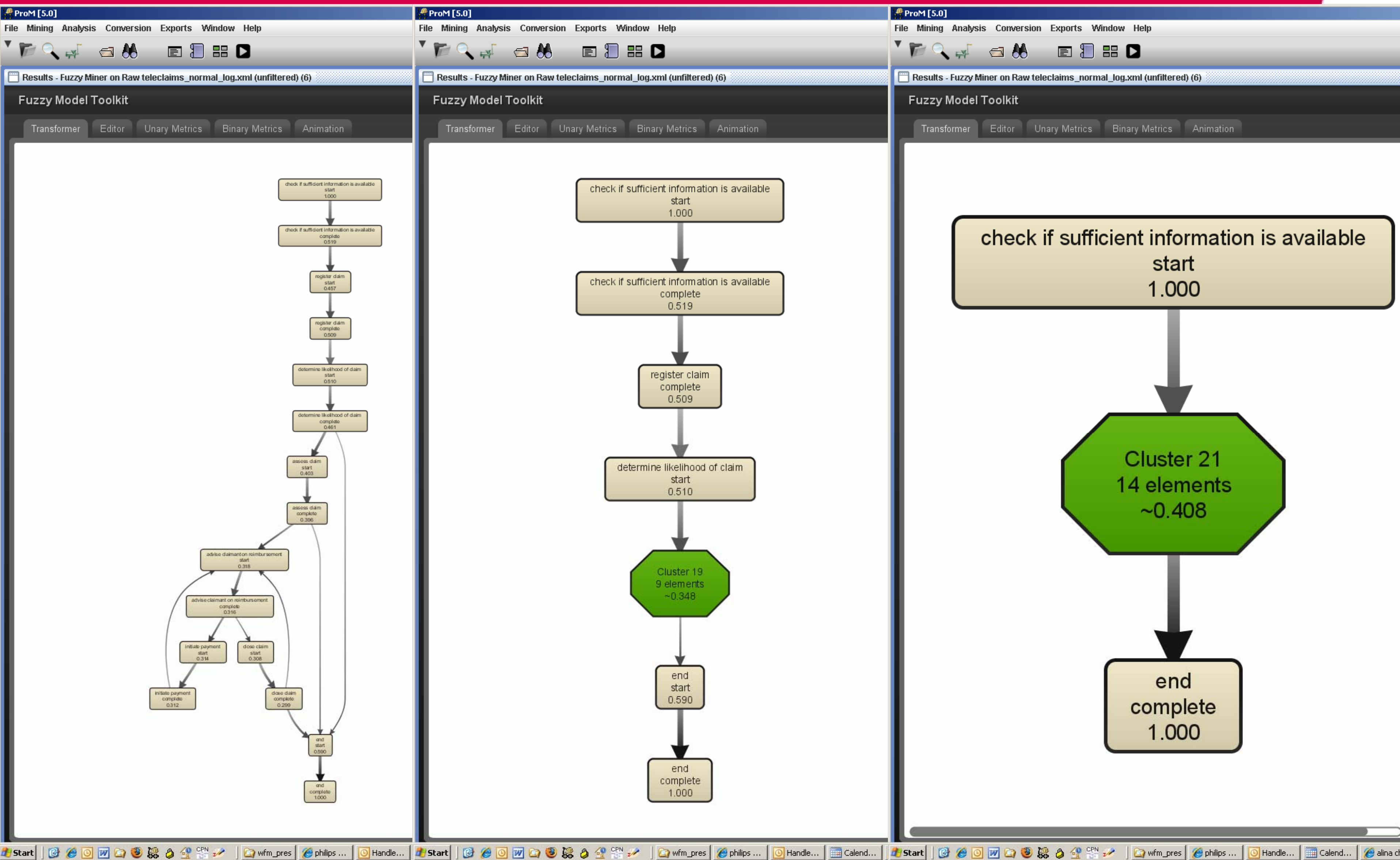
More to learn from maps...

Aggregation
Clustering of coherent, less significant structures

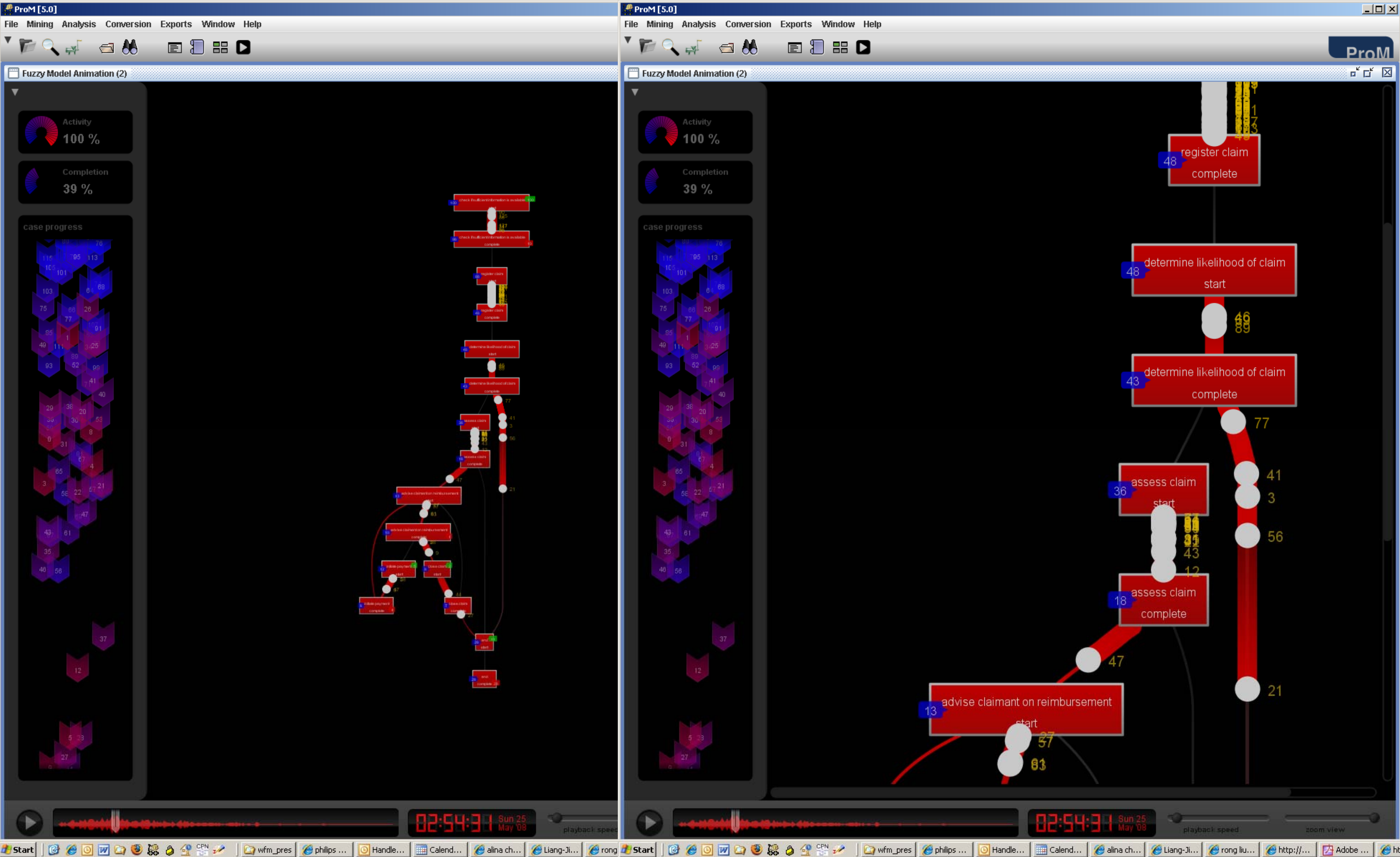
Abstraction
Removing isolated, less significant structures



Fuzzy miner


ProM [5.0]
File Mining Analysis Conversion Exports Window Help
Results - Fuzzy Miner on Raw teleclaims_normal_log.xml (unfiltered) (6)
Fuzzy Model Toolkit
Transformer Editor Unary Metrics Binary Metrics Animation
check if sufficient information is available start 1.000
check if sufficient information is available complete 0.519
register claim complete 0.509
determine likelihood of claim start 0.510
Cluster 19 9 elements ~0.348
end start 0.590
end complete 1.000
check if sufficient information is available start 1.000
Cluster 21 14 elements ~0.408
end complete 1.000

Showing reality



ProM Tool

ProM

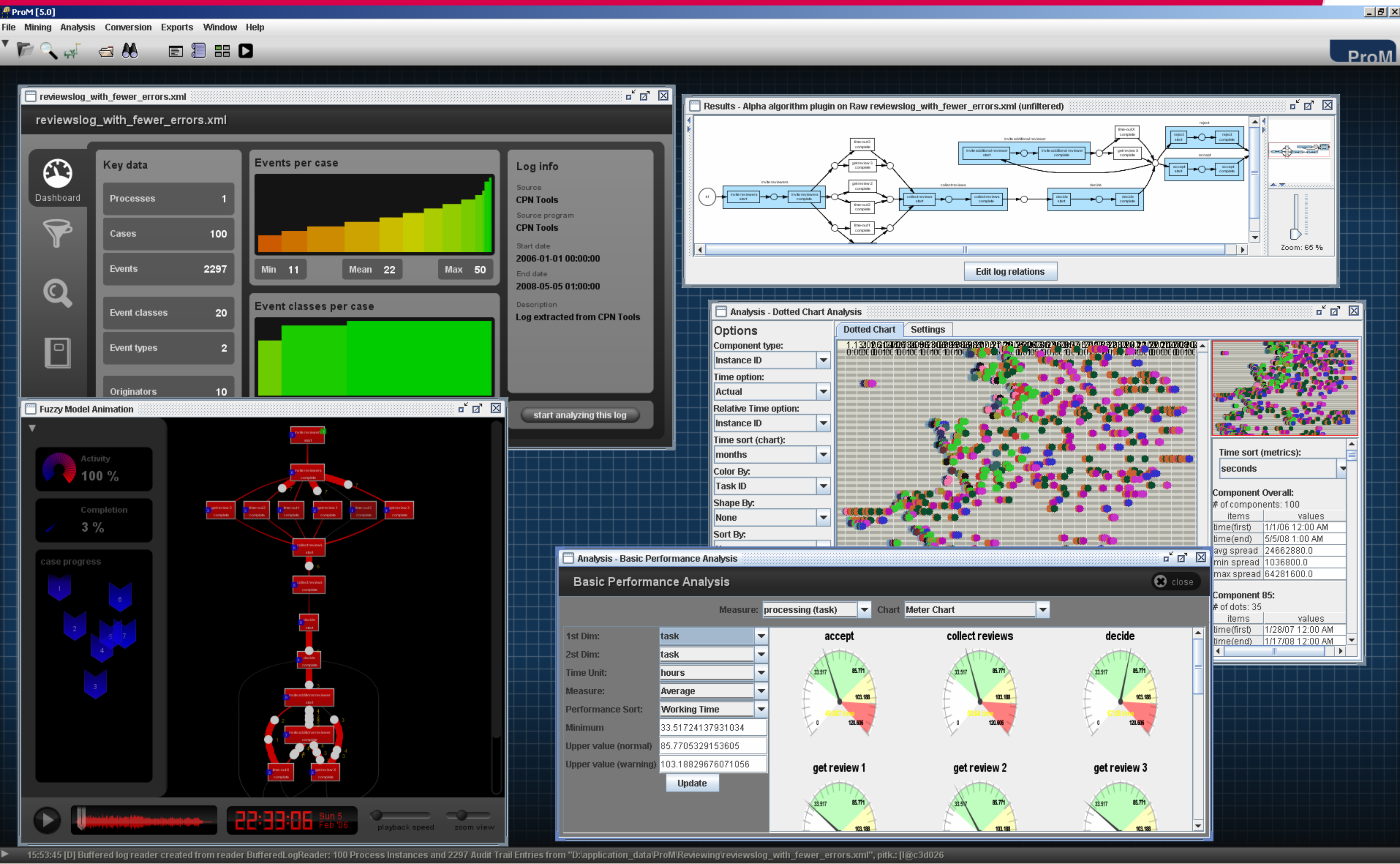
Technische Universiteit
Eindhoven
University of Technology

Where innovation starts

ProM

- Open source initiative started in 2003 after several early prototypes.
- Common Public License (CPL).
- Current version: 5.0.
- ProMimport: to extract MXML from all kinds of applications
- Plug-in architecture.
- About 250 plug-ins available:
  - mining plug-ins: 38 (all mining algorithms presented and many more)
  - analysis plug-ins: 71 (e.g., verification, SNA, LTL, conformance checking, etc.)
  - import: 21 (for loading EPCs, Petri nets, YAWL, BPMN, etc.)
  - export: 44 (for storing EPCs, Petri nets, YAWL, BPMN, BPEL, etc.)
  - conversion: 45 (e.g., translating EPCs or BPMN into Petri nets)
  - filter: 24 (e.g., removing infrequent activities)

Screenshot of ProM 5.0



Conclusion

Conclusion

- The existence of event data enables a wide variety of process mining techniques ranging from process discovery to conformance checking.
- ProM supports this through +/- 250 plug-ins.
- A reality check for people that are involved in process modeling.
- Interesting challenges for both researchers and practitioners.
- Please join us! (www.processmining.org)

References

Introduction to Process Mining and ProM

- W.M.P. van der Aalst, H.A. Reijers, A.J.M.M. Weijters, B.F. van Dongen, A.K. Alves de Medeiros, M. Song, and H.M.W. Verbeek. Business Process Mining: An Industrial Application. *Information Systems 32(1), 713-732.*
- W.M.P. van der Aalst, et al. ProM 4.0: Comprehensive Support for Real Process Analysis. In J. Kleijn, A. Yakovlev, Petri Nets 2007, Lecture Notes in Computer Science, Vol. 4546, pp. 484-494. Berlin: Springer, 2007.
- W.M.P. van der Aalst and A.J.M.M. Weijters. Process Mining. *In M. Dumas, W.M.P. van der Aalst, and A.H.M. ter Hofstede, editors, Process-Aware Information Systems: Bridging People and Software through Process Technology, pages 235-255. Wiley & Sons, 2005*
- W.M.P. van der Aalst , B.F. van Dongen, J. Herbst, L. Maruster, G. Schimm, and A.J.M.M. Weijters. Workflow Mining: A Survey of Issues and Approaches. *Data and Knowledge Engineering, 47(2):237-267, 2003.*

Note that these references are far from complete and not intended to provide a comprehensive overview. See www.processmining.org for a good overview of (at least) all ProM-related publications.

References

Algorithmic Approaches

- W.M.P. van der Aalst , A.J.M.M. Weijters, and L. Maruster. Workflow Mining: Discovering Process Models from Event Logs. IEEE Transactions on Knowledge and Data Engineering (TKDE), volume 16(9), pages 1128-1142, 2004.
- L. Wen, W.M.P. van der Aalst, J. Wang, and J. Sun. Mining process models with non-free-choice constructs. Data Mining and Knowledge Discovery, 15(2):145-180, 2007.
- W.M.P. van der Aalst and B.F. van Dongen. Discovering Workflow Performance Models from Timed Logs. In Y. Han, S. Tai, and D. Wikarski, editors, International Conference on Engineering and Deployment of Cooperative Information Systems (EDCIS 2002), volume 2480 of Lecture Notes in Computer Science, pages 45-63. Springer-Verlag, Berlin, 2002.
- B.F. van Dongen and W.M.P. van der Aalst. Multi-phase Process mining: Building Instance Graphs. Conceptual Modeling - ER 2004, LNCS 3288, pages 362-376, 2004.
- A.J.M.M. Weijters and W.M.P. van der Aalst. Rediscovering Workflow Models from Event-Based Data using Little Thumb. Integral Computer-Aided Engineering, 10(2):151-162, 2003.
- C.W. Günther and W.M.P. van der Aalst. Fuzzy Mining: Adaptive Process Simplification Based on Multi-perspective Metrics. In International Conference on Business Process Management (BPM 2007), volume 4714 of Lecture Notes in Computer Science, pages 328-343. Springer-Verlag, Berlin, 2007.

References

Genetic Mining

- A.K. Alves de Medeiros, A.J.M.M. Weijters and W.M.P. van der Aalst. Genetic Process Mining: An Experimental Evaluation. Data Mining and Knowledge Discovery, volume 14, issue 2, pages 245-304, 2007.
- W.M.P. van der Aalst, A.K. Alves de Medeiros and A.J.M.M. Weijters. Genetic Process Mining. 26th International Conference on Applications and Theory of Petri Nets (ICATPN 2005), G. Ciardo and P. Darondeau, LNCS 3536, pages 48-69, 2005.

References

Mining Based on Regions

- J.M.E.M. van der Werf, B.F. van Dongen, C.A.J. Hurkens, and A. Serebrenik. Process Discovery using Integer Linear Programming. In K. van Hee and R. Valk, editors, *Proceedings of the 29th International Conference on Applications and Theory of Petri Nets (Petri Nets 2008)*, volume 5062 of *Lecture Notes in Computer Science*, pages 368-387. Springer-Verlag, Berlin, 2008.
- W.M.P. van der Aalst, V. Rubin, B.F. van Dongen, E. Kindler, and C.W. Günther. Process Mining: A Two-Step Approach using Transition Systems and Regions. BPM Center Report BPM-06-30, BPMcenter.org, 2006.
- R. Bergenthum, J. Desel, R. Lorenz, and S. Mauser. Process Mining Based on Regions of Languages. In G. Alonso, P. Dadam, and M. Rosemann, editors, *International Conference on Business Process Management (BPM 2007)*, volume 4714 of *Lecture Notes in Computer Science*, pages 375-383. Springer-Verlag, Berlin, 2007.

References

Mining Other Perspectives

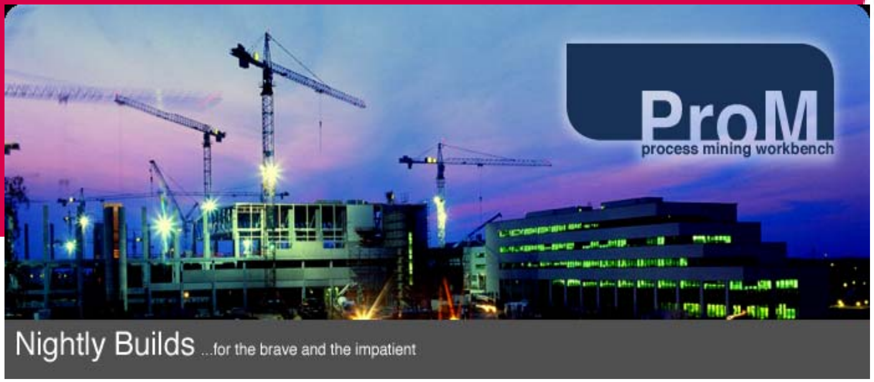
- A. Rozinat and W.M.P. van der Aalst. Decision Mining in ProM. In S. Dustdar, J.L. Faideiro, and A. Sheth, editors, *International Conference on Business Proces Management (BPM 2006)*, volume 4102 of *Lecture Notes in Computer Science*, pages 420-425. Springer-Verlag, Berlin, 2006.
- Wil M. P. van der Aalst, Hajo A. Reijers. Minseok Song: Discovering Social Networks from Event Logs. Computer Supported Cooperative Work 14(6): 549-593, 2005.
- Anne Rozinat, R. S. Mans, Minseok Song, Wil M. P. van der Aalst: Discovering colored Petri nets from event logs. STTT 10(1): 57-74, 2008.

References

Conformance Checking and Extension

- A. Rozinat and W.M.P. van der Aalst. Conformance Checking of Processes Based on Monitoring Real Behavior. Information Systems, Volume 33, Issue 1, Pages 64-95, 2008.
- W.M.P. van der Aalst, M. Dumas, C. Ouyang, A. Rozinat, and H.M.W. Verbeek. Conformance Checking of Service Behavior. ACM Transactions on Internet Technology (TOIT), Volume 8, Issue 3, 2008.
- A. Rozinat, M.T. Wynn, W.M.P. van der Aalst, A.H.M. ter Hofstede and C.J. Fidge. Workflow Simulation for Operational Decision Support Using Design, Historic and State Information. In M. Dumas, M. Reichert, and M.-C. Shan (Eds.): BPM 2008, LNCS 5240, pp. 196–211, Springer-Verlag Berlin.
- Anne Rozinat, Ana Karla Alves de Medeiros, Christian W. Günther, A. J. M. M. Weijters, Wil M. P. van der Aalst: The Need for a Process Mining Evaluation Framework in Research and Practice. Business Process Management Workshops 2007: 84-89.

Relevant WWW sites



- http://www.processmining.org
- **http:// promimport.sourceforge.net**
- **http://prom.sourceforge.net**
- **http://www.workflowpatterns.com**
- **http://www.workflowcourse.com**
- **http://www.vdaalst.com**