Process Mining: Beyond Business Intelligence

ProM

prof.dr.ir. Wil van der Aalst

www.processmining.org

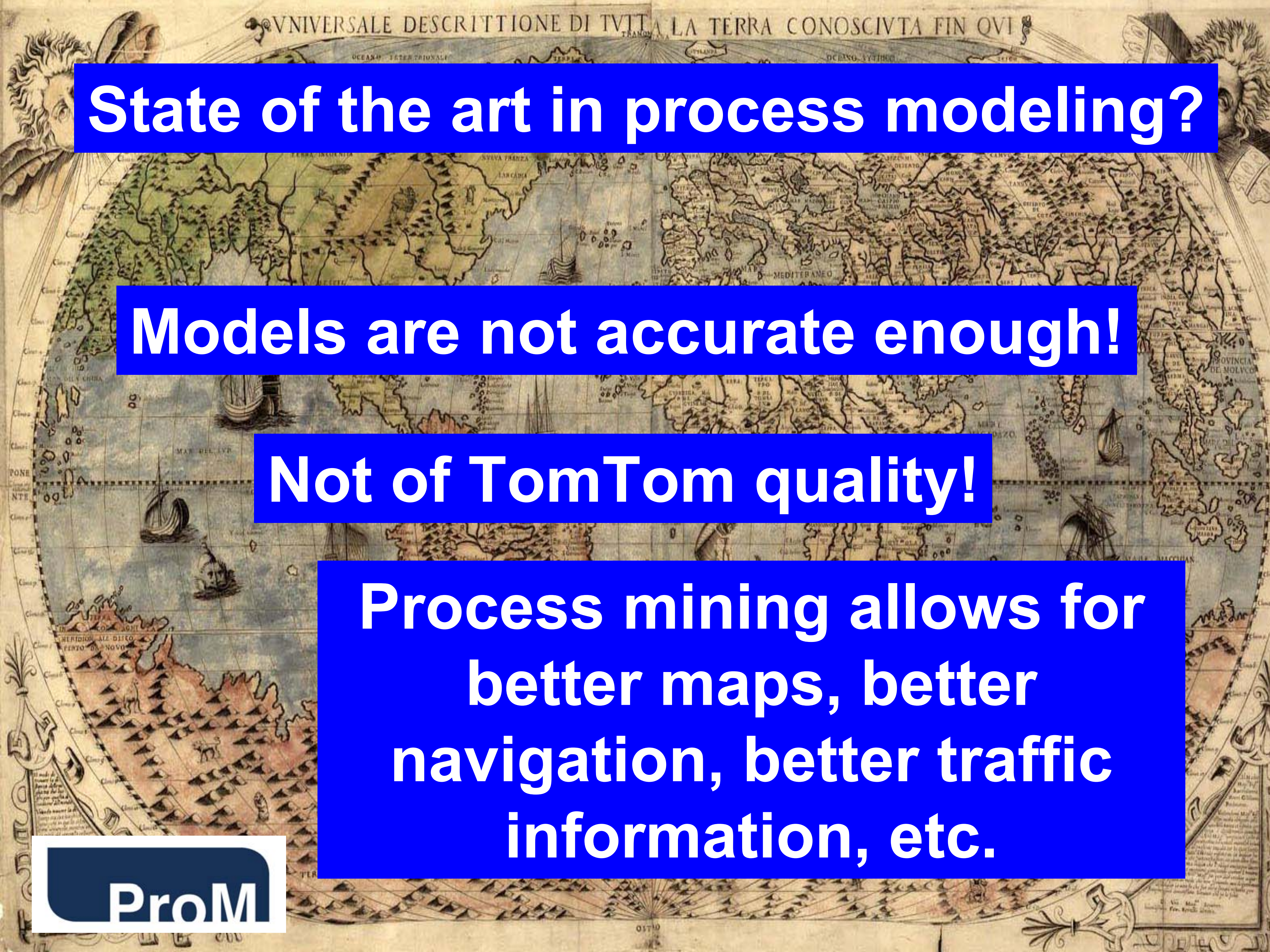
TU/e Technische Universiteit Eindhoven University of Technology

Where innovation starts

Our Benchmark: TomTom

Today's information systems are really crappy compared to a TomTom system!

- Good maps?
- Navigation by PowerPoints?
- Traffic information?
- Where is the next fuel station?
- Who is in charge?
- Seamless zoom?
- Customizable views?
- When will the destination be reached?



State of the art in process modeling?

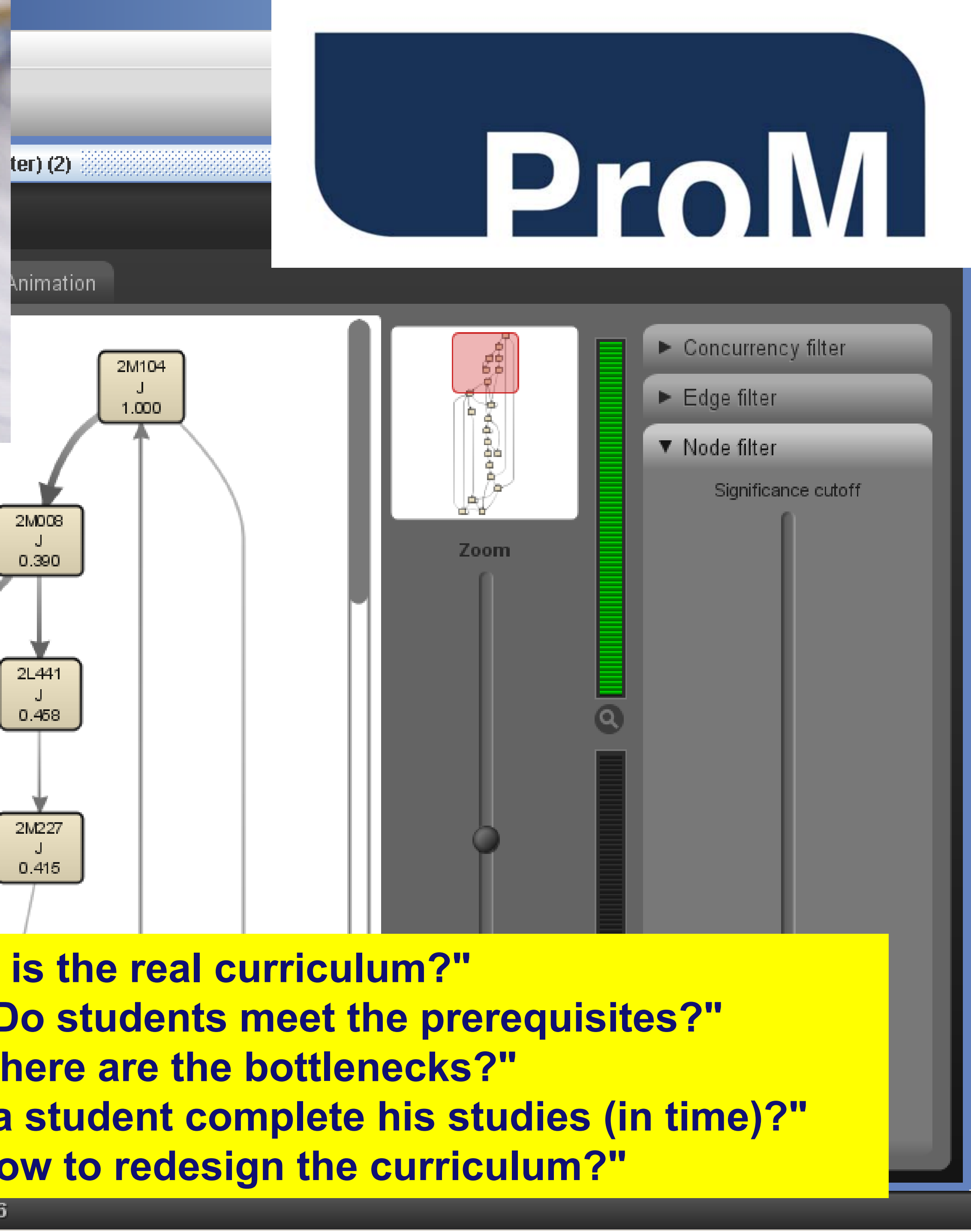
Models are not accurate enough!

Not of TomTom quality!

Process mining allows for better maps, better navigation, better traffic information, etc.

Process Mining

- Process discovery: "What is really happening?"
- Conformance checking: "Do we do what was agreed upon?"
- Performance analysis: "Where are the bottlenecks?"
- Process prediction: "Will this case be late?"
- Process improvement: "How to redesign this process?"
- Etc.



- Process discovery: "What is the real curriculum?"
- Conformance checking: "Do students meet the prerequisites?"
- Performance analysis: "Where are the bottlenecks?"
- Process prediction: "Will a student complete his studies (in time)?"
- Process improvement: "How to redesign the curriculum?"

Outline of tutorial

- **Part I : Introduction to Process Mining**
- **Part II : Process Discovery – The Alpha Algorithm**
- **Part III : Hands-on with ProM**

Part I

Introduction to Process Mining

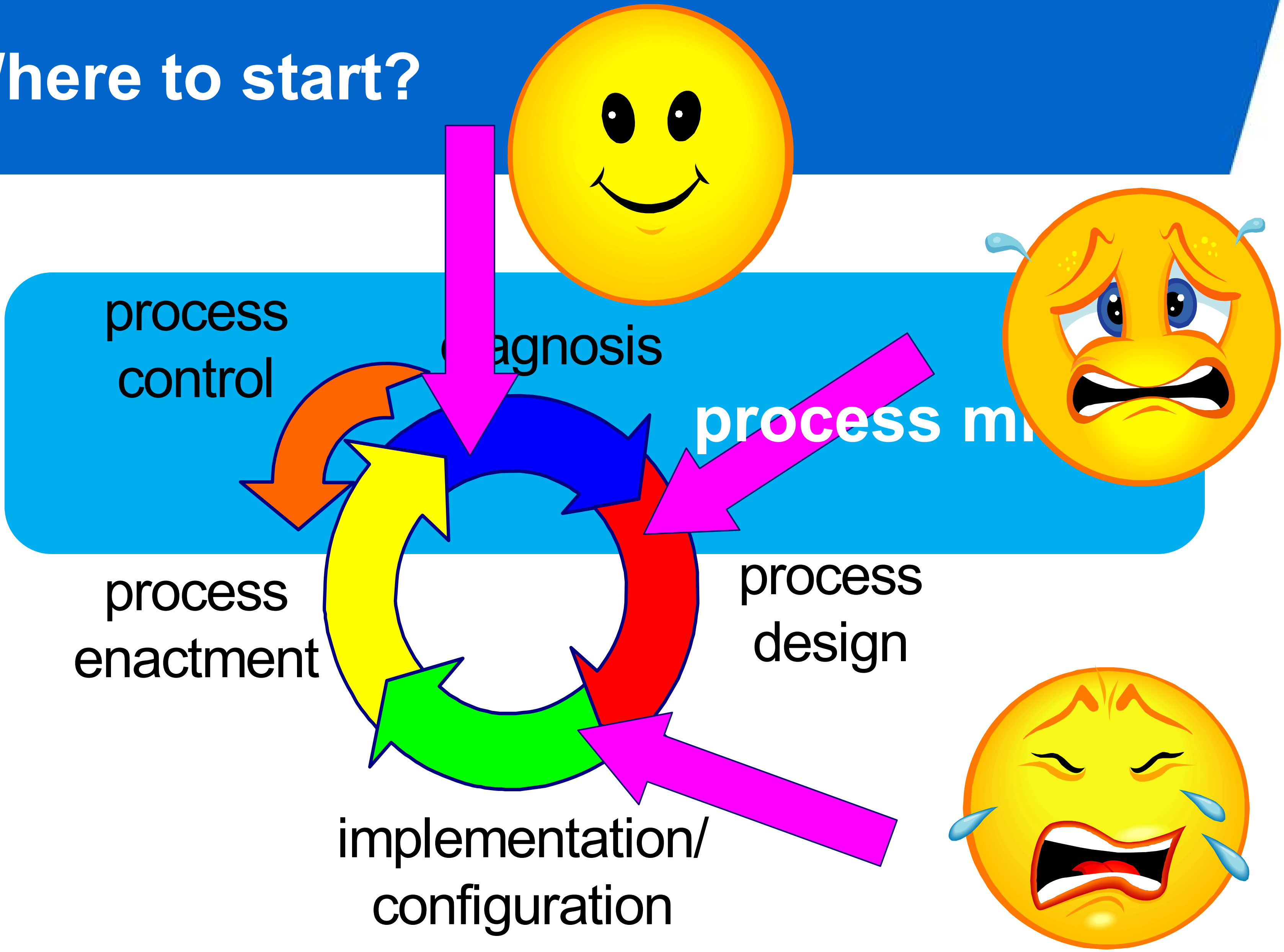
www.processmining.org

ProM

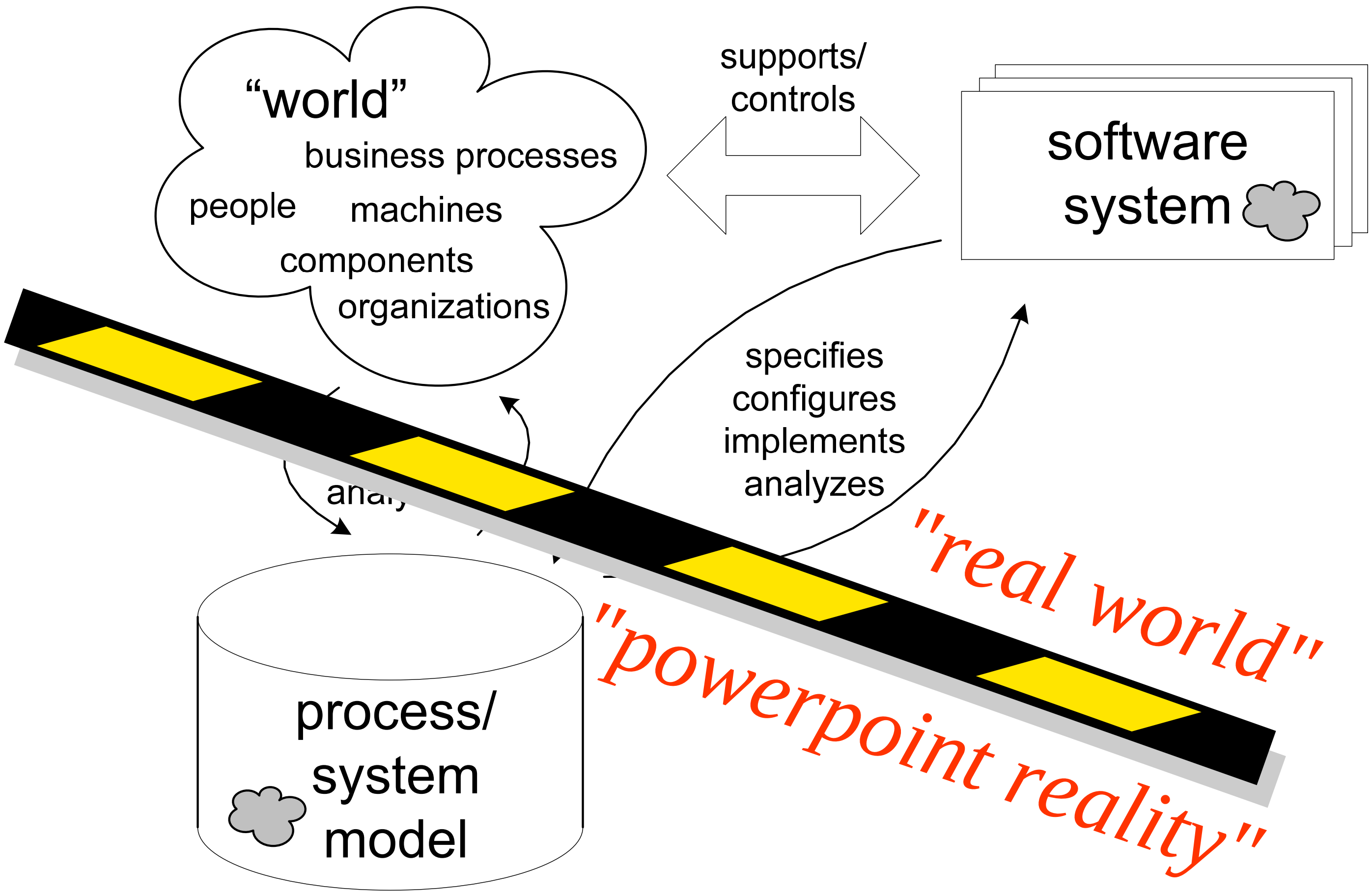
Technische Universiteit **Eindhoven** University of Technology

Where innovation starts

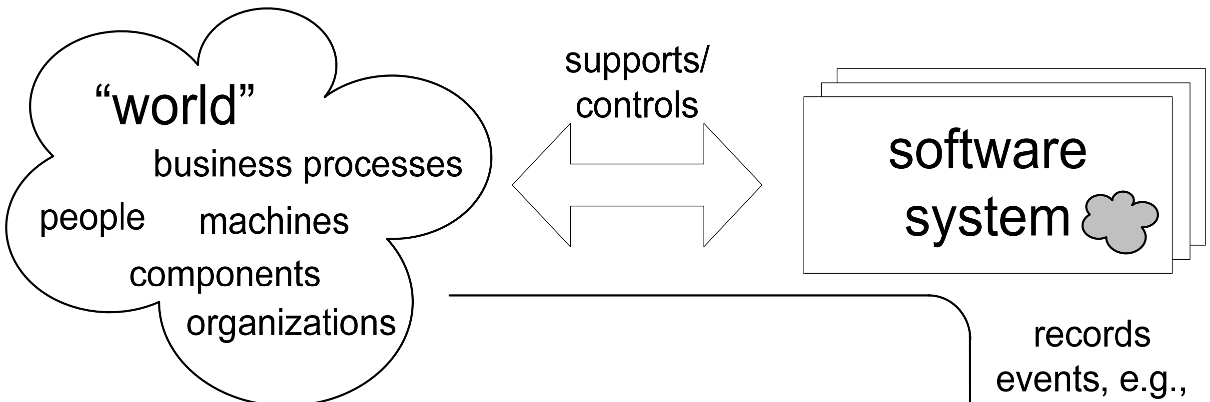
Where to start?

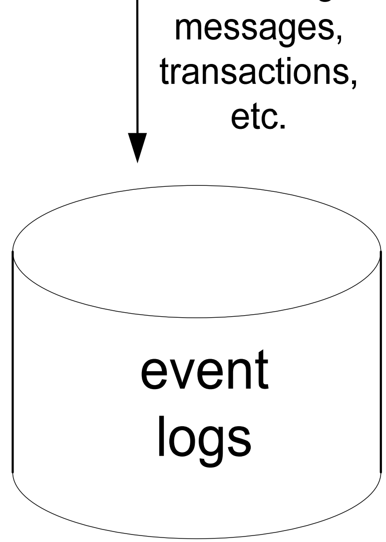


Role of models



Event logs are a reflection of reality





Examples:

staffware.
THE POWER OF PROCESS

IBM

FILENET
An IBM® Company

océ

Pallas Athena

PHILIPS
sense and simplicity

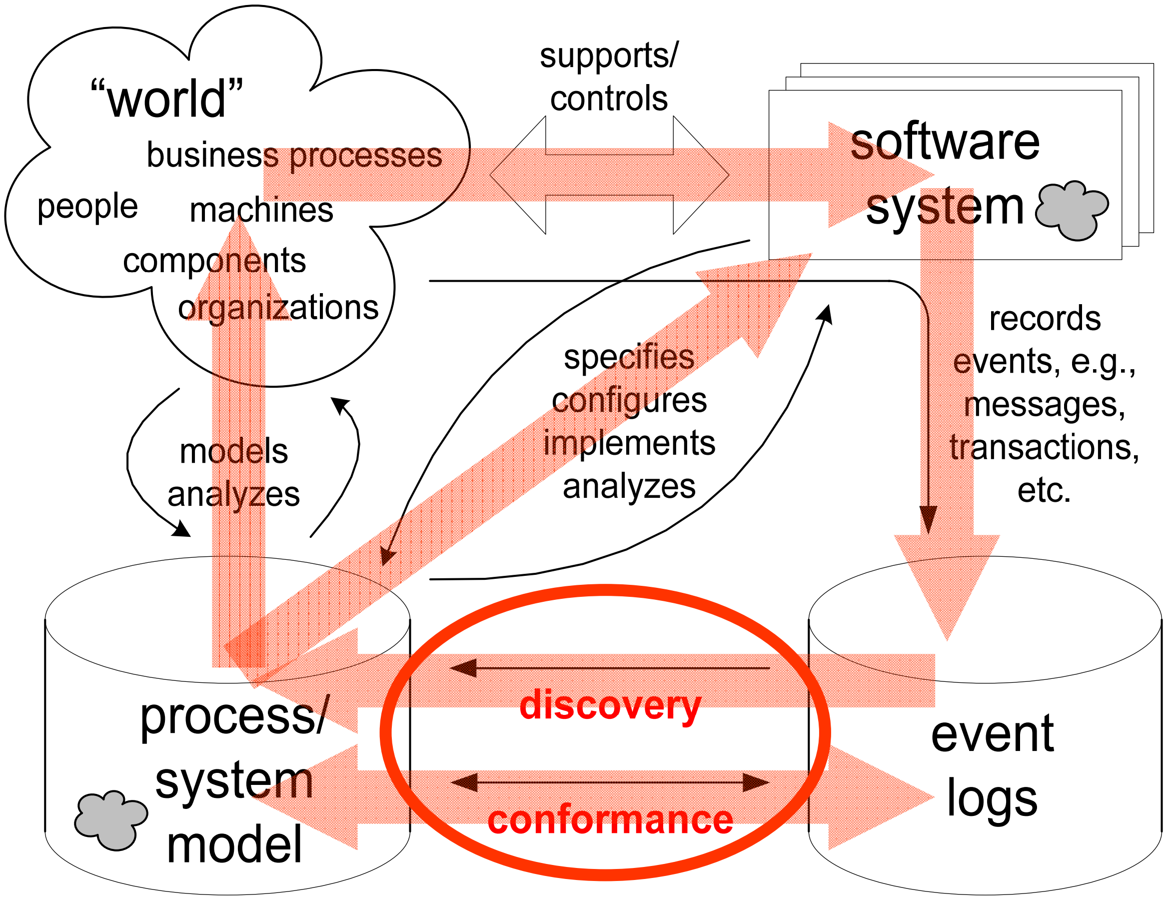
SAP

ORACLE

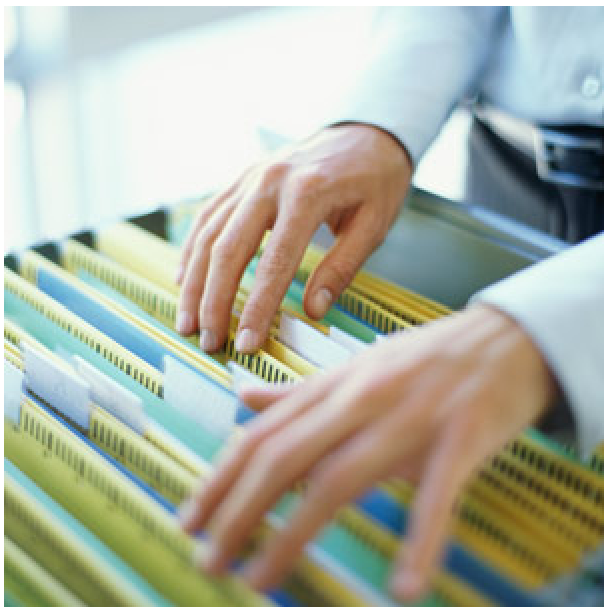
ASML

WebSphere. software

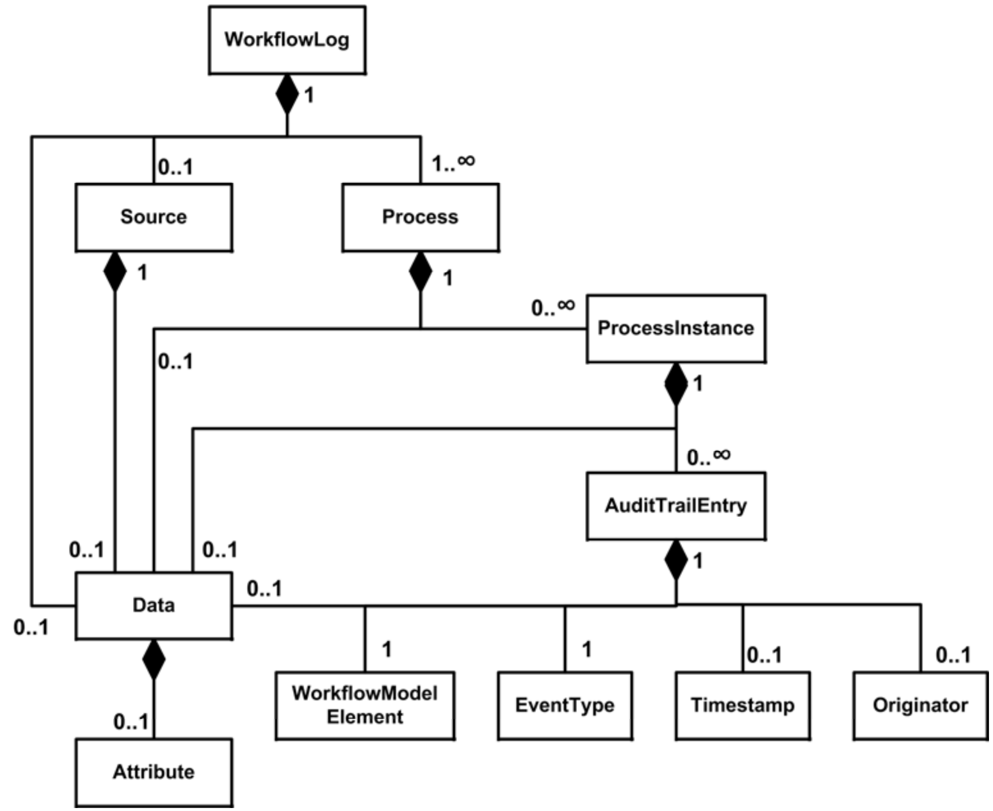
Process mining: Linking events to models



Starting point: event logs

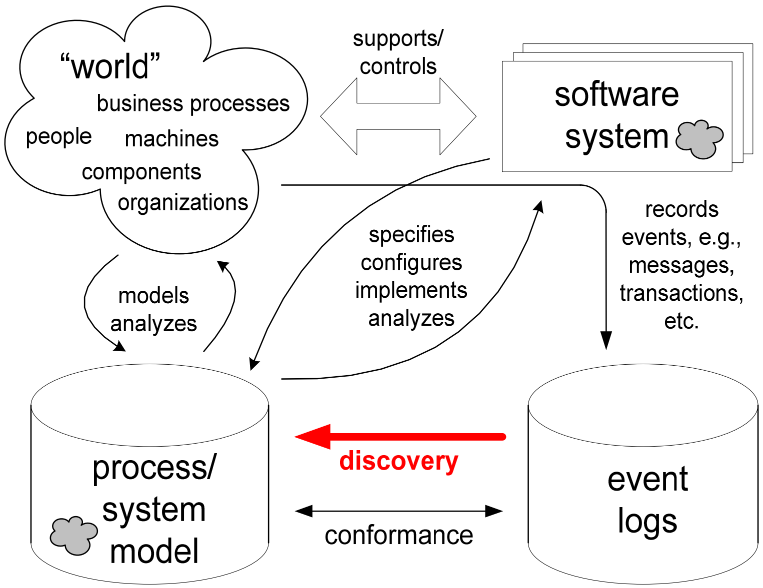


event logs, audit trails, databases, message logs, etc.



unified event log (MXML)

Discovery



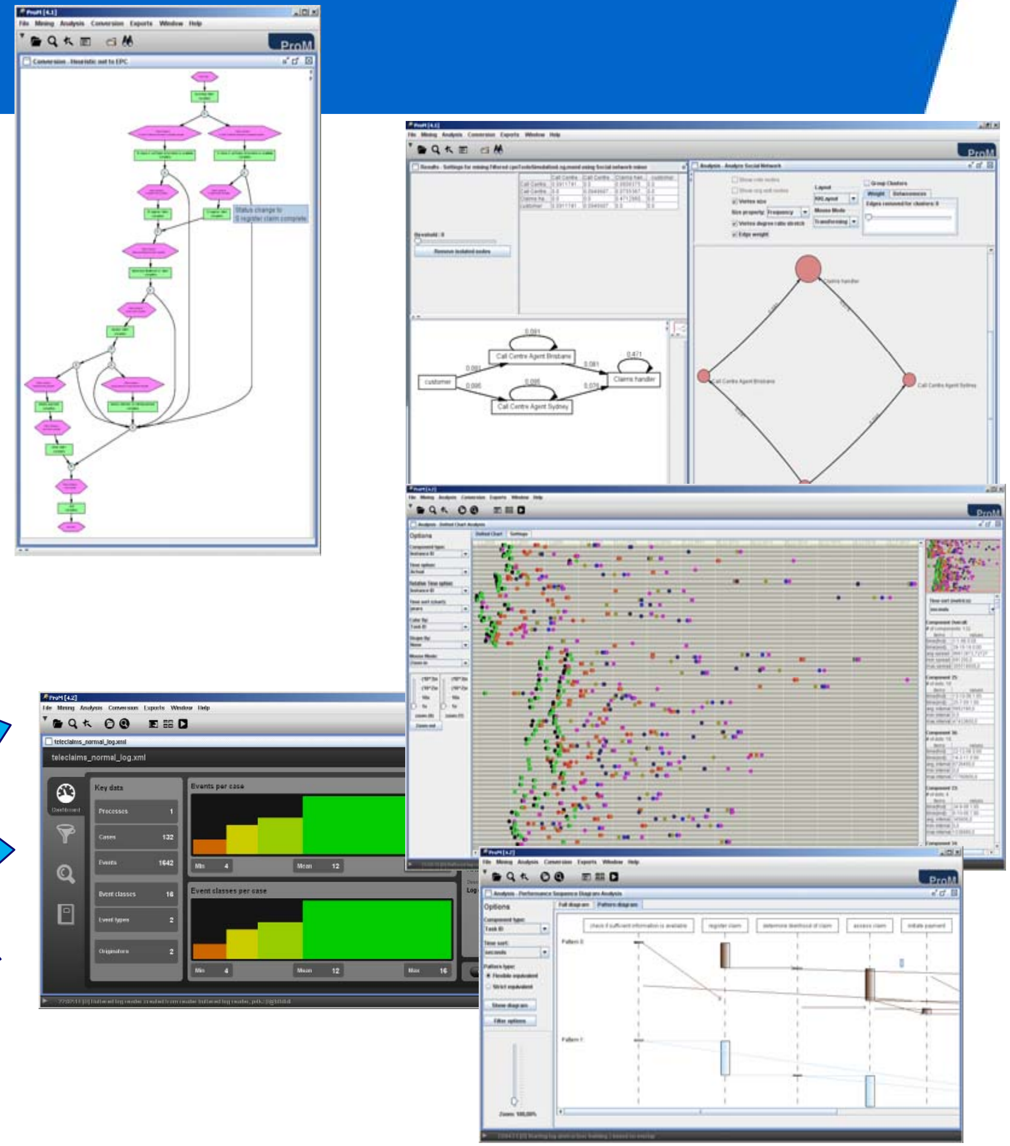
"world"
business processes
people
machines
components
organizations
supports/
controls
software
system
records
events, e.g.,
messages,
transactions,
etc.
specifies
configures
implements
analyzes
models
analyzes
process/
system
model
discovery
conformance
event
logs

What to discover?

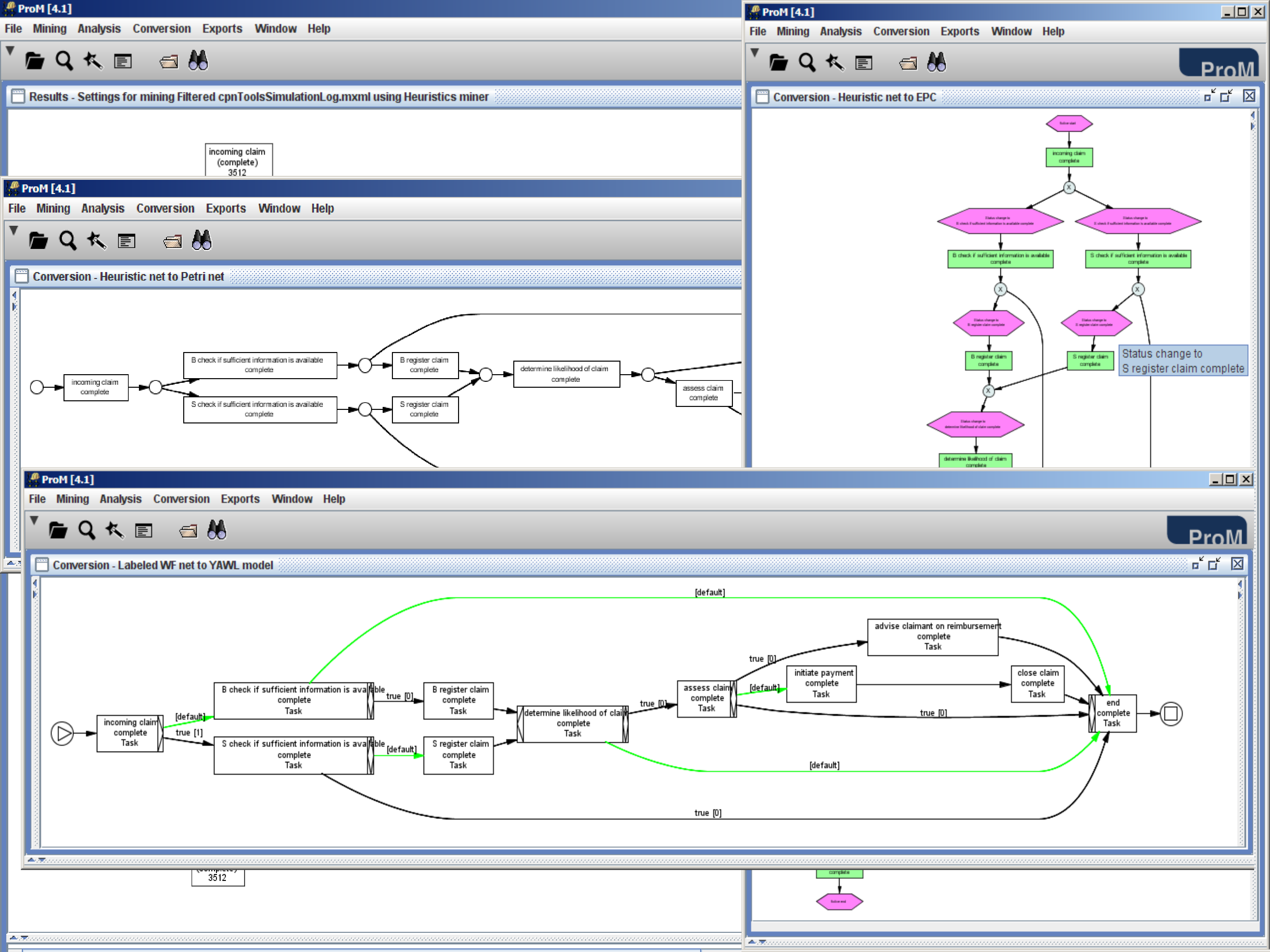
- process models (Petri nets, EPCs, BPMN, etc.),
- organizational models,
- social networks,
- sequence diagrams,
- business rules,
- bottlenecks,
- simulation models,
- etc.

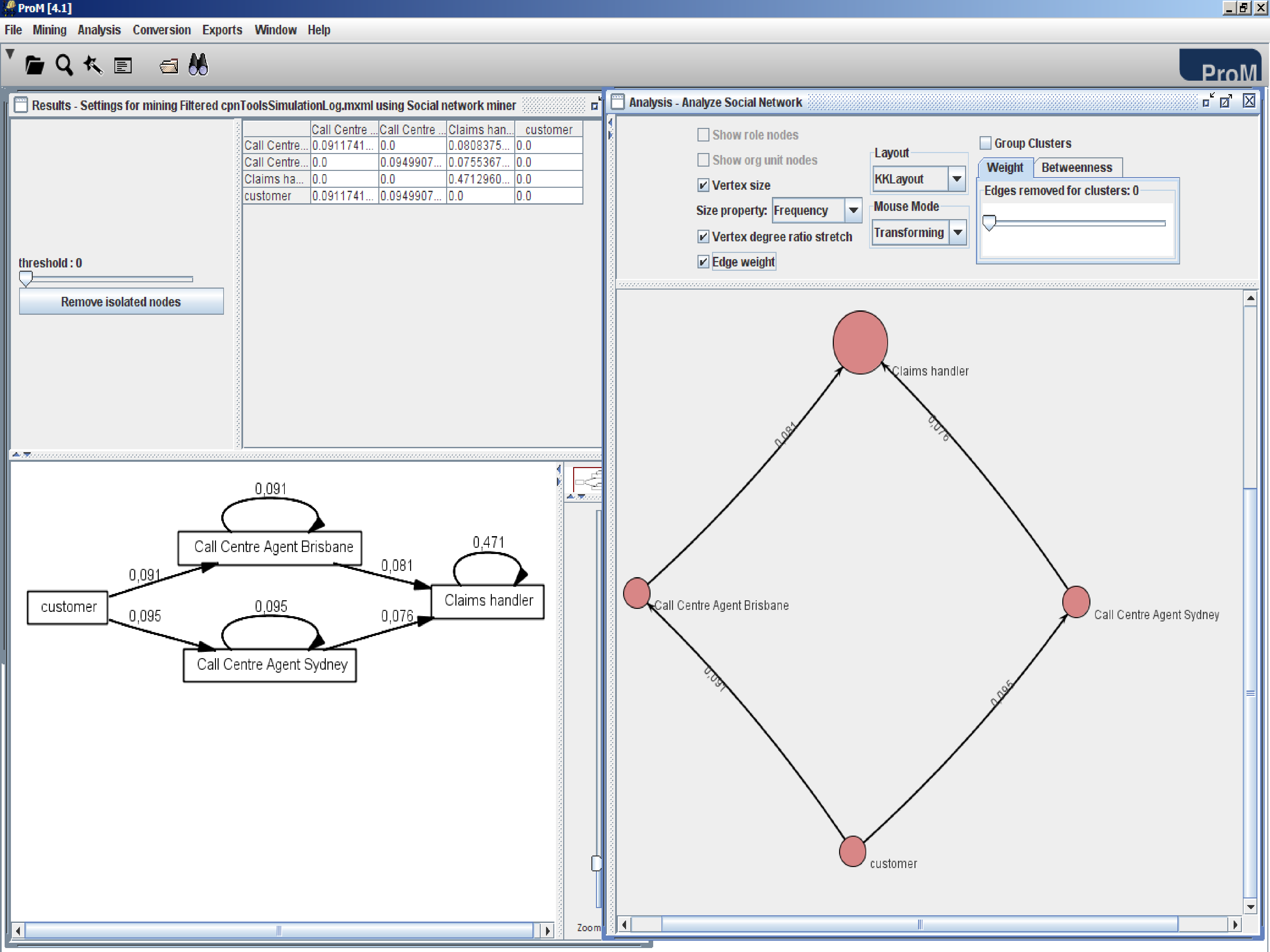
i.e., beyond "slice and dice" and showing KPIs on a dashboard ...

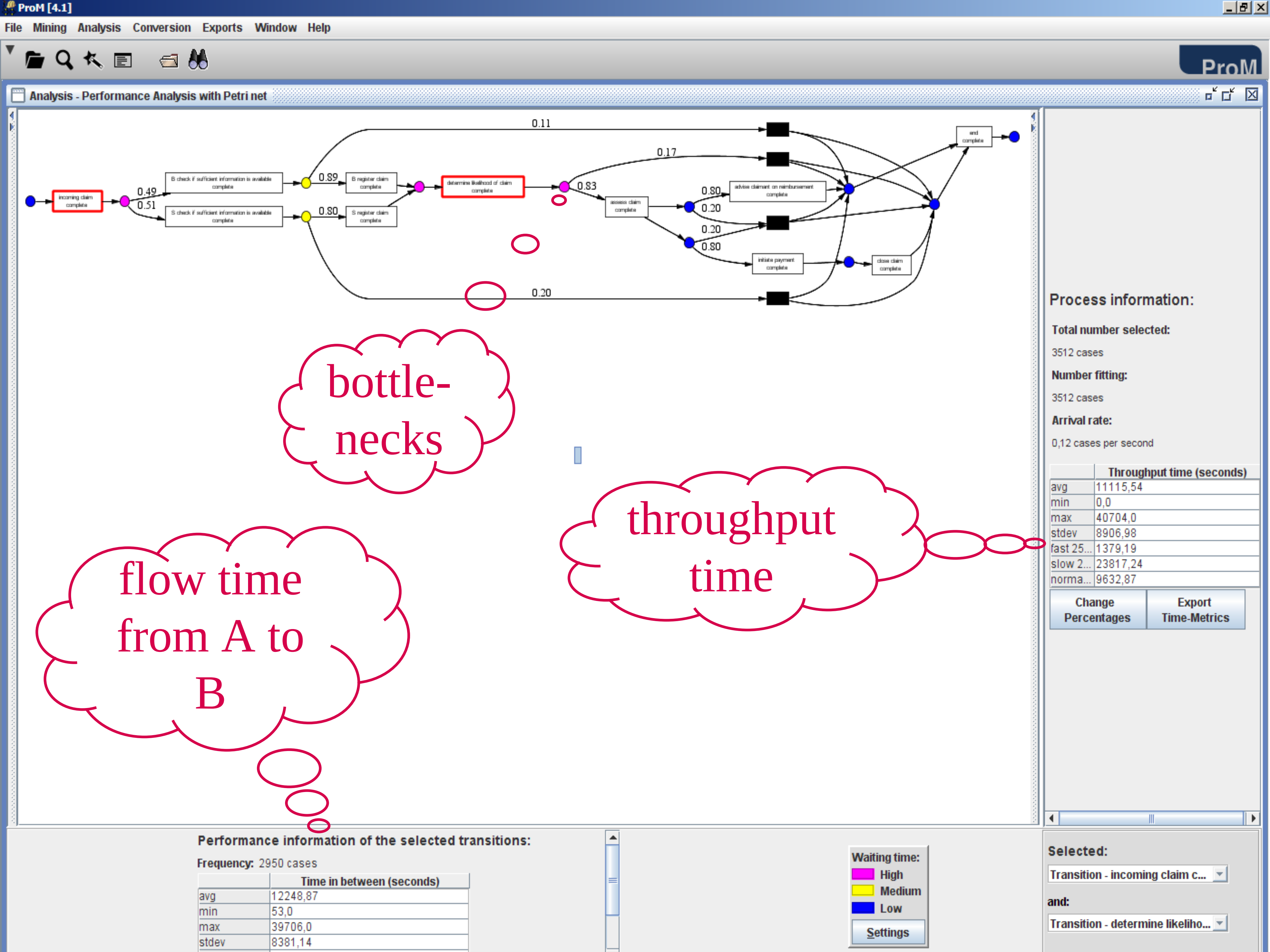
MXML Log
- **instances:** 3512
- **audit trail entries:** 46138

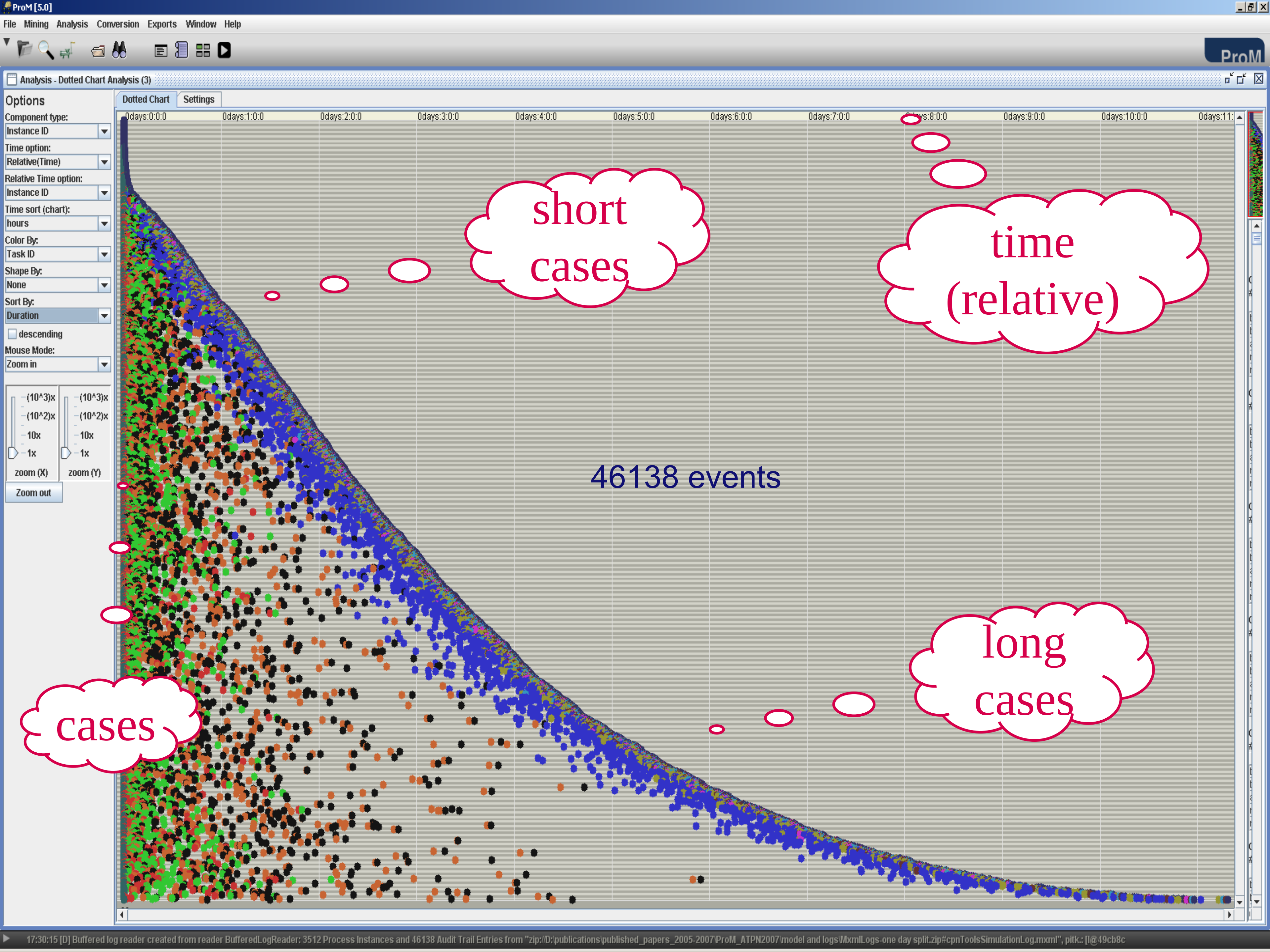


ProM supports +40 types of model discovery!


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
customer
Call Centre Agent Brisbane
Call Centre Agent Sydney
Claims handler
0,091
0,095
0,081
0,076
0,471
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
Call Centre Agent Brisbane
Call Centre Agent Sydney
customer
0,081
0,076
0,091
0,095

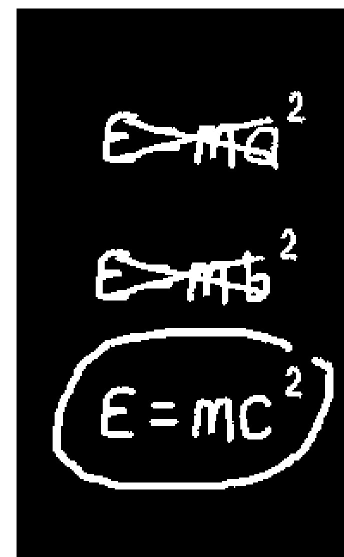

ProM [4.1]
File Mining Analysis Conversion Exports Window Help
ProM
Analysis - Performance Analysis with Petri net
0.11
0.17
0.89
0.49
0.51
0.80
0.83
0.80
0.20
0.20
0.80
0.20
bottle-necks
throughput time
flow time from A to B
Process information:
Total number selected:
3512 cases
Number fitting:
3512 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...



ProM [5.0]
File Mining Analysis Conversion Exports Window Help
Analysis - Dotted Chart Analysis (3)
Options
Component type:
Instance ID
Time option:
Relative(Time)
Relative Time option:
Instance ID
Time sort (chart):
hours
Color By:
Task ID
Shape By:
None
Sort By:
Duration
descending
Mouse Mode:
Zoom in
(10^3)x
(10^2)x
10x
1x
zoom (X)
zoom (Y)
Zoom out
Dotted Chart
Settings
0days:0:0:0
0days:1:0:0
0days:2:0:0
0days:3:0:0
0days:4:0:0
0days:5:0:0
0days:6:0:0
0days:7:0:0
0days:9:0:0
0days:10:0:0
short cases
time (relative)
46138 events
long cases
cases
17:30:15 [D] Buffered log reader created from reader BufferedLogReader: 3512 Process Instances and 46138 Audit Trail Entries from "zip:/D:\publications\published_papers_2005-2007\ProM_ATPN2007\model and logs\MxmlLogs-one day split.zip#cpnToolsSimulationLog.mxml", pitk.: [I@49cb8c

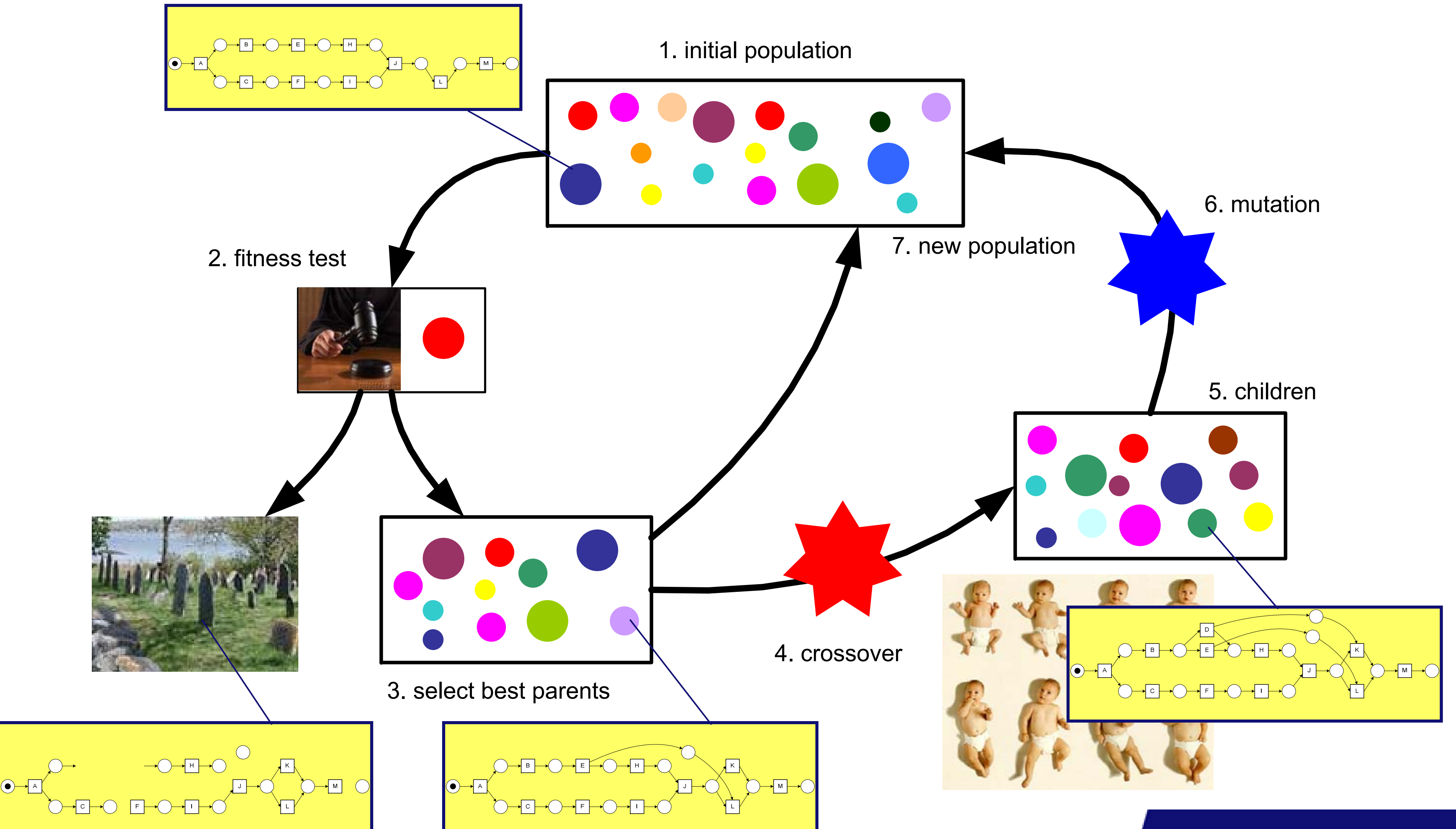
A bit of theory:
Process discovery techniques

- **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**



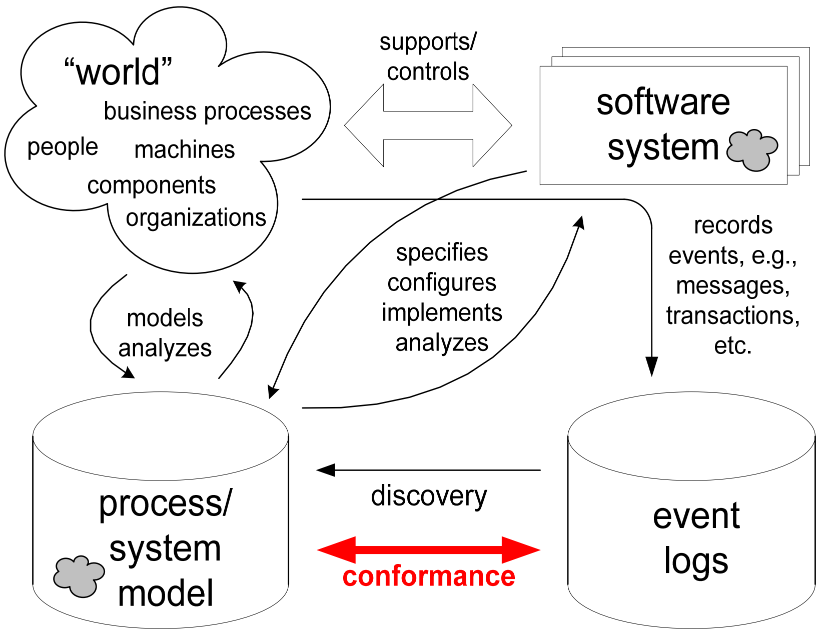
cf. www.processmining.org for an overview

Example: Genetic Mining



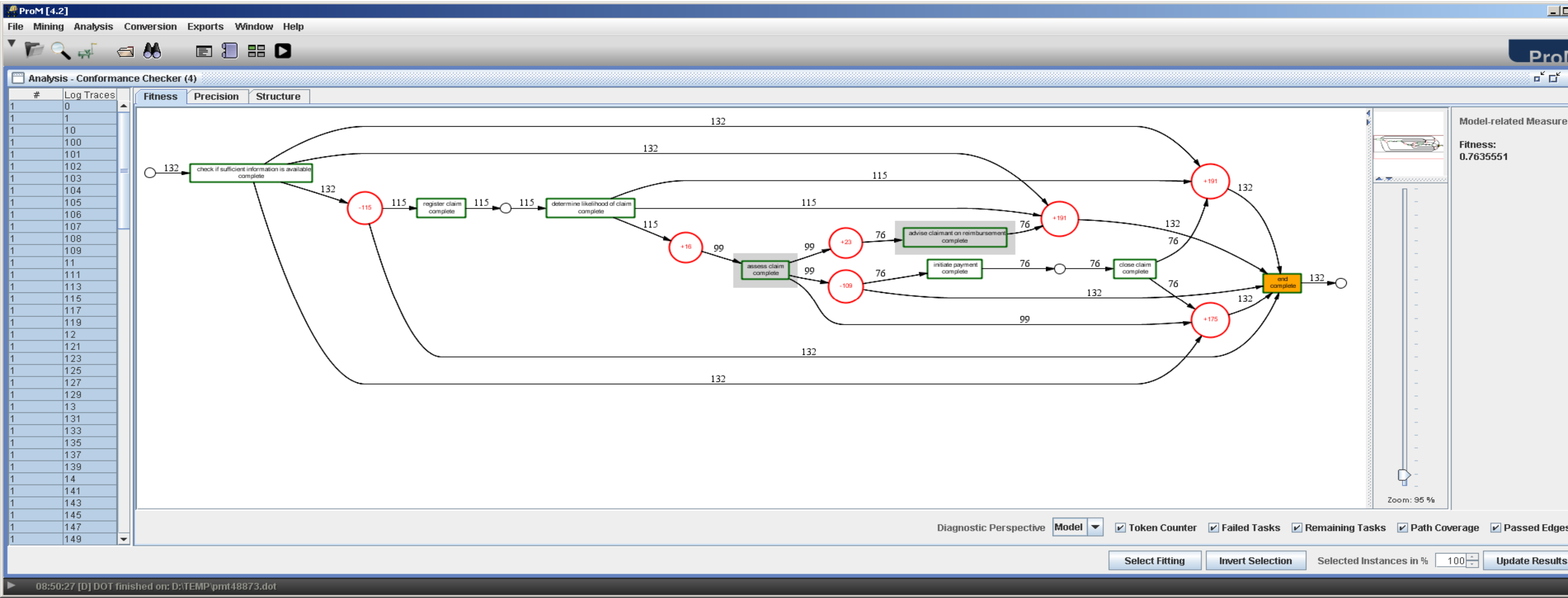
used in e.g. ProM, Futura Reflex, BPM|one

Conformance Checking



Conformance Checking

- Compare process model and event log: highlight deviations and measure conformance.
- Compare constraints/business rules and event logs: check e.g. the 4-eyes principle.

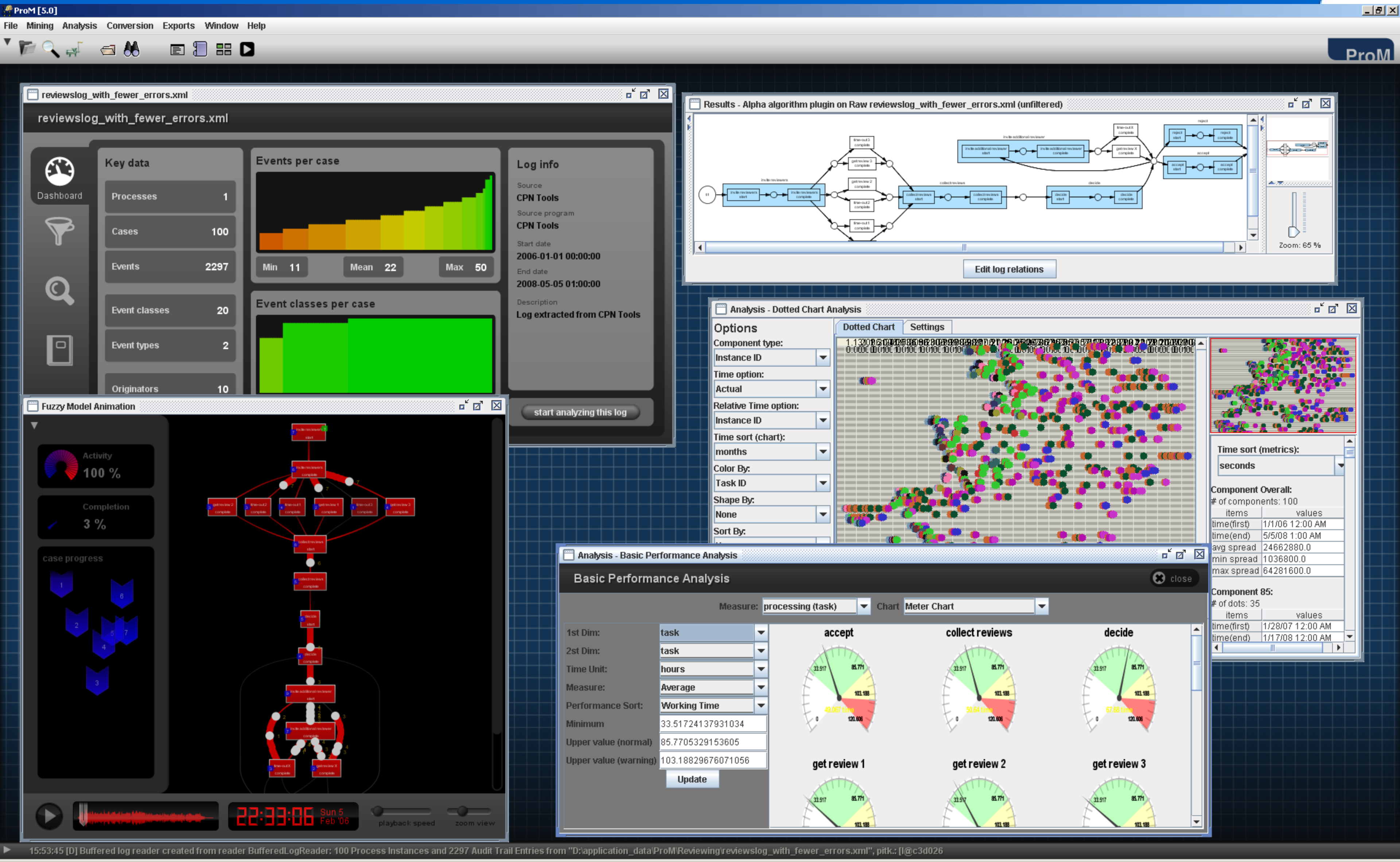


Process mining as a mirror ...

Tool support

- Open source initiative started in 2003 after several early prototypes.
- Common Public License (CPL).
- Current version: 5.0 (5.2)
- 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



Business Intelligence Tools?

- **Business Objects (SAP)**
- **Cognos Business Intelligence (IBM)**
- **Oracle Business Intelligence**
- **Hyperion (Oracle)**
- **SAS Business Intelligence**
- **Microsoft Business Intelligence**
- **SAP Business Intelligence (SAP BI)**
- **Jaspersoft (Open Source Business Intelligence)**
- **Pentaho BI Suite (Open Source)**
- **....**

- Dashboards, reports, scorecards, ...
- Slicing and dicing, data mining, ...

Process Mining Software

Futura Reflect

BPM|one

Comprehend

ARIS Process Performance Manager

Interstage Automated Business Process Discovery & Visualization

Process Discovery Focus

Enterprise Visualization Suite

Process Mining: Applications

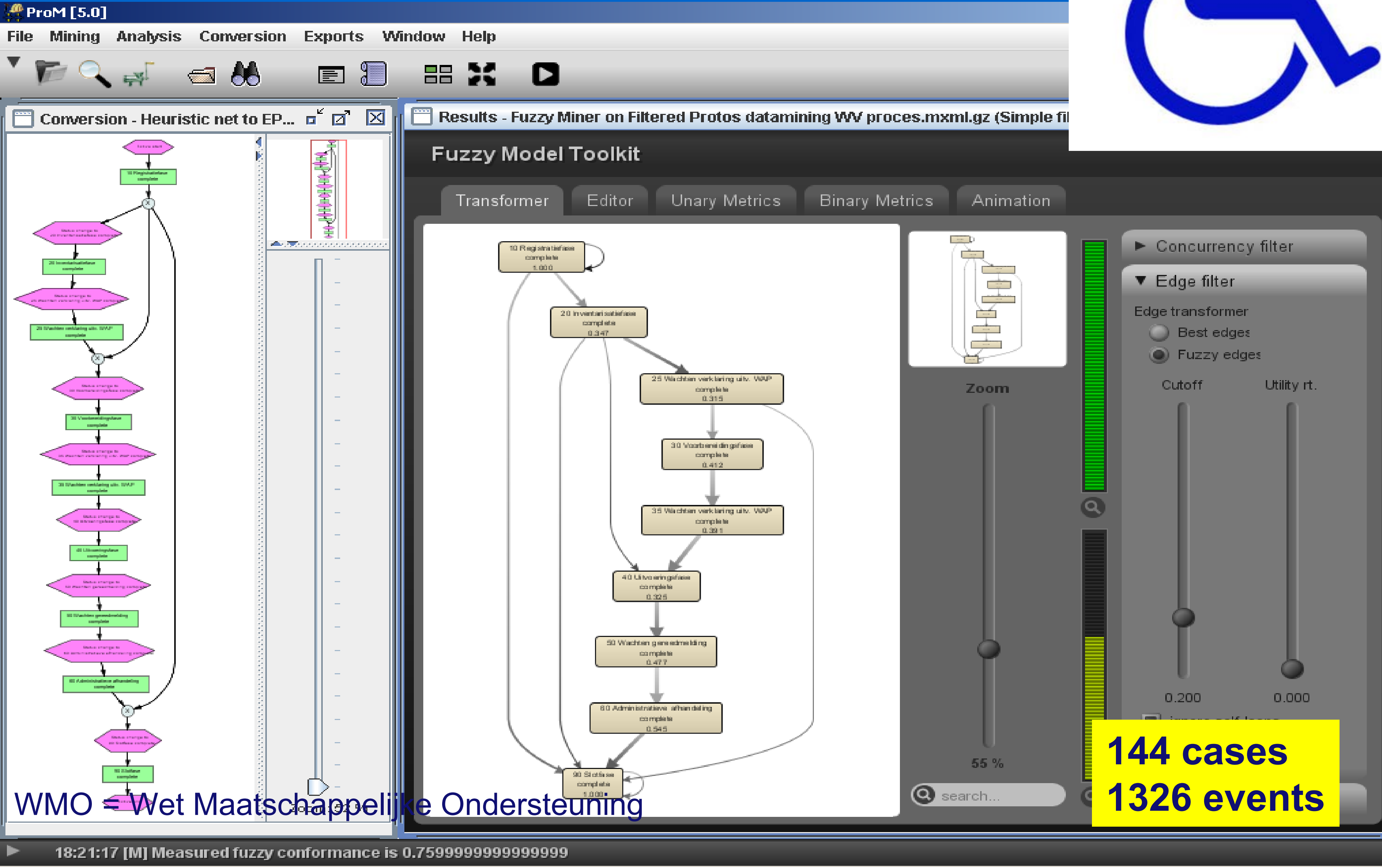
Technische Universiteit **Eindhoven** University of Technology

Where innovation starts

Where did we apply process mining?

- **Municipalities** (e.g., Alkmaar, Heusden, Harderwijk, etc.)
- **Government agencies** (e.g., Rijkswaterstaat, Centraal Justitieel Incasso Bureau, Justice department)
- **Insurance related agencies** (e.g., UWV)
- **Banks** (e.g., ING Bank)
- **Hospitals** (e.g., AMC hospital, Catharina hospital)
- **Multinationals** (e.g., DSM, Deloitte)
- **High-tech system manufacturers and their customers** (e.g., Philips Healthcare, ASML, Thales)
- **Media companies** (e.g. Winkwaves)
- **...**

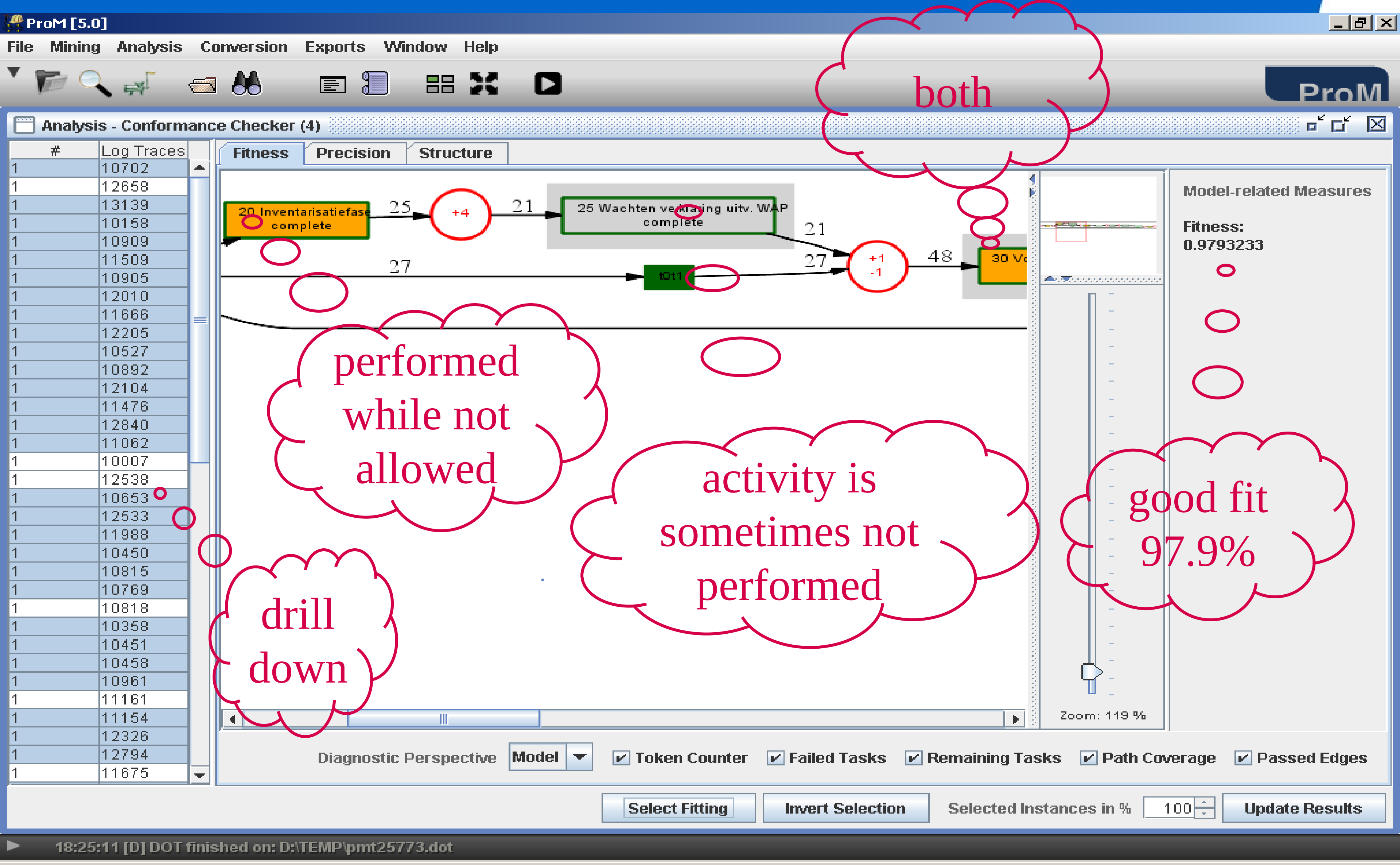
Example: WMO process of a Dutch Municipality



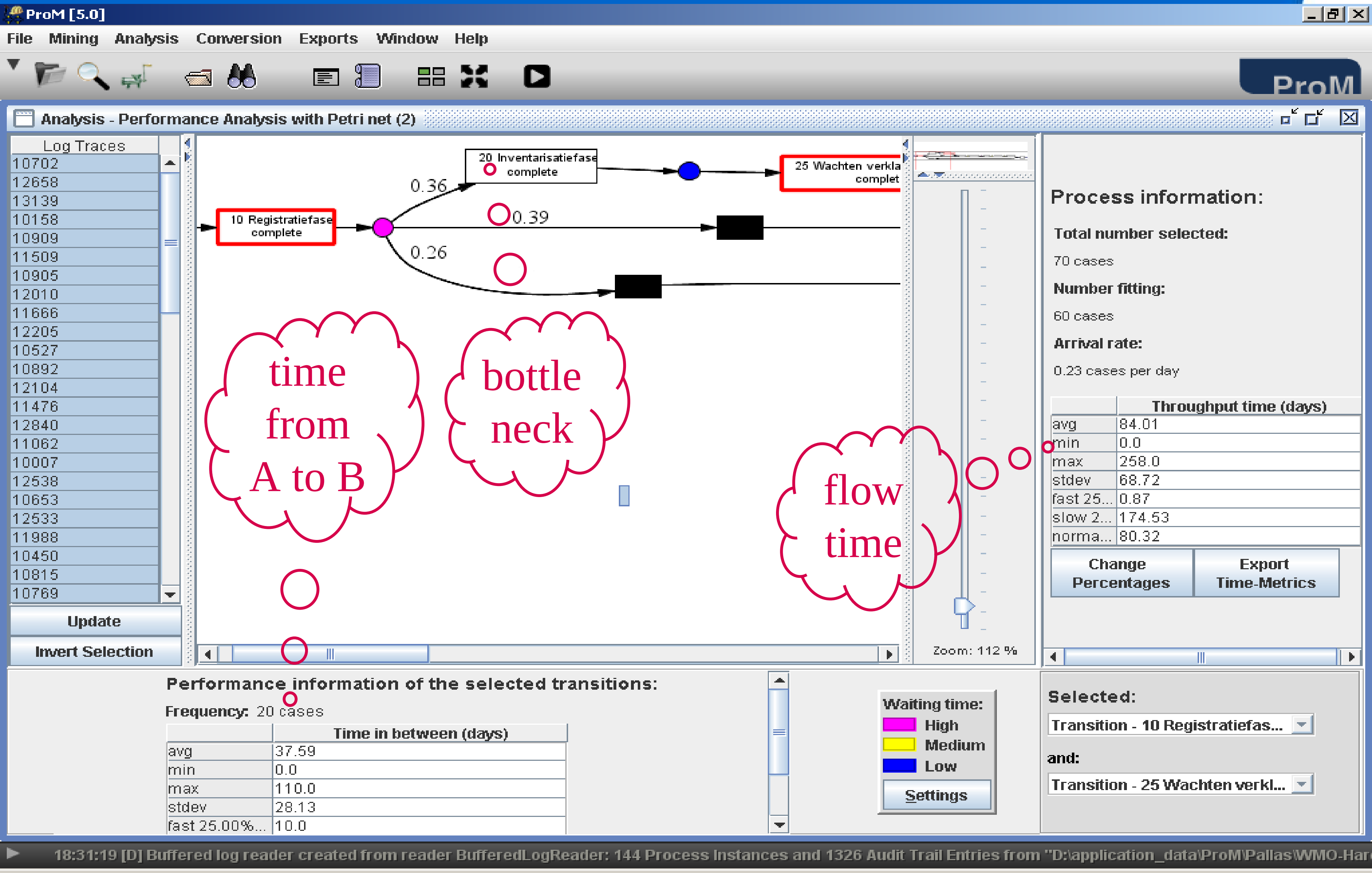
WMO = Wet Maatschappelijke Ondersteuning

144 cases
1326 events

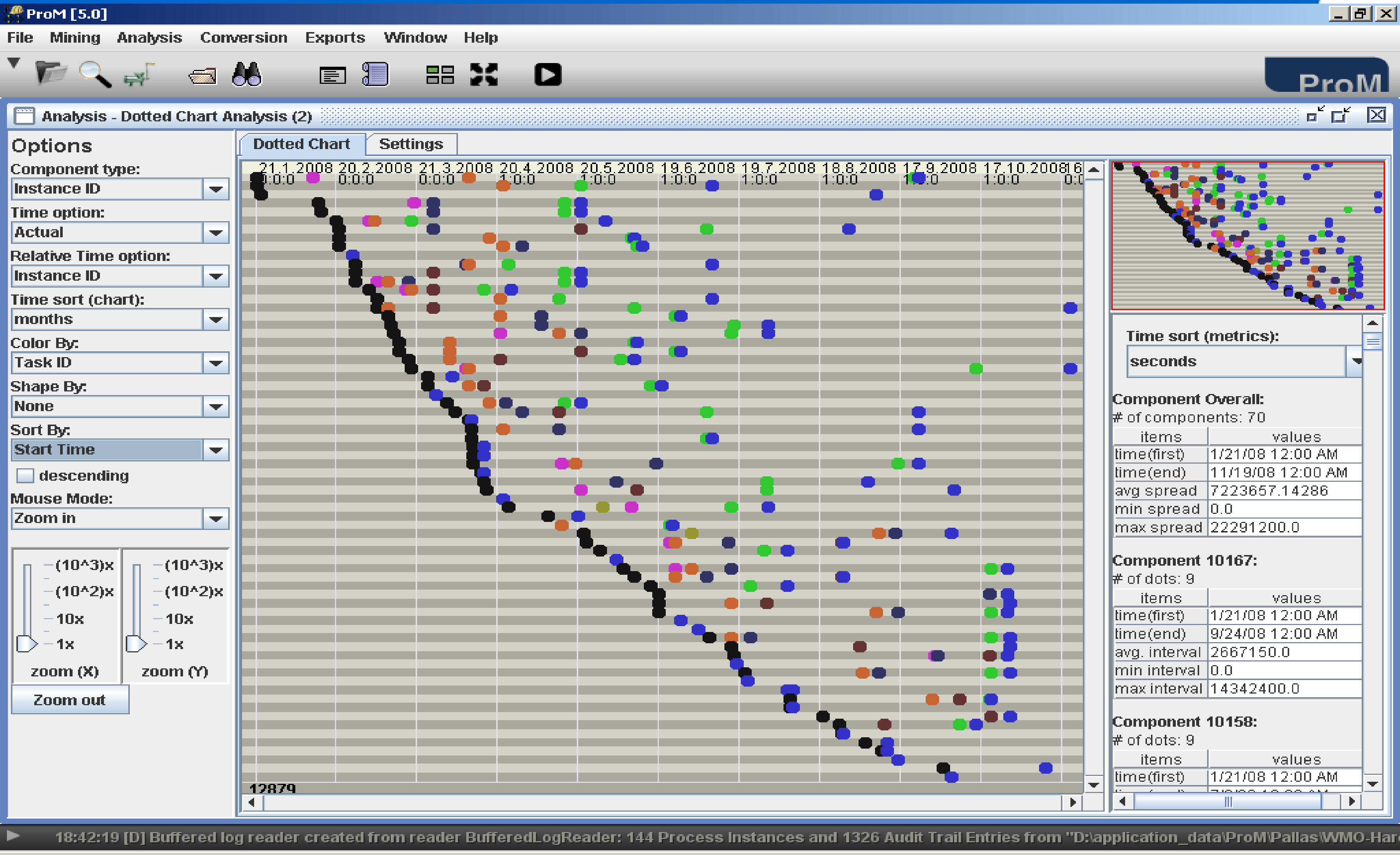
Conformance check of discovered model



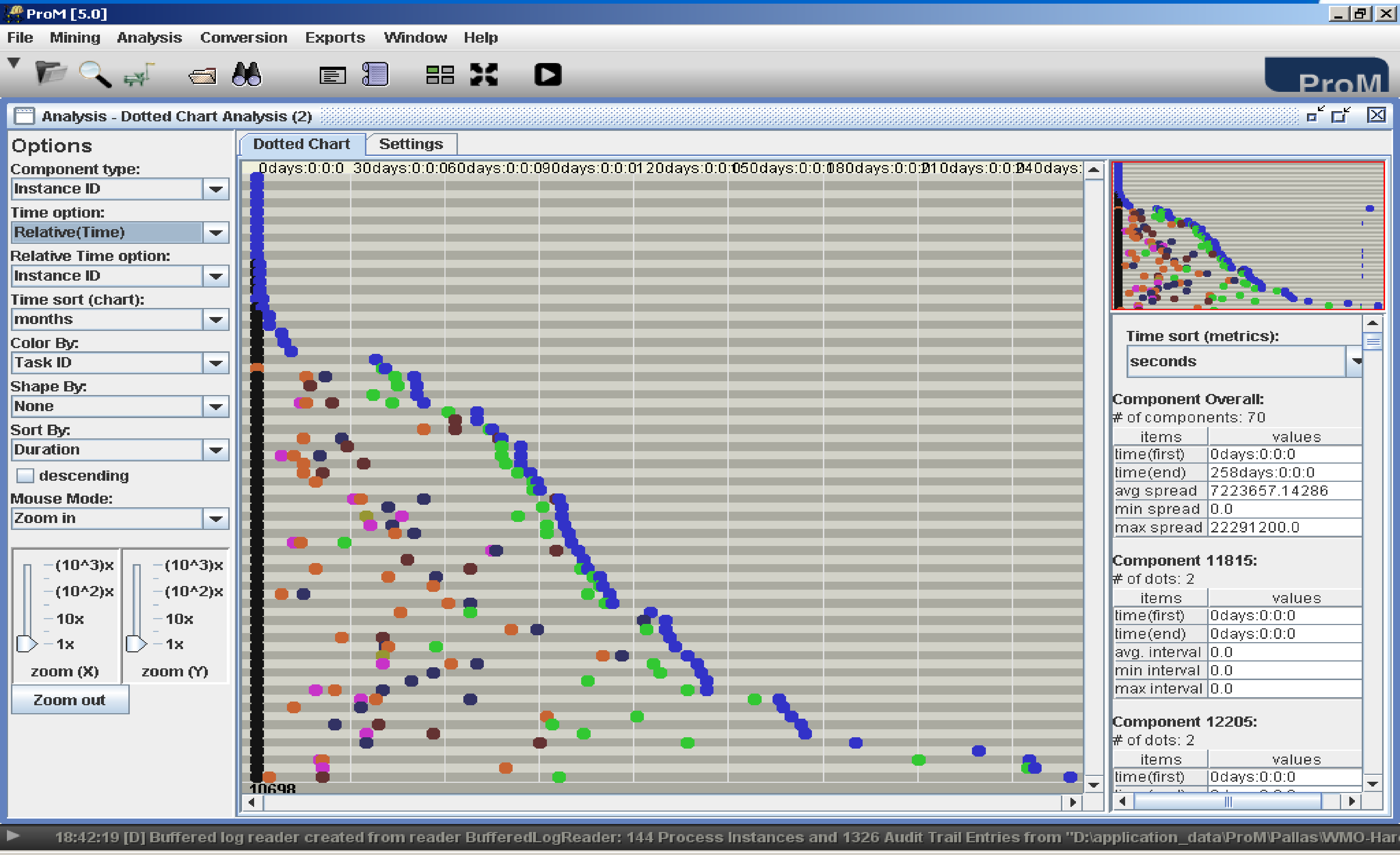
Performance analysis



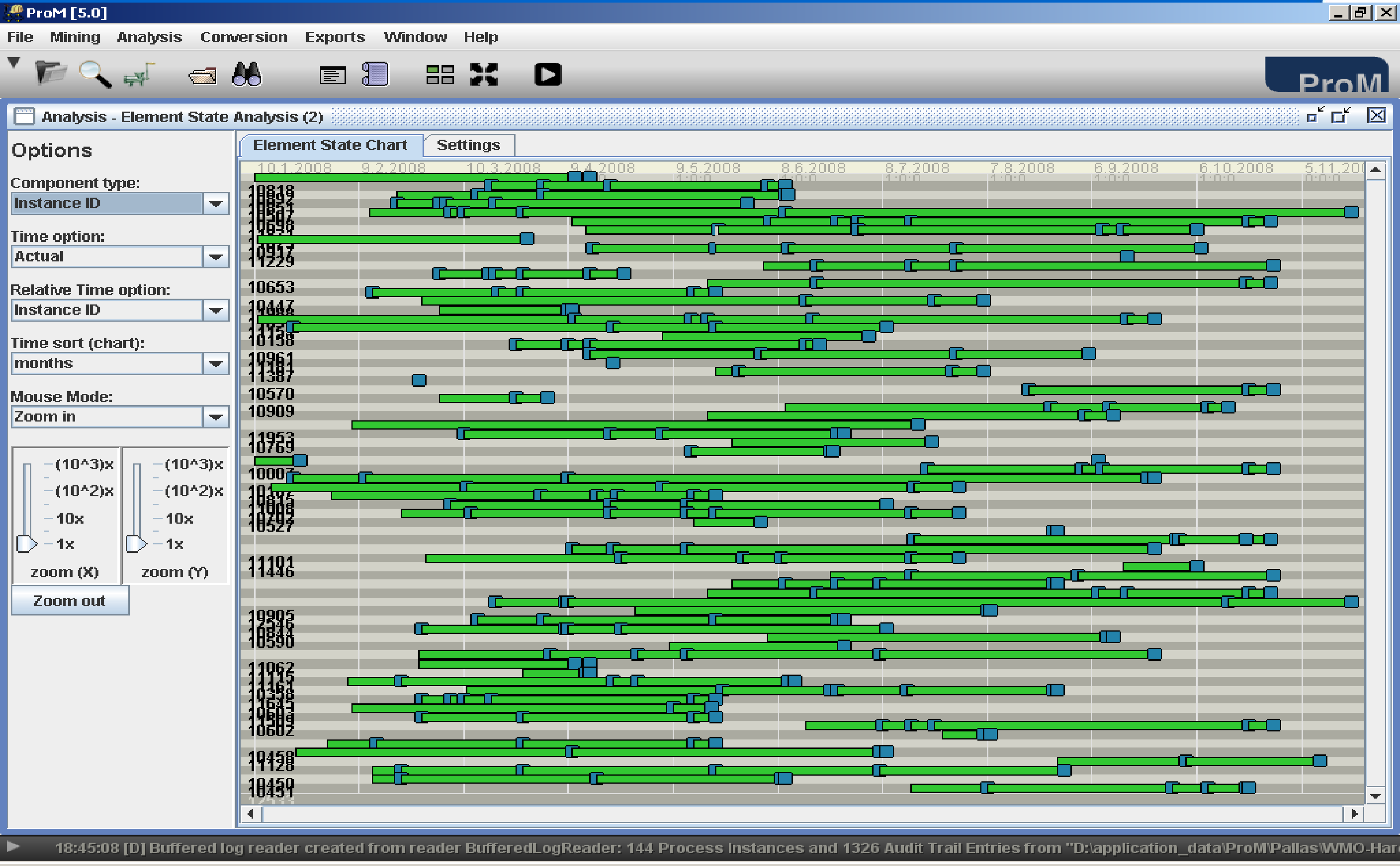
Events sorted by start time of case



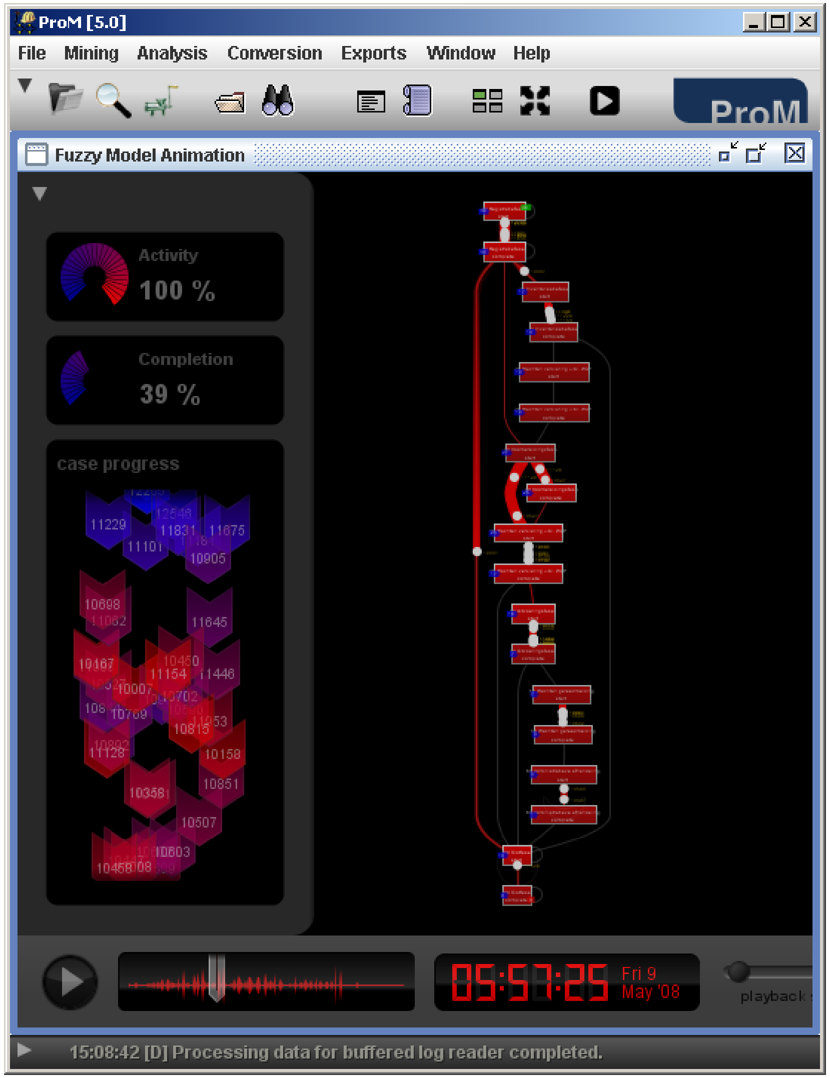
Events sorted by duration

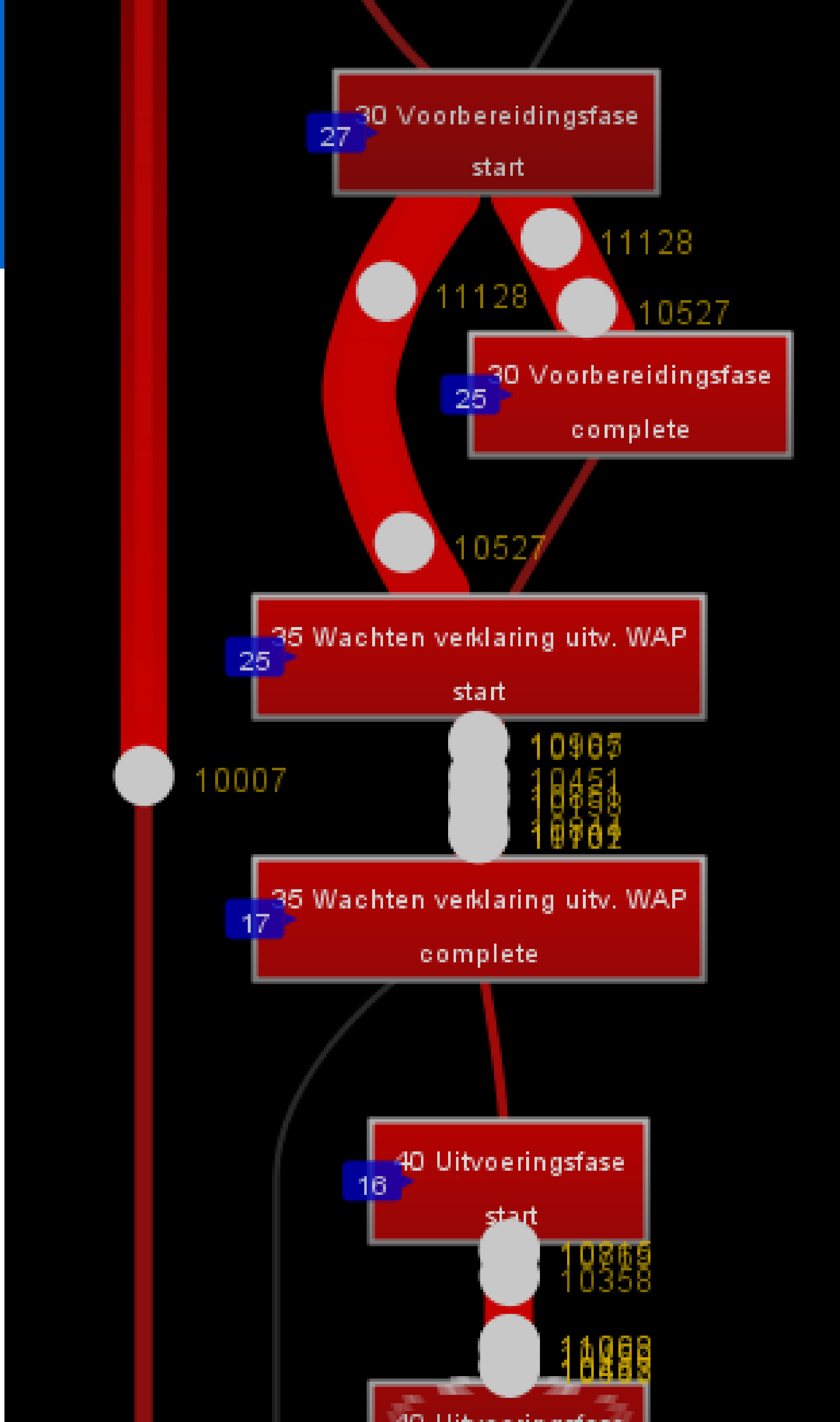


Idle time versus working time

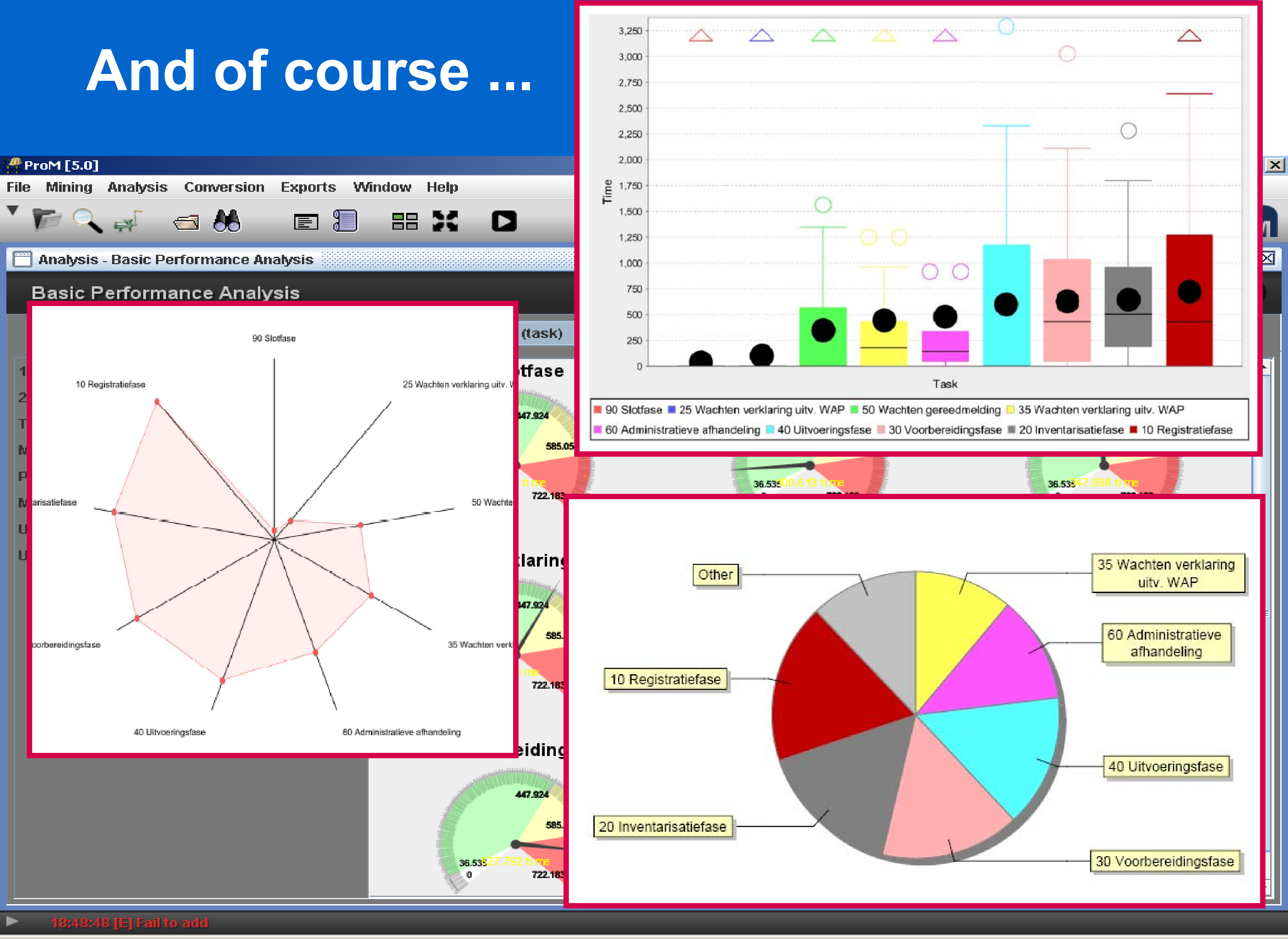


"Real" animation



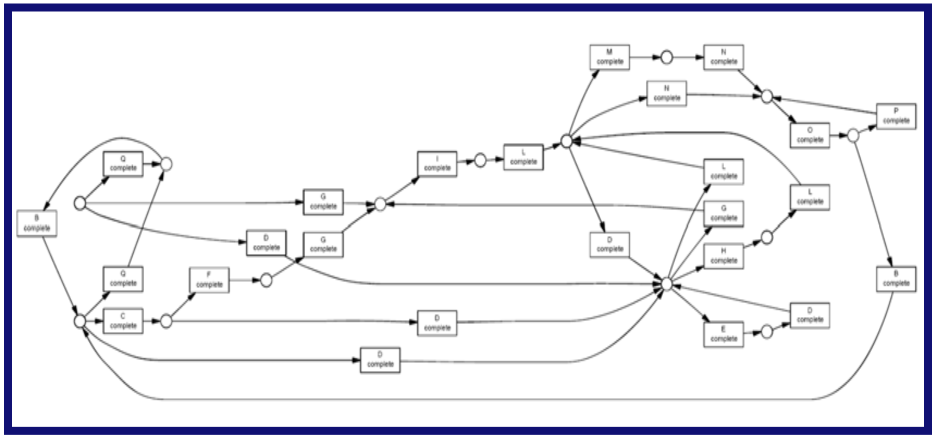


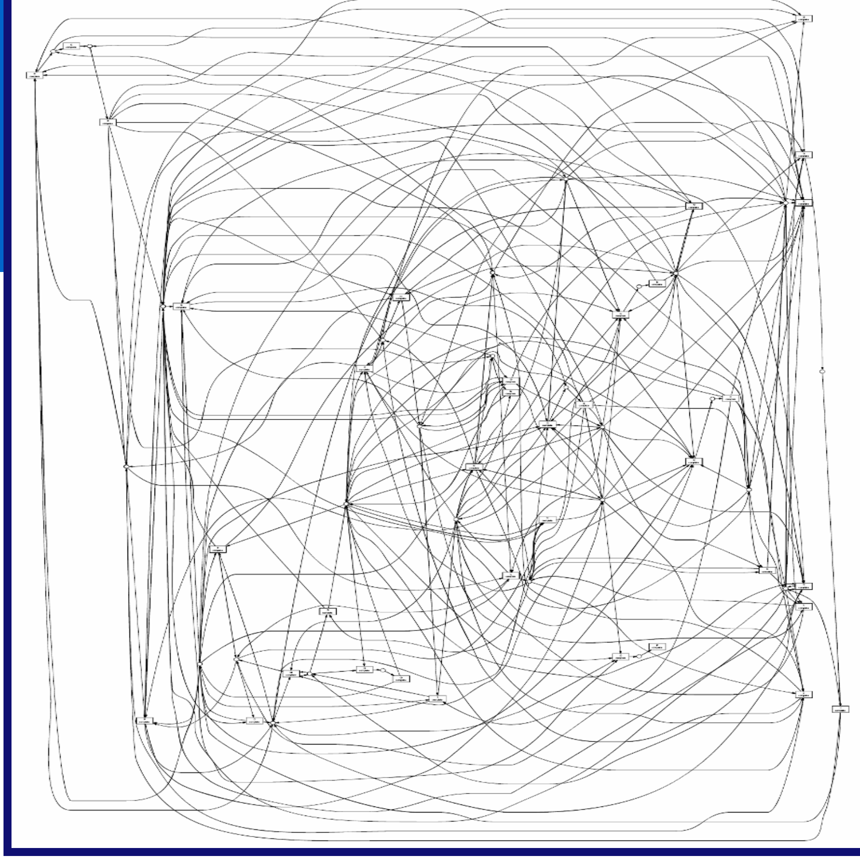
And of course ...



Reality ≠ PowerPoint (or Visio)

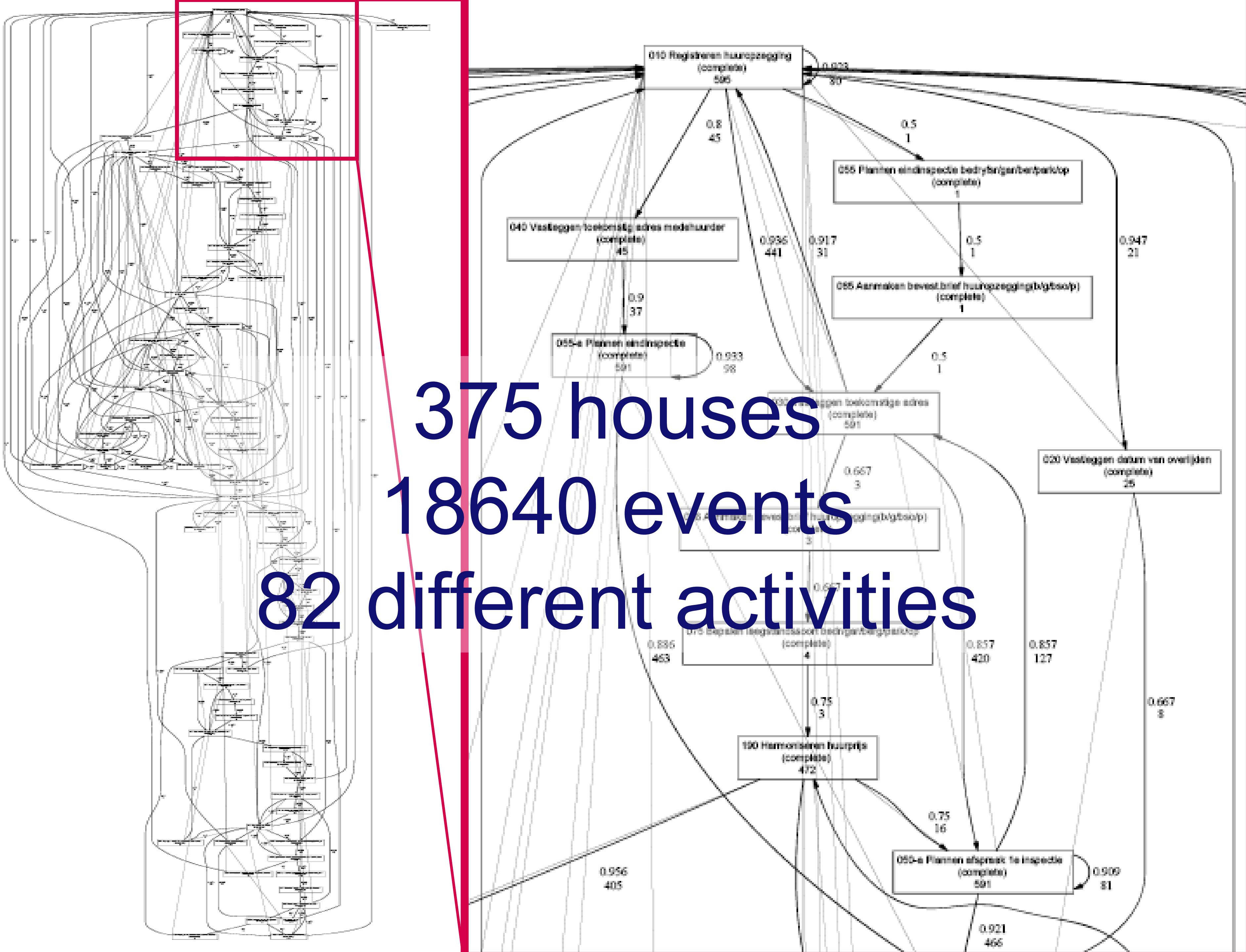
Process spectrum





structured (Lasagna)

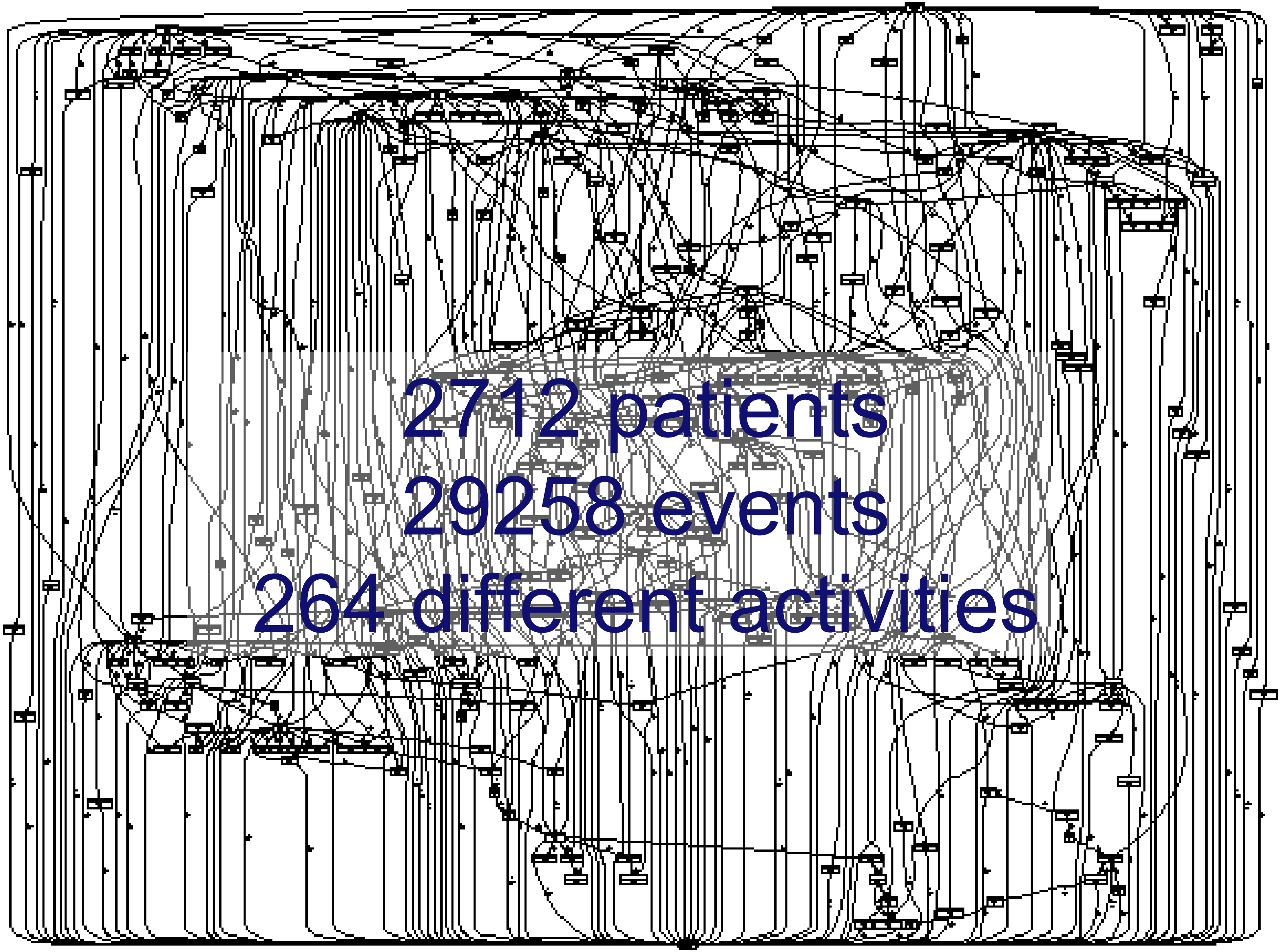
unstructured (Spaghetti)

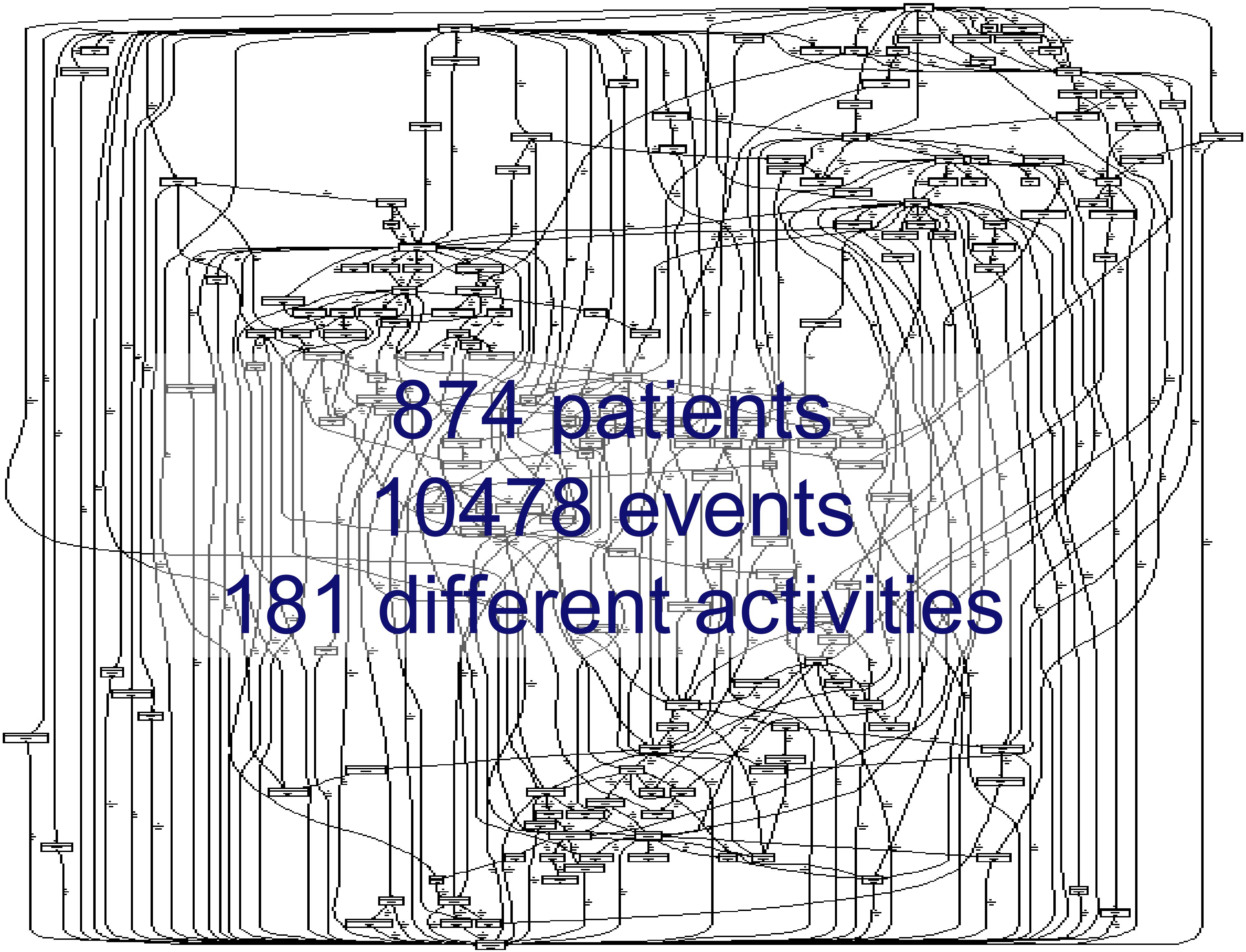


375 houses

18640 events

82 different activities

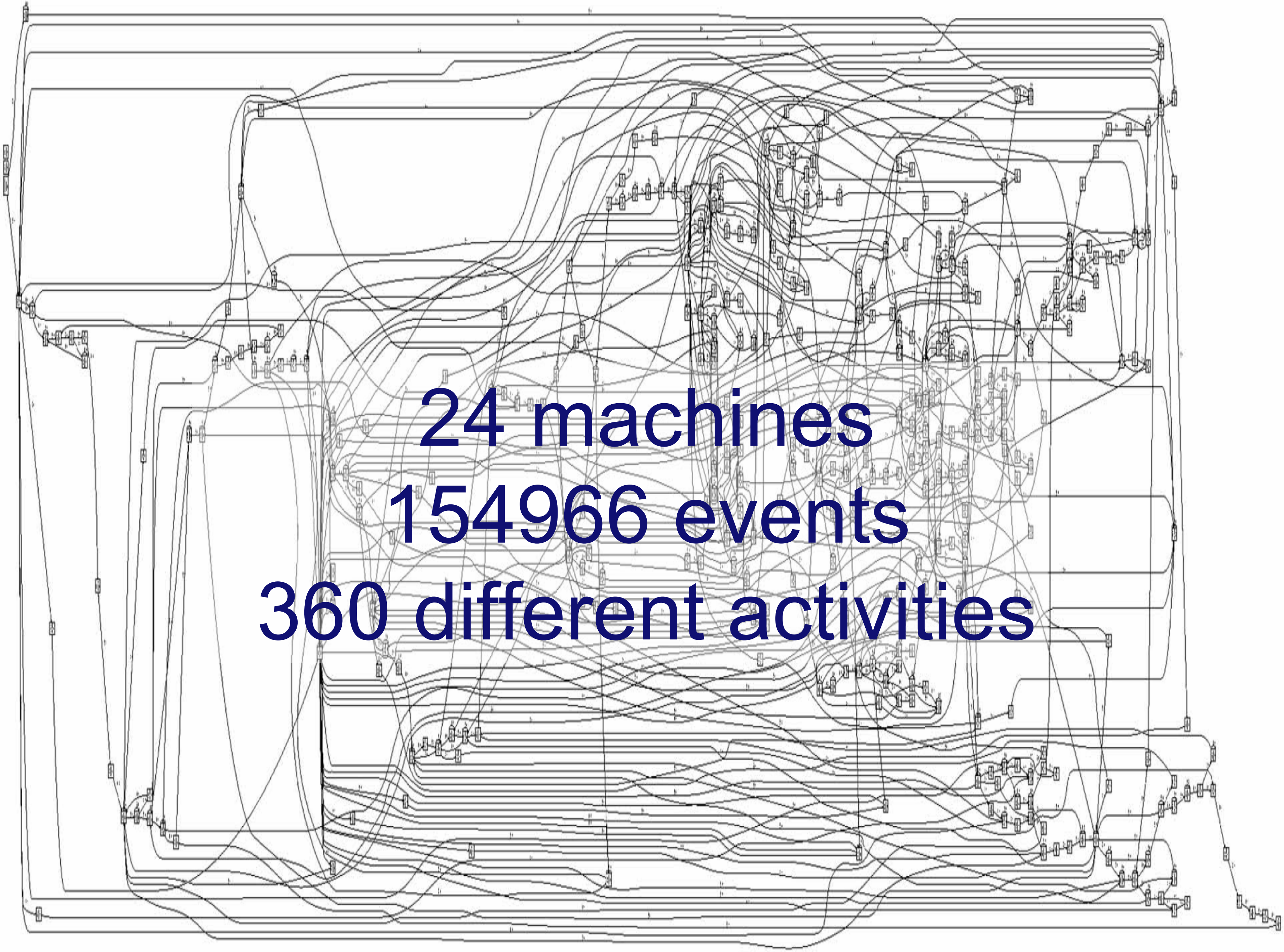

2712 patients
29258 events
264 different activities



874 patients

10478 events

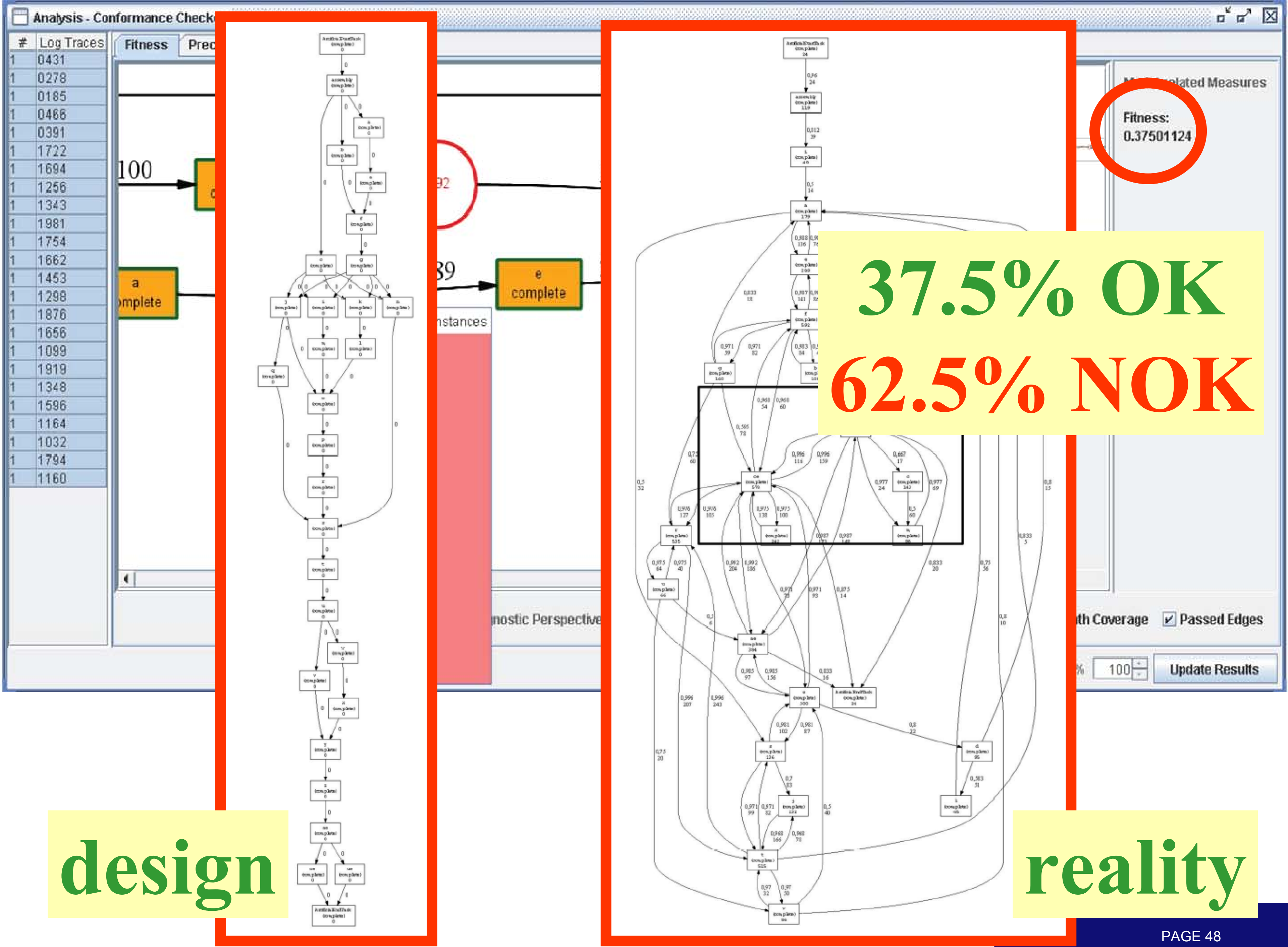
181 different activities



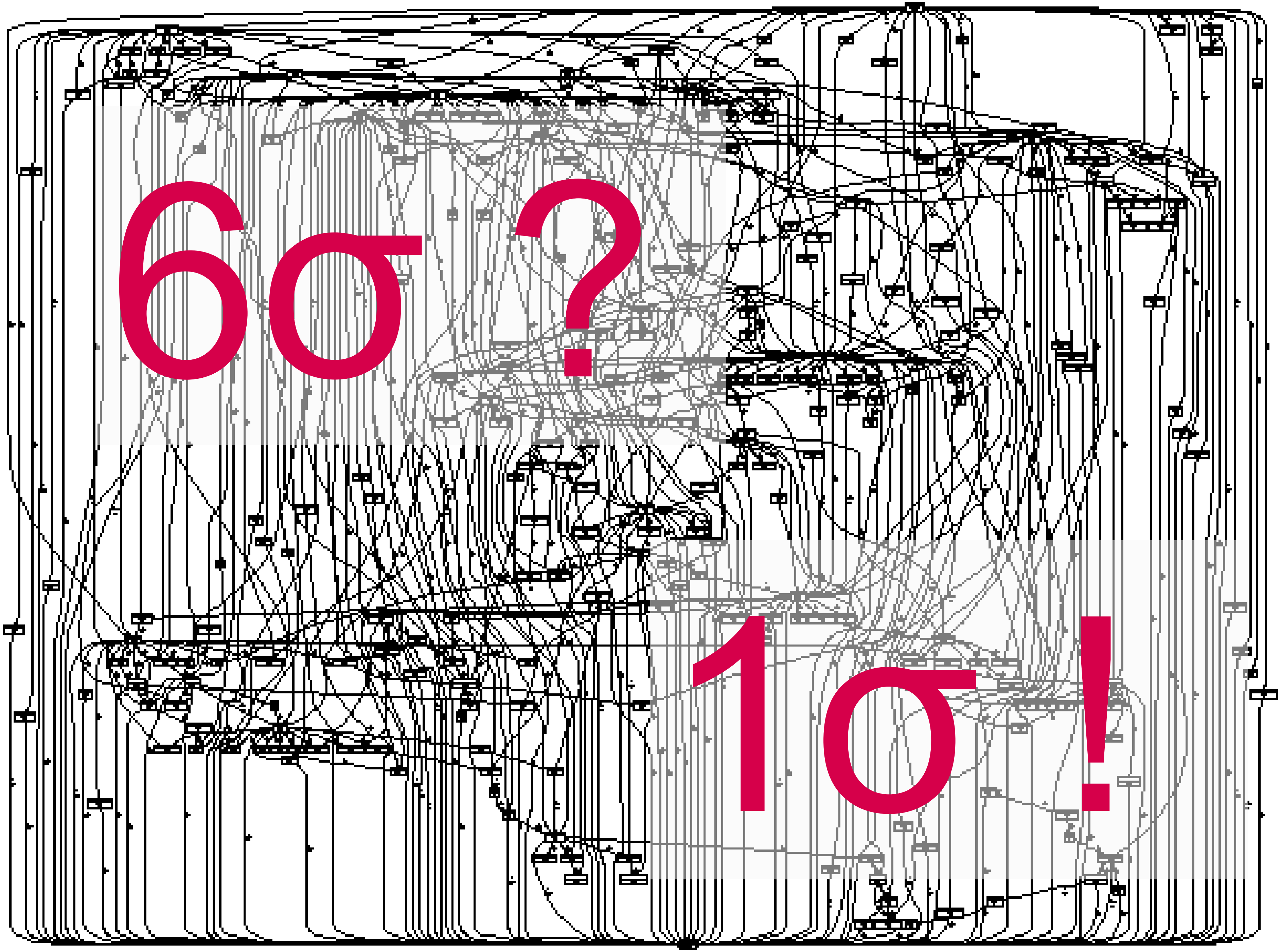
24 machines

154966 events

360 different activities



Analysis - Conformance Checker
Fitness
Fitness:
0.37501124
37.5% OK
62.5% NOK
design
reality
Passed Edges
Update Results


6σ ?
1σ !

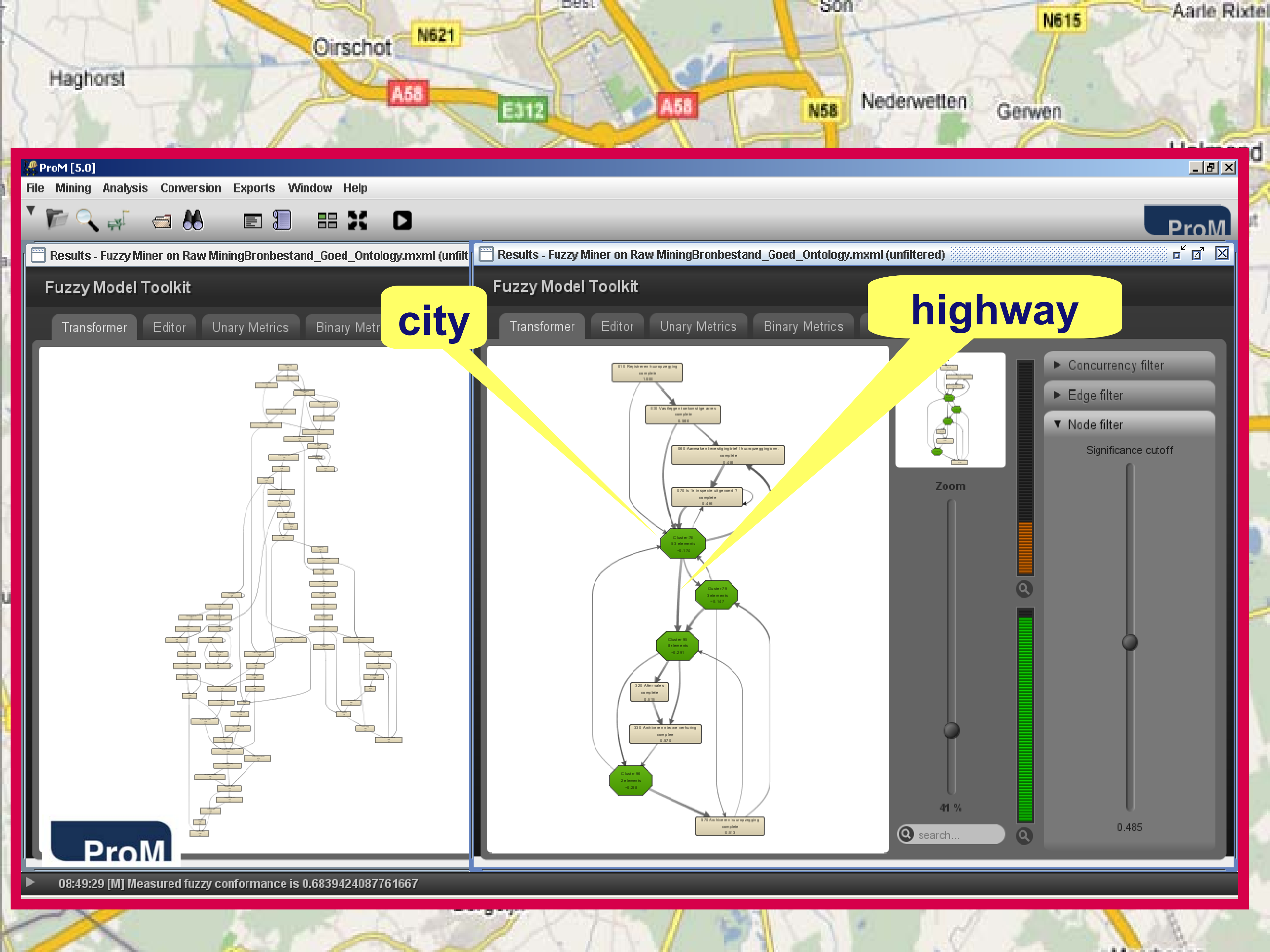
Process Mining: TomTom for Business Processes

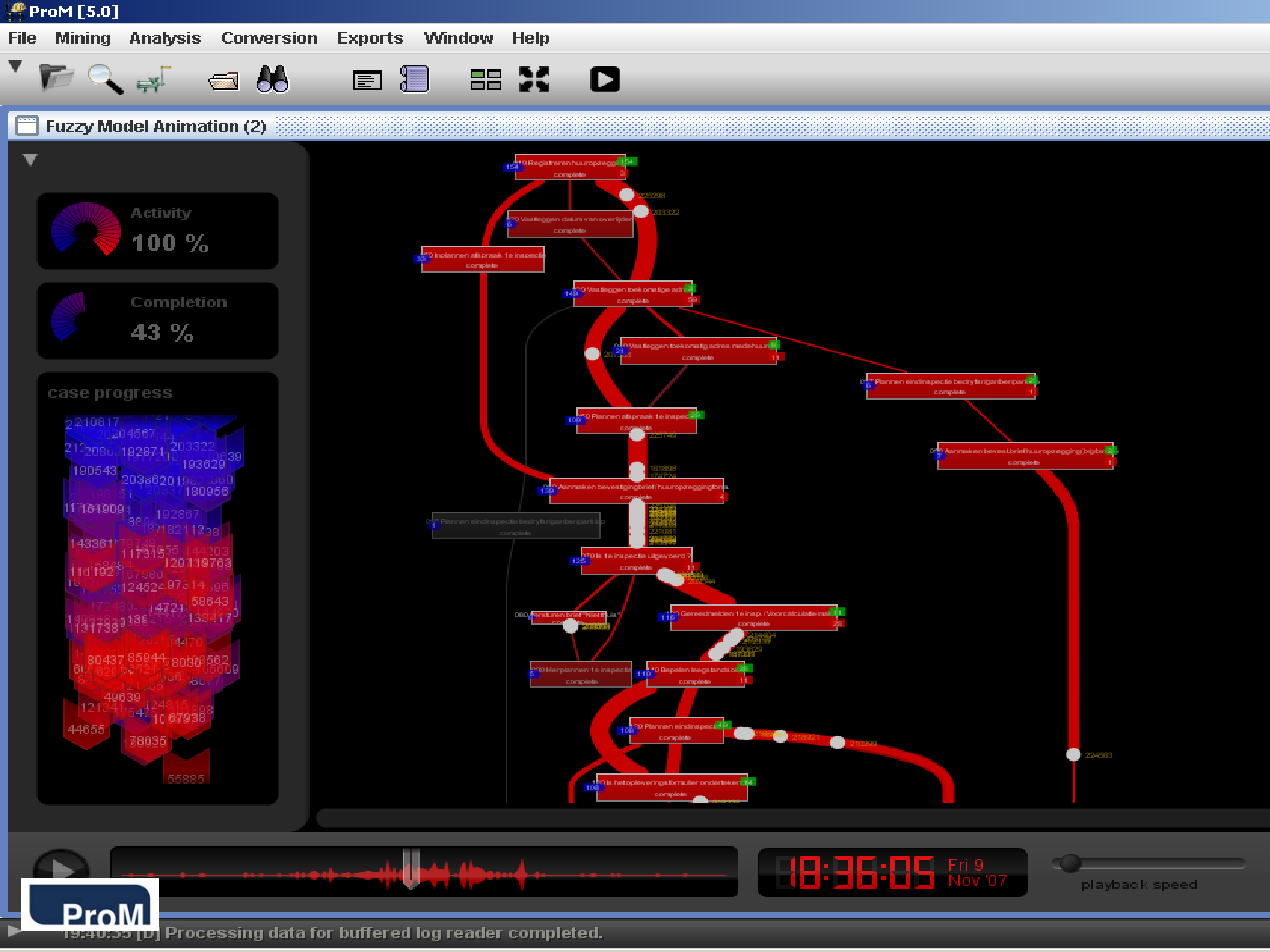
Technische Universiteit
Eindhoven
University of Technology

Where innovation starts

How can process mining help?

- Good maps?
- Navigation by PowerPoints?
- Traffic information?
- Where is the next fuel station?
- Who is in charge?
- Seamless zoom?
- Customizable views?
- When will the destination be reached?


Oirschot
Haghorst
N621
A58
E312
A58
N58
Nederwetten
Gerwen
N615
Son
ProM [5.0]
File Mining Analysis Conversion Exports Window Help
Results - Fuzzy Miner on Raw MiningBronbestand_Goed_Ontology.mxml (unfiltered)
Fuzzy Model Toolkit
Transformer Editor Unary Metrics Binary Metrics
city
highway
Concurrency filter
Edge filter
Node filter
Significance cutoff
Zoom
41 %
0.485
search...
ProM
08:49:29 [M] Measured fuzzy conformance is 0.6839424087761667


ProM [5.0]
File Mining Analysis Conversion Exports Window Help
Fuzzy Model Animation (2)
Activity
100 %
Completion
43 %
case progress
18:36:05 Fri 9 Nov '07
playback speed
ProM
19:40:35 [D] Processing data for buffered log reader completed.

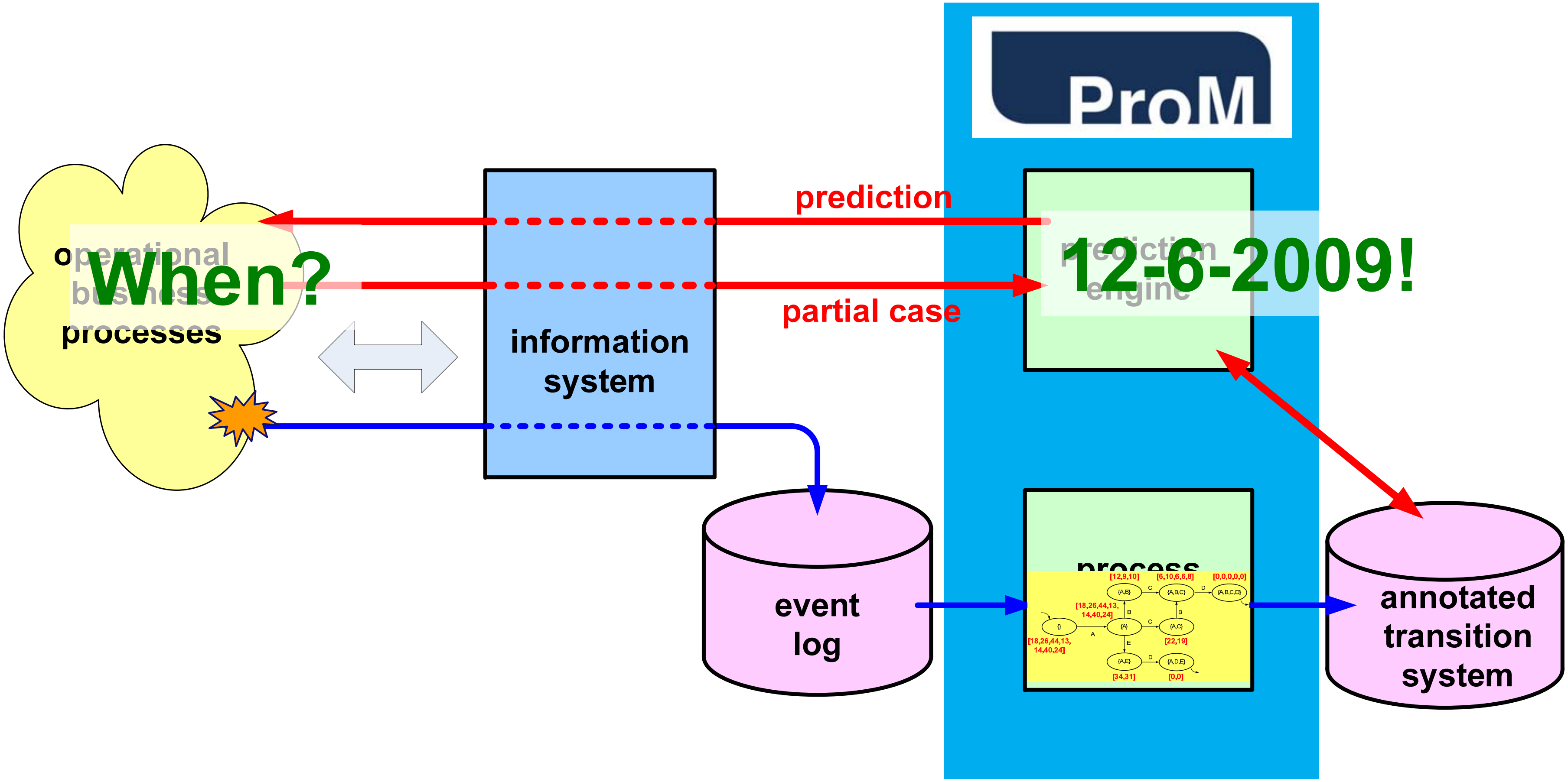
When will I be home?

Technische Universiteit
Eindhoven
University of Technology

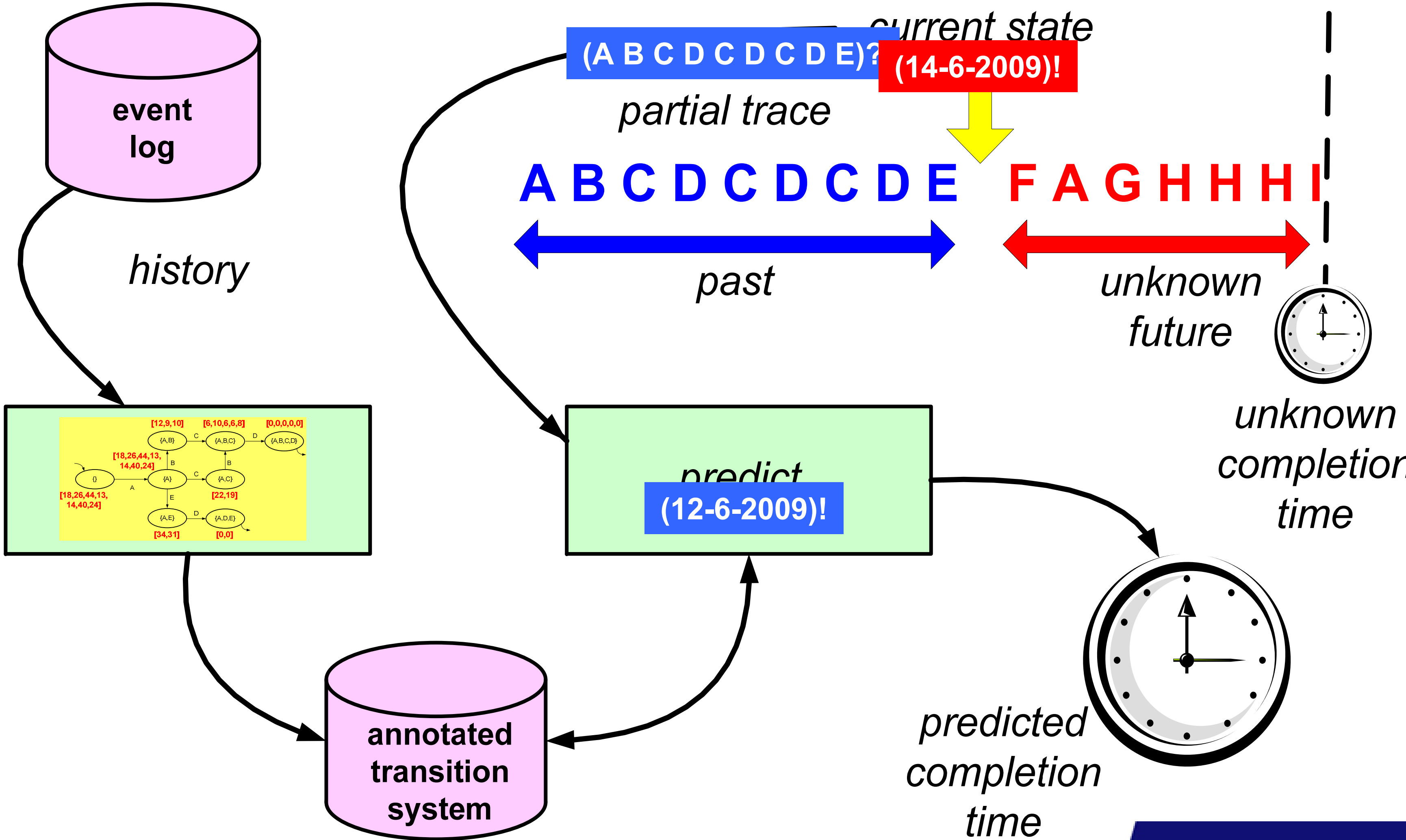
Where innovation starts


A9
Amsterdam
600
meter
11:02
11:26
100 km/h
TOMTOM

Approach


ProM
operational business processes
When?
information system
prediction
partial case
prediction engine
12-6-2009!
event log
annotated transition system

Input: partial trace and historic information

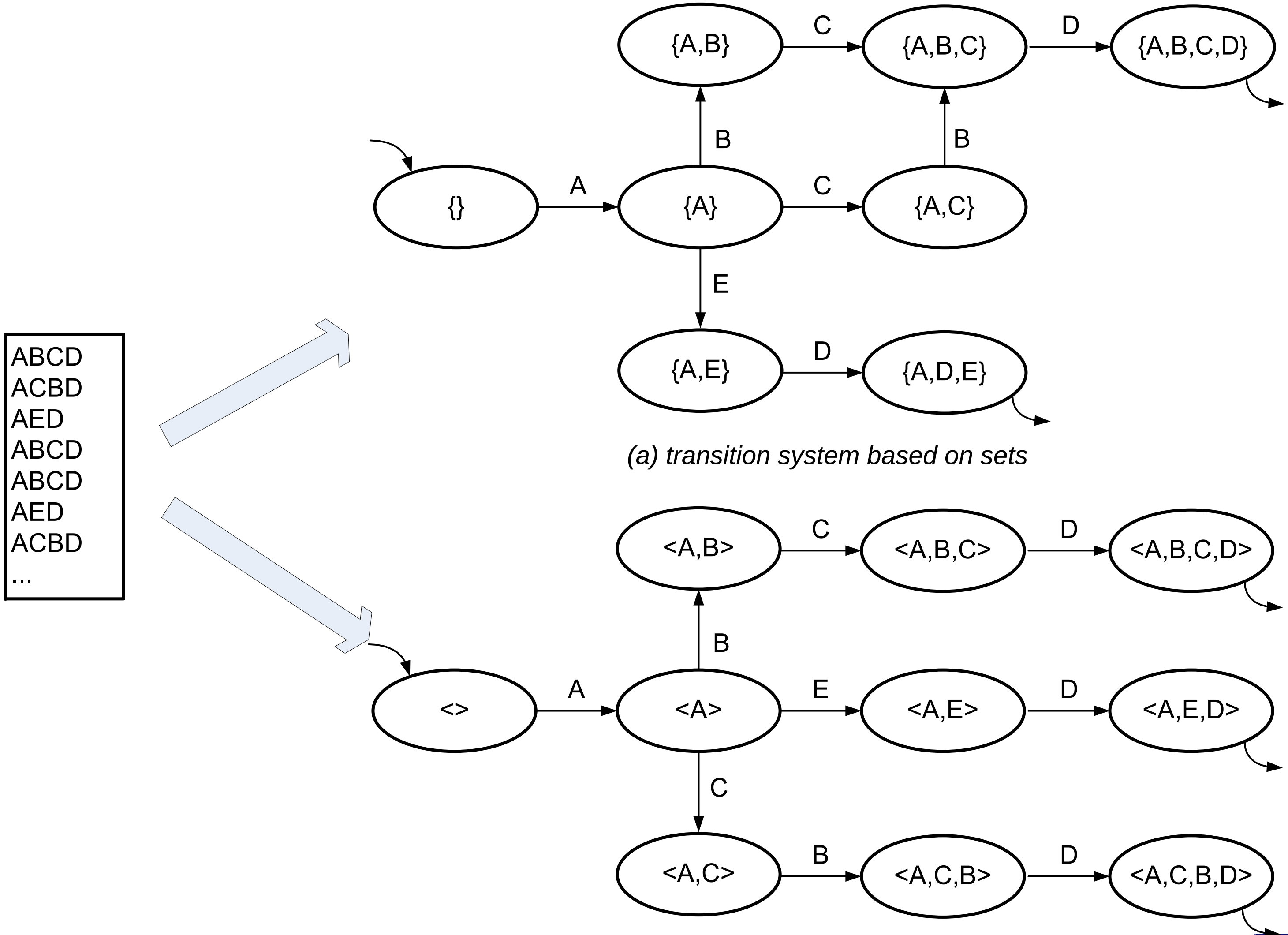


Input

event id	properties				
	timestamp	activity	resource	cost	...
35654423	30-12-2008:11.10	A	John	300	...
35654424	30-12-2008:15.21	B	John	400	...
35654425	30-12-2008:15.35	C	John	100	...
35654426	30-12-2008:15.55	D	John	400	...
35655526	29-12-2008:16.15	A	Ann	300	...
35655527	30-12-2008:16.05	C	[illegible]	[illegible]	
35655528	30-12-2008:16.25	B			
35655529	31-12-2008:10.55	D			
...	...	...			

1	$\langle A^{00}, B^{06}, C^{12}, D^{18} \rangle$
2	$\langle A^{10}, C^{14}, B^{26}, D^{36} \rangle$
3	$\langle A^{12}, E^{22}, D^{56} \rangle$
4	$\langle A^{15}, B^{19}, C^{22}, D^{28} \rangle$
5	$\langle A^{18}, B^{22}, C^{26}, D^{32} \rangle$
6	$\langle A^{19}, E^{28}, D^{59} \rangle$
7	$\langle A^{20}, C^{25}, B^{36}, D^{44} \rangle$

Building transition systems



(a) transition system based on sets

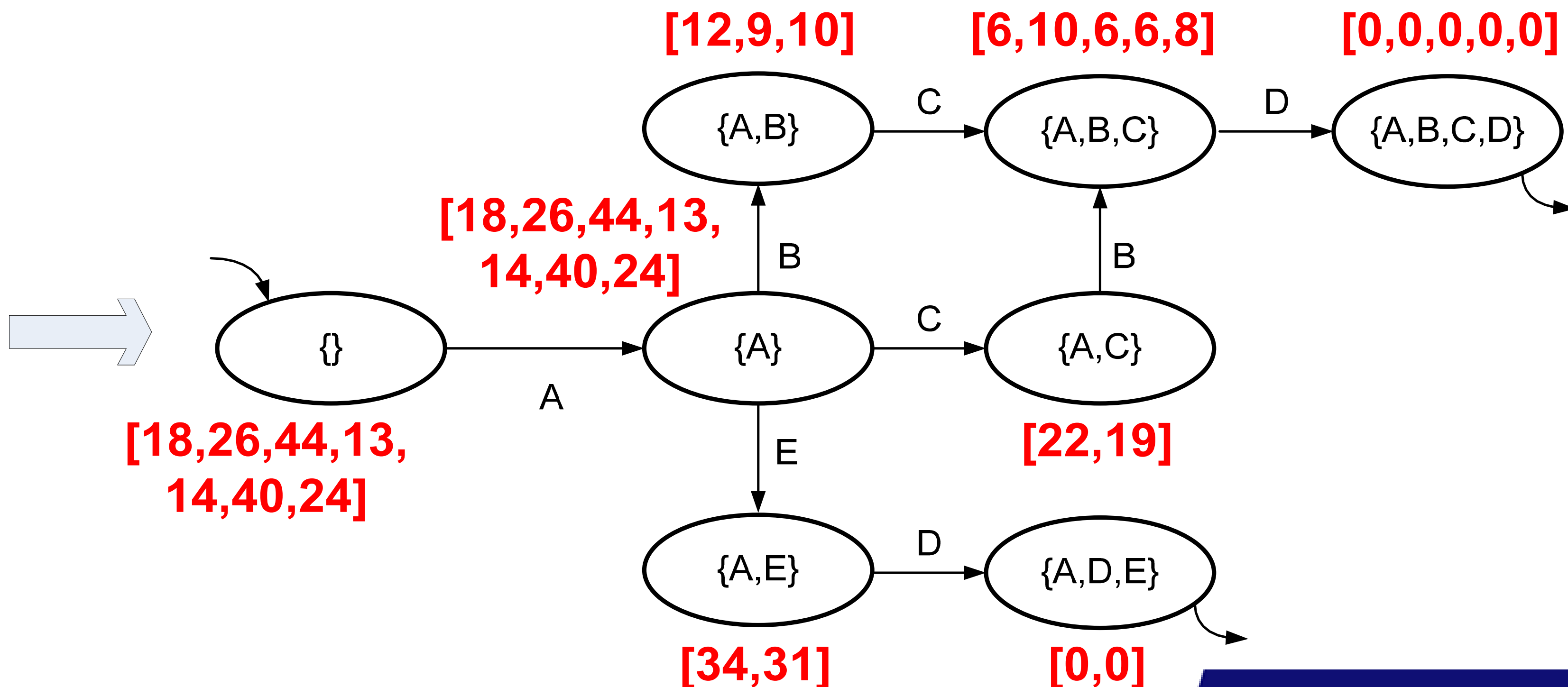
(b) transition system based on sequences

many abstractions are possible and supported by ProM's FSM miner

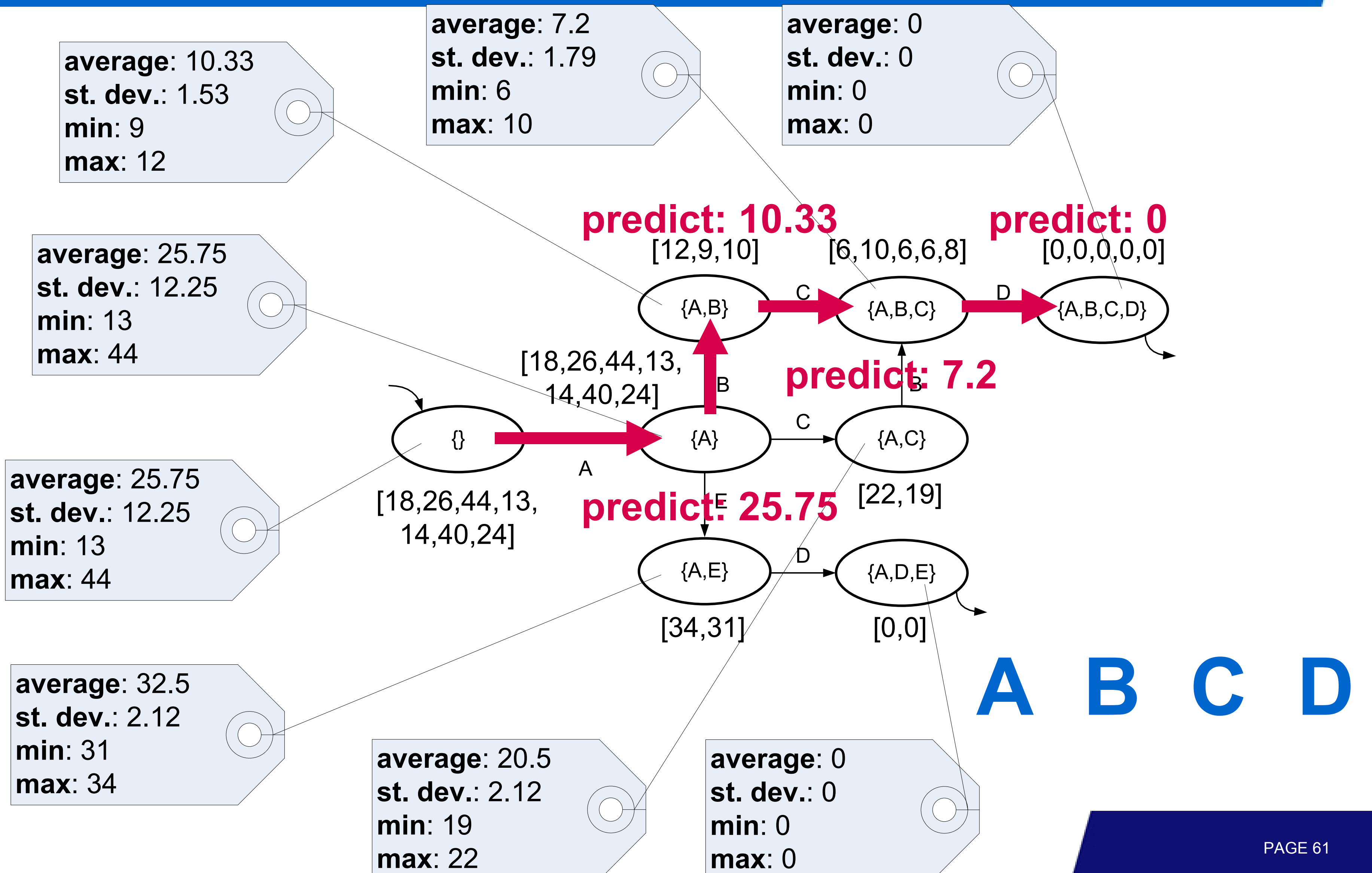
Annotated transition system based on remaining time

1	$\langle A^{00}, B^{06}, C^{12}, D^{18} \rangle$
2	$\langle A^{10}, C^{14}, B^{26}, D^{36} \rangle$
3	$\langle A^{12}, E^{22}, D^{56} \rangle$
4	$\langle A^{15}, B^{19}, C^{22}, D^{28} \rangle$
5	$\langle A^{18}, B^{22}, C^{26}, D^{32} \rangle$
6	$\langle A^{19}, E^{28}, D^{59} \rangle$
7	$\langle A^{20}, C^{25}, B^{36}, D^{44} \rangle$

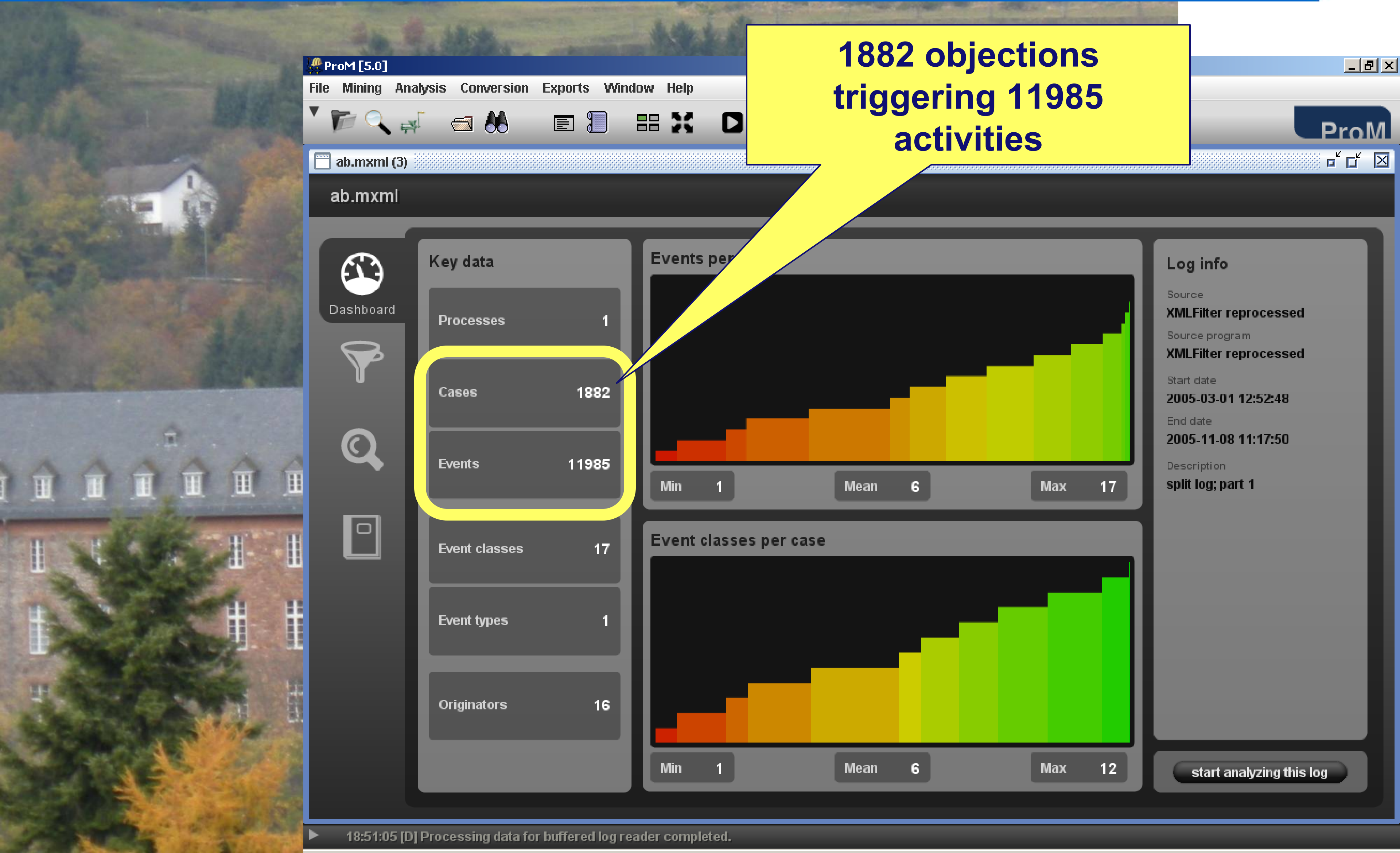
ABCD
ACBD
AED
ABCD
ABCD
AED
ACBD



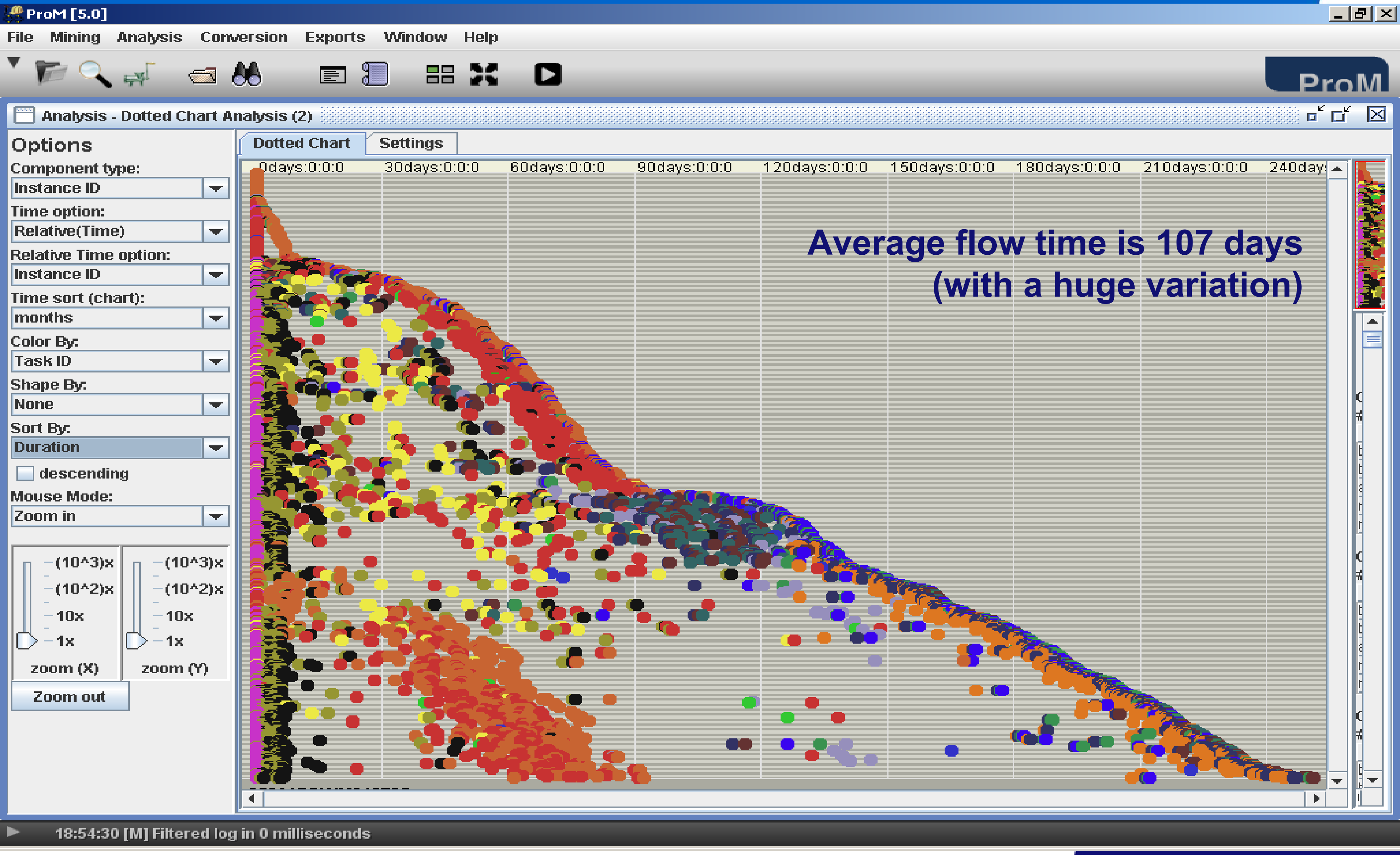
Predictive information

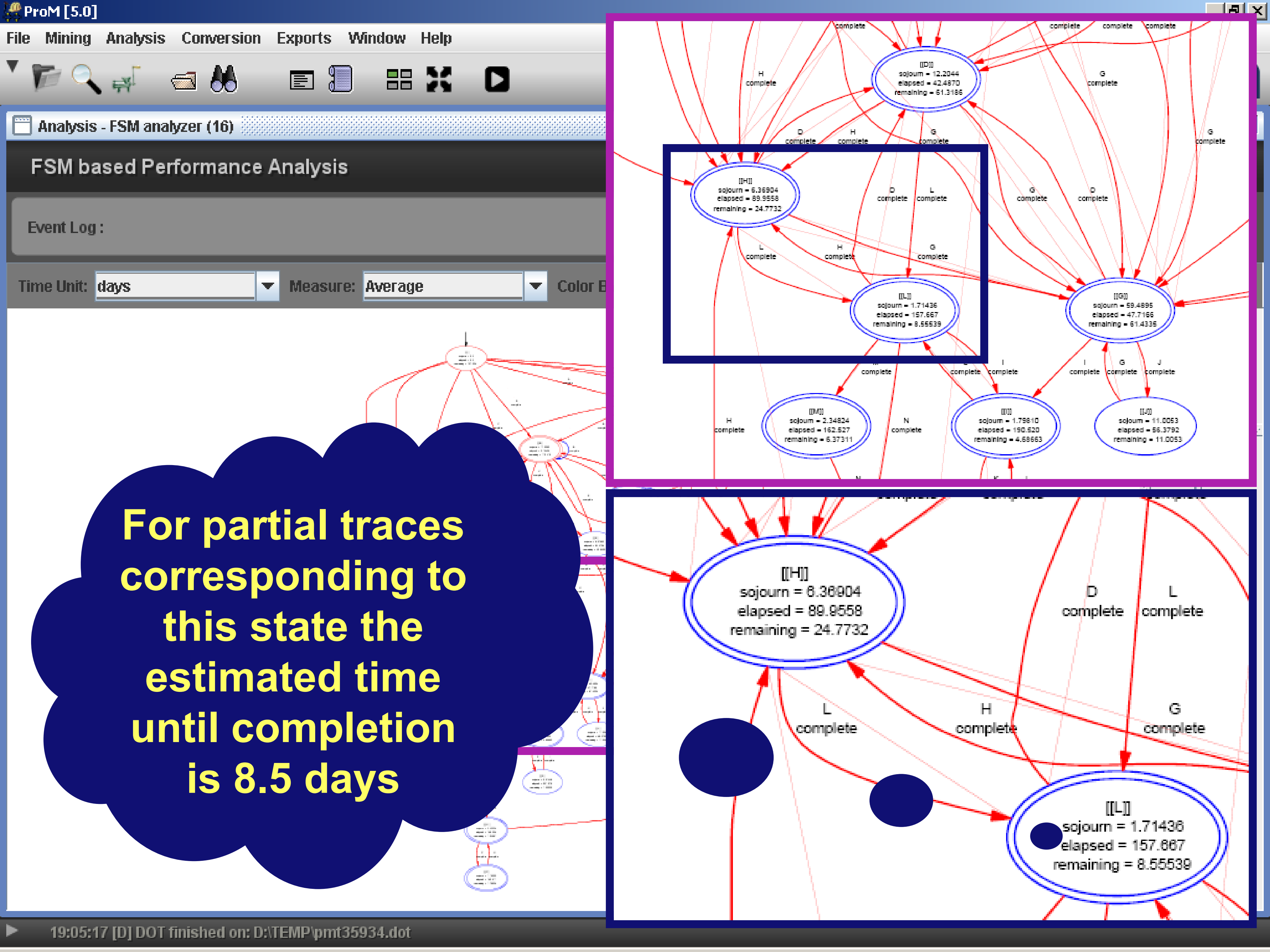


Example: WOZ process in Dutch Municipality



All 11985 events at a glance




ProM [5.0]
File Mining Analysis Conversion Exports Window Help
Analysis - FSM analyzer (16)
FSM based Performance Analysis
Event Log :
Time Unit: days
Measure: Average
For partial traces corresponding to this state the estimated time until completion is 8.5 days
[[H]]
sojourn = 6.36904
elapsed = 89.9558
remaining = 24.7732
[[L]]
sojourn = 1.71436
elapsed = 157.667
remaining = 8.55539
D complete
L complete
H complete
G complete
19:05:17 [D] DOT finished on: D:\TEMP\pmt35934.dot

Cross validation: training set and test set

rts Window Help

ProM

Mean Average Error (MAE)

rooted MSE

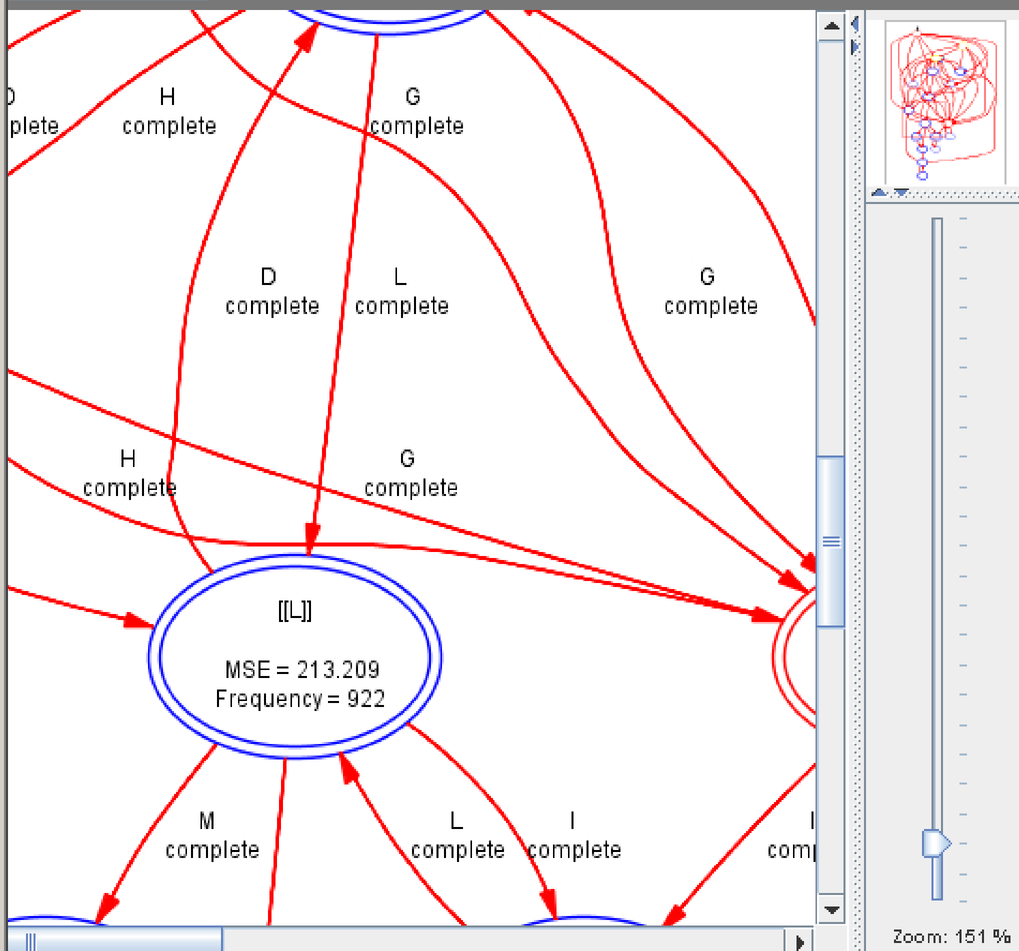
MAPE

graph view text view

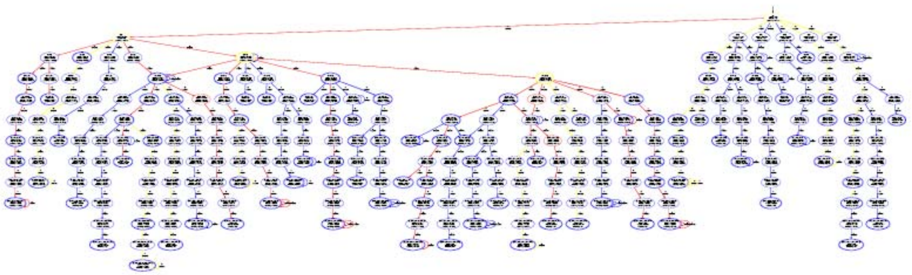
[[B]]	[illegible]	[illegible]	[illegible]	1488
[[C]]	64.8881	72.8452	406.781	896
[[D]]	63.1935	69.4784	51727.3	1240
[[G]]	72.8231	74.6510	24.3339	1546
[[I]]	3.25089	5.59291	103.264	629
[[L]]	9.82483	14.6016	406.304	922
[[M]]	8.89983	13.6616	1331.10	866
[[N]]	7.74203	12.2609	1478.89	799
[[O]]	3.17161	9.07740	1351.93	932
[[P]]	2.98513	8.61544	554.336	709
[[F]]	34.0397	44.4393	52294.3	16
[[H]]	16.2393	22.9282	447.301	326
[[E]]	46.4377	55.1125	1232.75	27
[[Q]]	36.6227	47.2489	133.769	238
[[J]]	2.67055	2.67055	25.7842	2
[[K]]	0.0	0.0	0.0	1
overall(mean)	31.3298	37.1198	11359.0	18
overall(aggregated	[illegible]	[illegible]	[illegible]	[illegible]

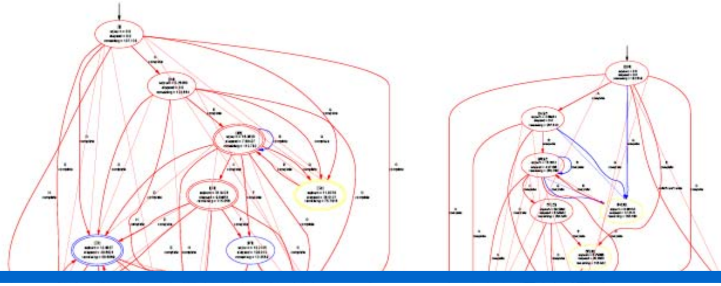
pmt35935.dot

min 60 max 80 Update



Some results


(a) Set abstraction based on all activities



Abstraction	MAE	RMSE	MAPE
Set abstraction based on all activities (cf. Figure 17(a))	41.648	47.513	1505.07
Set abstraction based on last activity (cf. Figure 17(b))	43.080	49.666	1818.49
Set abstraction based on last activity and additional information related to the occurrence of "I" (cf. Figure 17(c))	17.129	23.550	900.07
Complete abstraction (cf. Figure 17(d))	63.391	74.965	7169.55
Simple heuristic: half of average total flow time (53.57 days)	61.750	75.505	6188.04


Hardenbergstraße
65
17.7
11:01
0:25
11:26
Marchstraße
3 min
TOMTOM

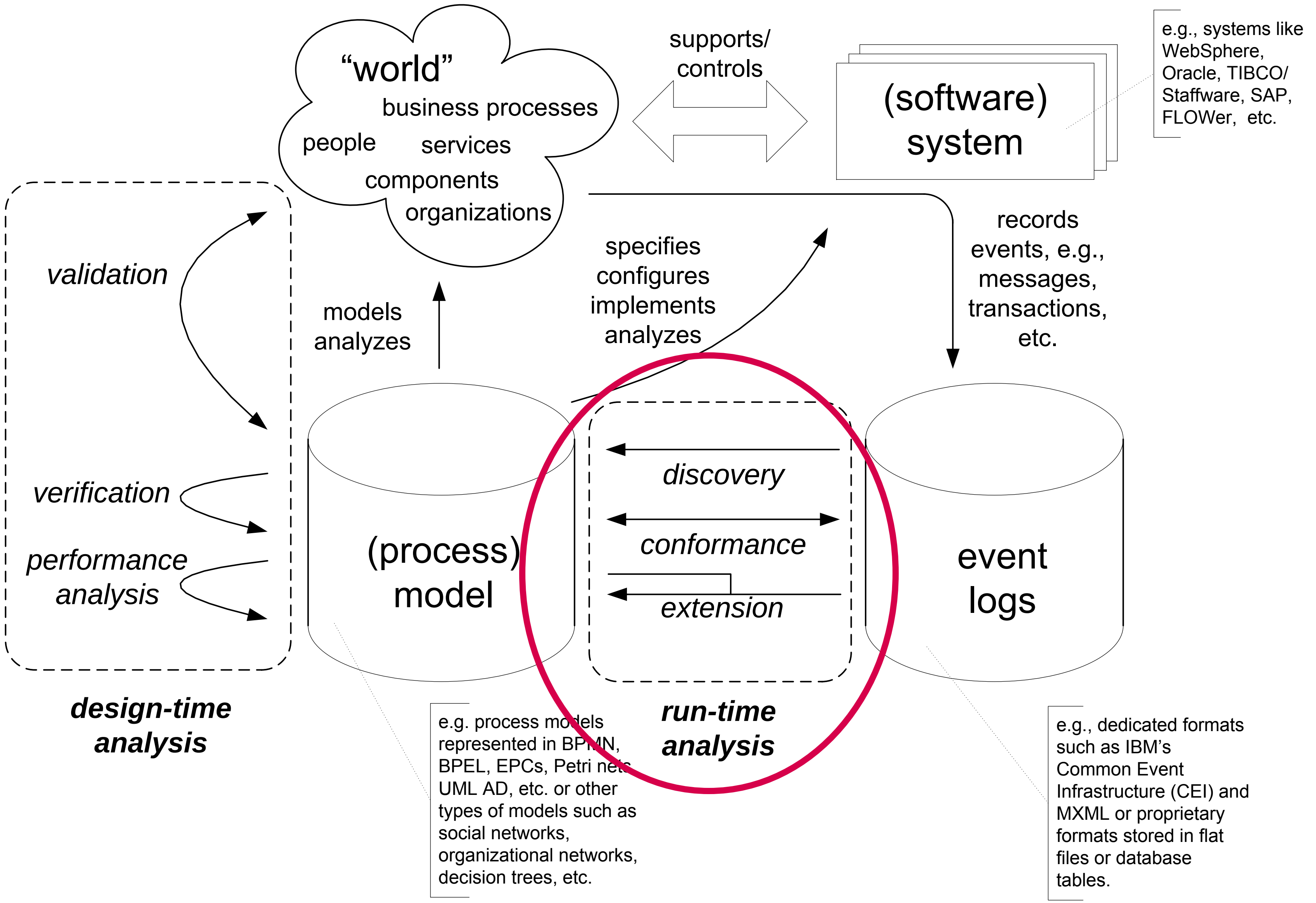
Part II

Process Discovery – The Alpha Algorithm

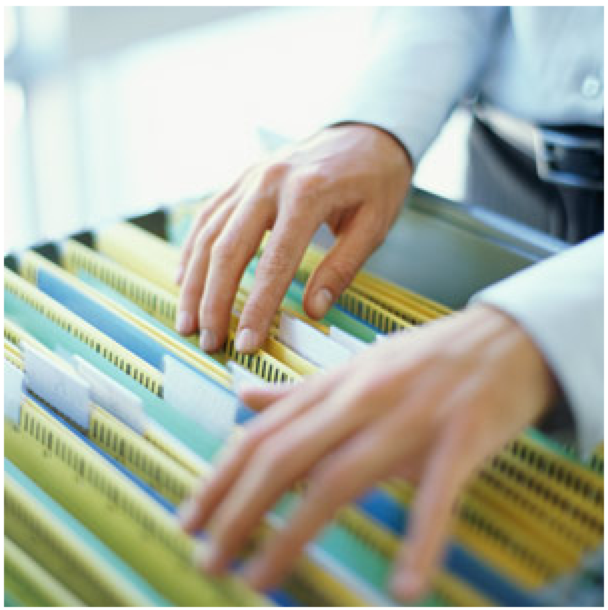
www.processmining.org

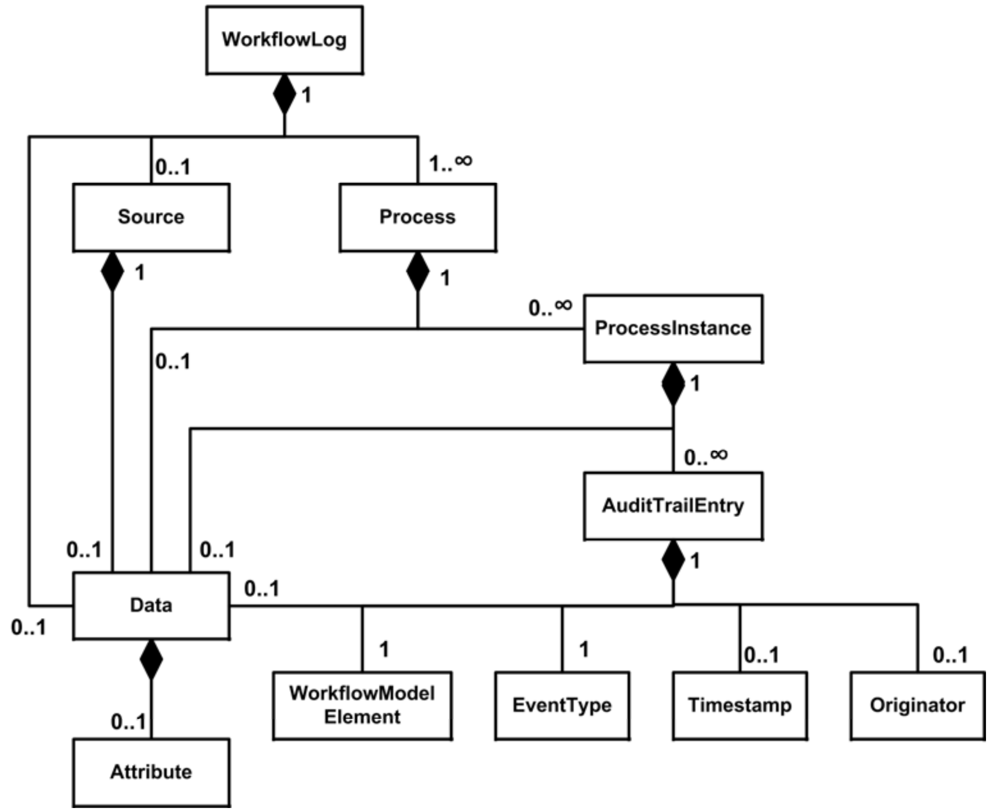
Where innovation starts

Design-time analysis vs run-time analysis



Starting point: event logs





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

unified event log (MXML)

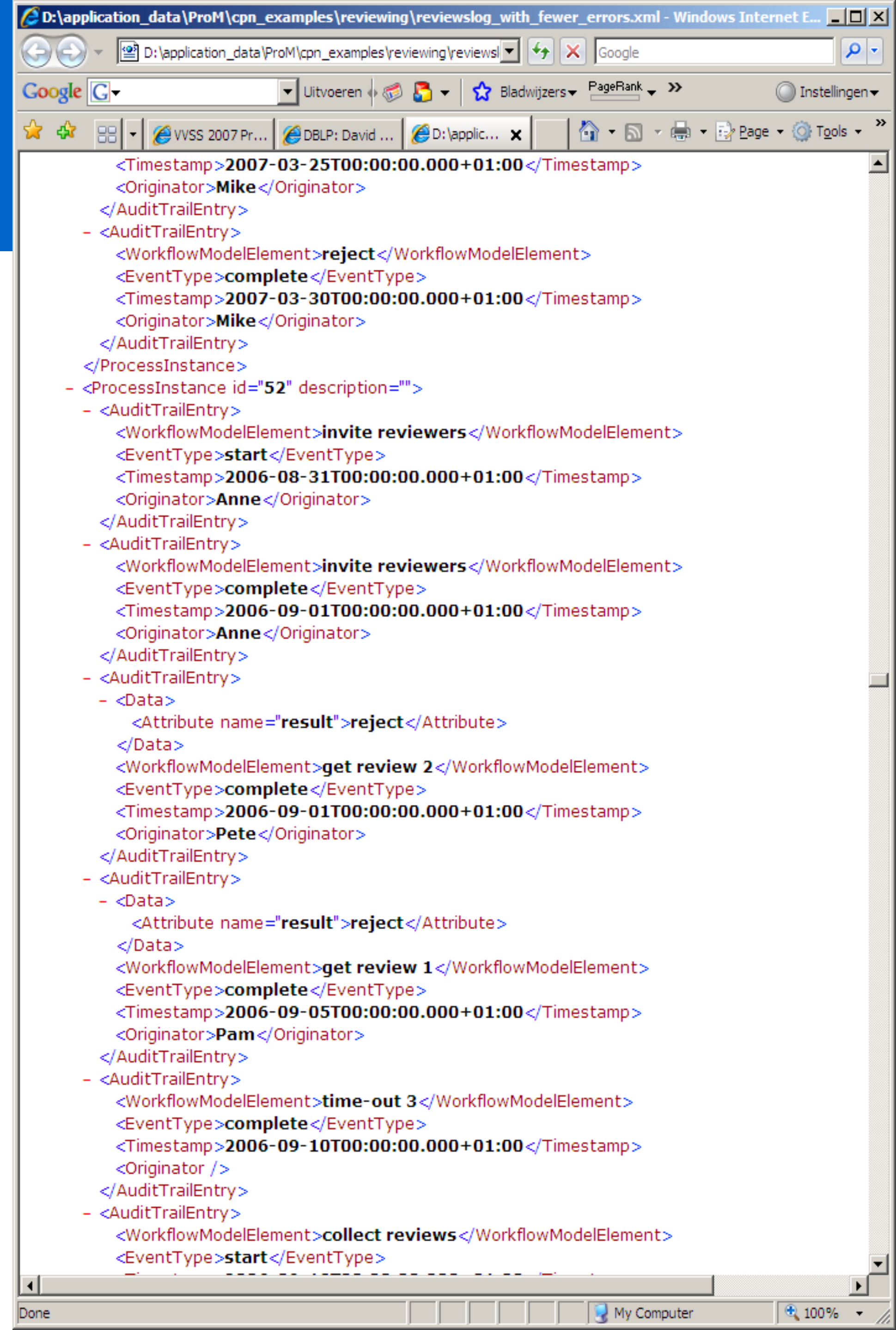
MXML

Event log:

- processes
  - process instances
    - events

Per event:

- activity name
- (event type)
- (originator)
- (timestamp)
- (data)




```
<Timestamp>2007-03-25T00:00:00.000+01:00</Timestamp>
<Originator>Mike</Originator>
</AuditTrailEntry>
- <AuditTrailEntry>
<WorkflowModelElement>reject</WorkflowModelElement>
<EventType>complete</EventType>
<Timestamp>2007-03-30T00:00:00.000+01:00</Timestamp>
<Originator>Mike</Originator>
</AuditTrailEntry>
</ProcessInstance>
- <ProcessInstance id="52" description="">
- <AuditTrailEntry>
<WorkflowModelElement>invite reviewers</WorkflowModelElement>
<EventType>start</EventType>
<Timestamp>2006-08-31T00:00:00.000+01:00</Timestamp>
<Originator>Anne</Originator>
</AuditTrailEntry>
- <AuditTrailEntry>
<WorkflowModelElement>invite reviewers</WorkflowModelElement>
<EventType>complete</EventType>
<Timestamp>2006-09-01T00:00:00.000+01:00</Timestamp>
<Originator>Anne</Originator>
</AuditTrailEntry>
- <AuditTrailEntry>
- <Data>
<Attribute name="result">reject</Attribute>
</Data>
<WorkflowModelElement>get review 2</WorkflowModelElement>
<EventType>complete</EventType>
<Timestamp>2006-09-01T00:00:00.000+01:00</Timestamp>
<Originator>Pete</Originator>
</AuditTrailEntry>
- <AuditTrailEntry>
- <Data>
<Attribute name="result">reject</Attribute>
</Data>
<WorkflowModelElement>get review 1</WorkflowModelElement>
<EventType>complete</EventType>
<Timestamp>2006-09-05T00:00:00.000+01:00</Timestamp>
<Originator>Pam</Originator>
</AuditTrailEntry>
- <AuditTrailEntry>
<WorkflowModelElement>time-out 3</WorkflowModelElement>
<EventType>complete</EventType>
<Timestamp>2006-09-10T00:00:00.000+01:00</Timestamp>
<Originator />
</AuditTrailEntry>
- <AuditTrailEntry>
<WorkflowModelElement>collect reviews</WorkflowModelElement>
<EventType>start</EventType>
```

```
</ProcessInstance>
- <ProcessInstance id="51" description="">
- <AuditTrailEntry>
    <WorkflowModelElement>invite reviewers</WorkflowModelElem
    <EventType>start</EventType>
    <Timestamp>2006-08-28T00:00:00.000+01:00</Timestamp>
    <Originator>Mike</Originator>
  </AuditTrailEntry>
- <AuditTrailEntry>
    <WorkflowModelElement>invite reviewers</WorkflowModelEle
    <EventType>complete</EventType>
    <Timestamp>2006-08-31T00:00:00.000+01:00</Timestamp
    <Originator>Mike</Originator>
  </AuditTrailEntry>
- <AuditTrailEntry>
  - <Data>
      <Attribute name="result">reject</Attribute>
    </Data>
    <WorkflowModelElement>get review 3</WorkflowModelEleme
    <EventType>complete</EventType>
    <Timestamp>2006-09-02T00:00:00.000+01:00</Timestam
    <Originator>Mary</Originator>
  </AuditTrailEntry>
- <AuditTrailEntry>
    <WorkflowModelElement>time-out 1</WorkflowModelElement>
    <EventType>complete</EventType>
    <Timestamp>2006-09-03T00:00:00.000+01:00</Timestamp>
    <Originator />
  </AuditTrailEntry>
- <AuditTrailEntry>
```

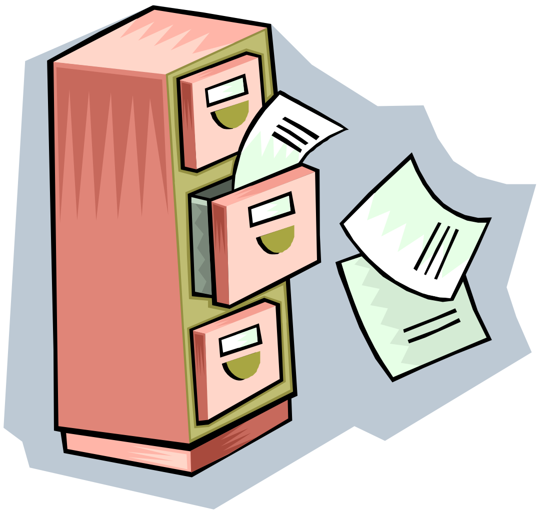

attributes of an event

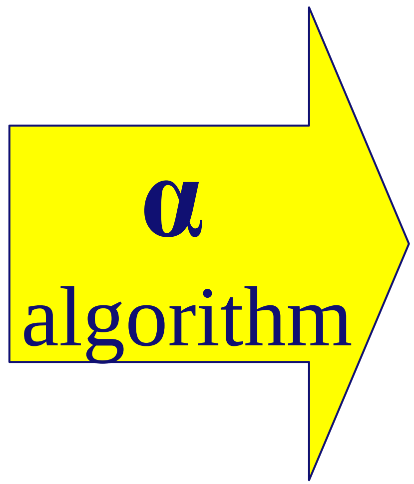
end of activity

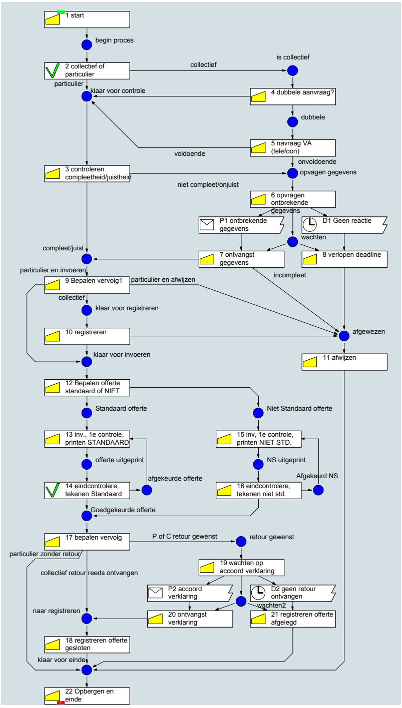
activity

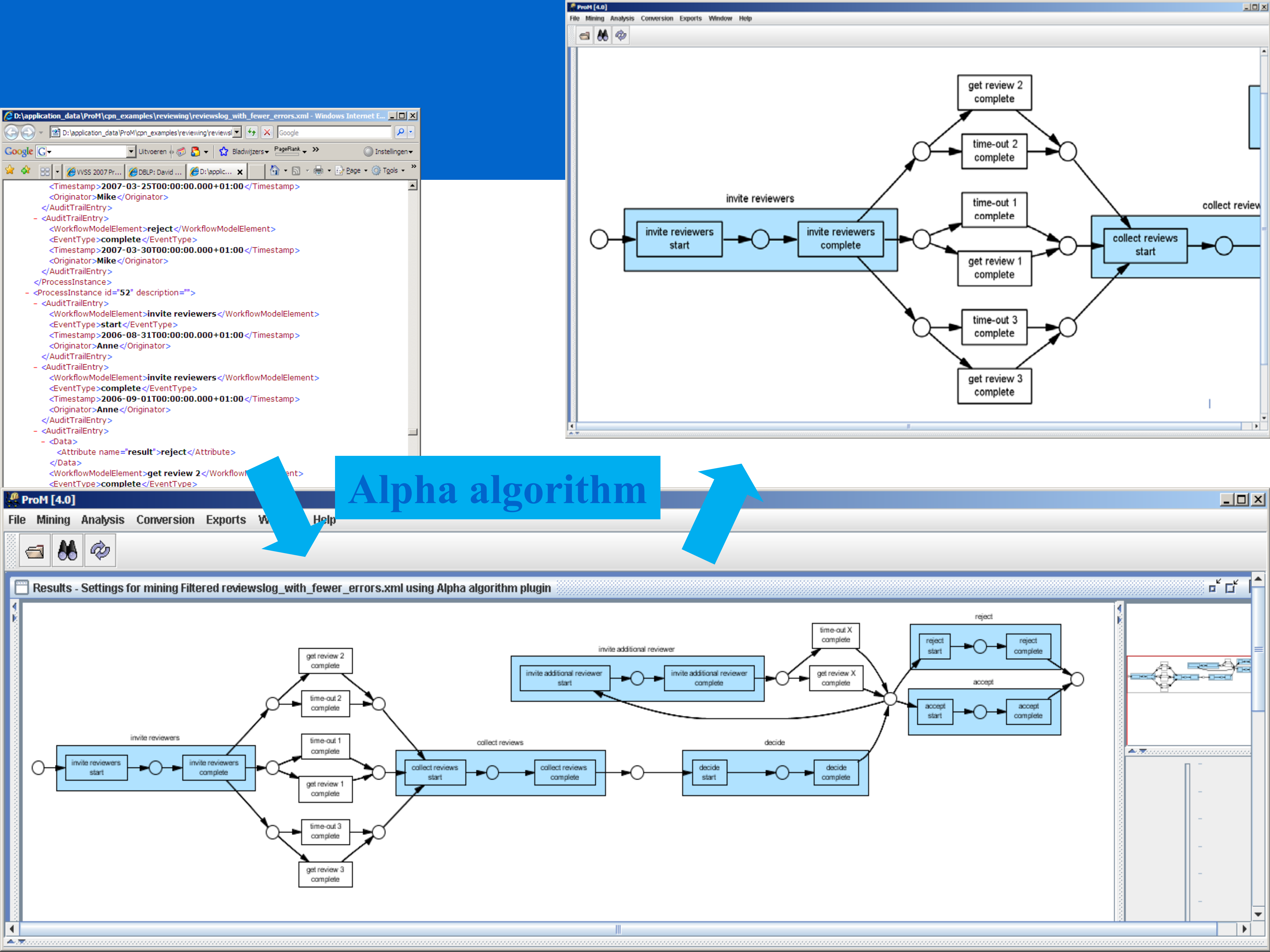
instance

Process Mining: The alpha algorithm



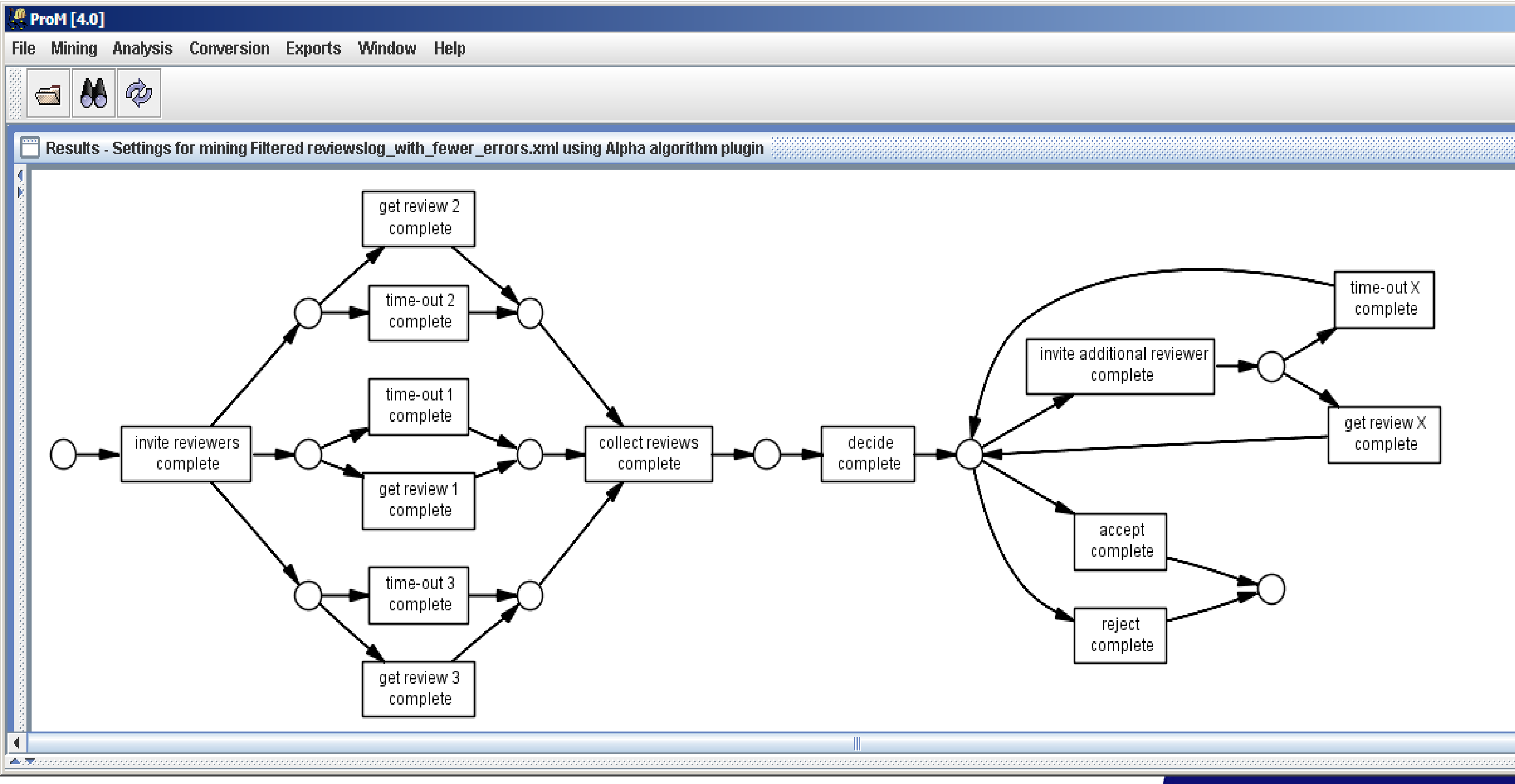






D:\application_data\ProM\cpn_examples\reviewing\reviewslog_with_fewer_errors.xml - Windows Internet E...
<Timestamp>2007-03-25T00:00:00.000+01:00</Timestamp>
<Originator>Mike</Originator>
</AuditTrailEntry>
<AuditTrailEntry>
<WorkflowModelElement>reject</WorkflowModelElement>
<EventType>complete</EventType>
<Timestamp>2007-03-30T00:00:00.000+01:00</Timestamp>
<Originator>Mike</Originator>
</AuditTrailEntry>
</ProcessInstance>
<ProcessInstance id="52" description="">
<AuditTrailEntry>
<WorkflowModelElement>invite reviewers</WorkflowModelElement>
<EventType>start</EventType>
<Timestamp>2006-08-31T00:00:00.000+01:00</Timestamp>
<Originator>Anne</Originator>
</AuditTrailEntry>
<AuditTrailEntry>
<WorkflowModelElement>invite reviewers</WorkflowModelElement>
<EventType>complete</EventType>
<Timestamp>2006-09-01T00:00:00.000+01:00</Timestamp>
<Originator>Anne</Originator>
</AuditTrailEntry>
<AuditTrailEntry>
<Data>
<Attribute name="result">reject</Attribute>
</Data>
<EventType>complete</EventType>
ProM [4.0]
File Mining Analysis Conversion Exports Window Help
invite reviewers
invite reviewers start
invite reviewers complete
get review 2 complete
time-out 2 complete
time-out 1 complete
get review 1 complete
time-out 3 complete
get review 3 complete
collect reviews start
Alpha algorithm
Results - Settings for mining Filtered reviewslog_with_fewer_errors.xml using Alpha algorithm plugin
collect reviews
collect reviews complete
decide
decide start
decide complete
invite additional reviewer
invite additional reviewer start
invite additional reviewer complete
time-out X complete
get review X complete
reject
reject start
reject complete
accept
accept start
accept complete

Without transactional information (just completes)



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
```

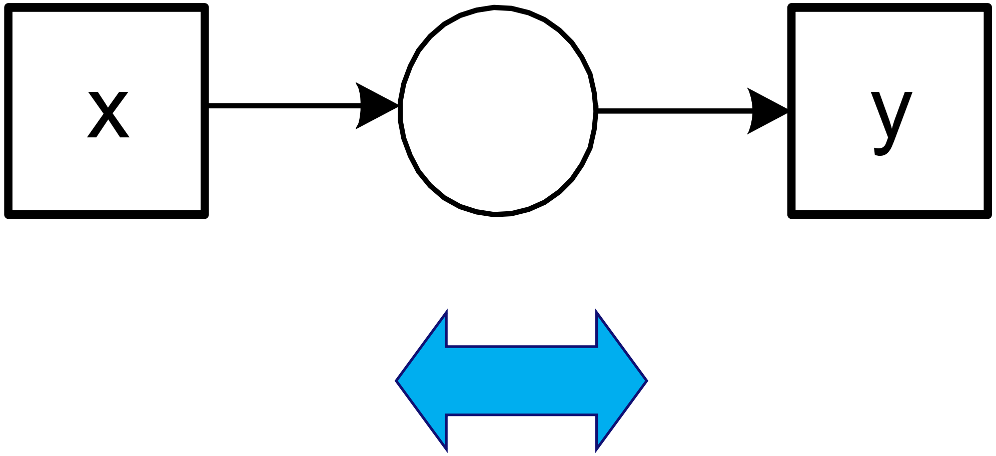
ABCD
ACBD
EF

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

A→B
A→C
B→D
C→D
E→F

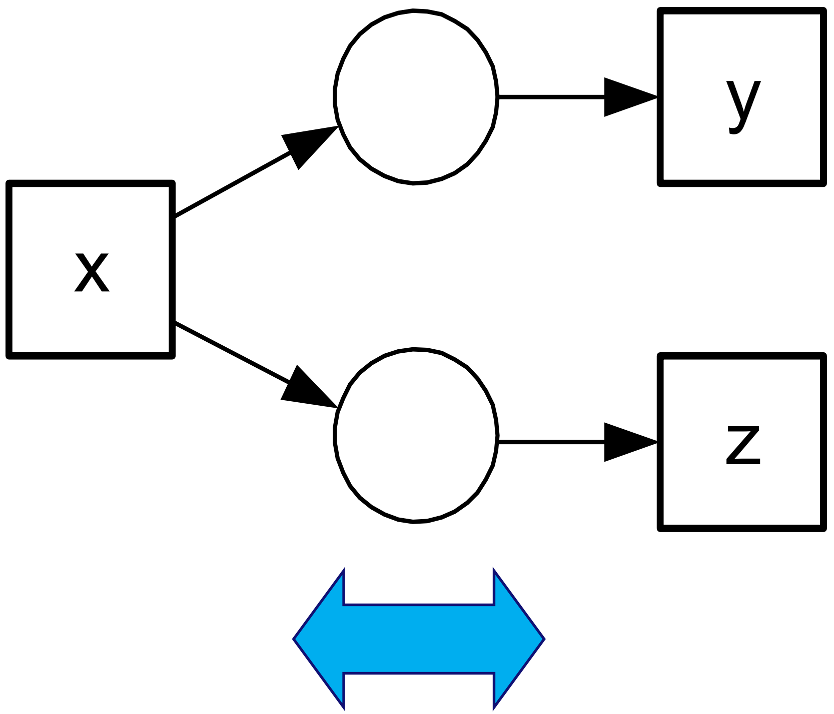
B||C
C||B

Basic idea (1)



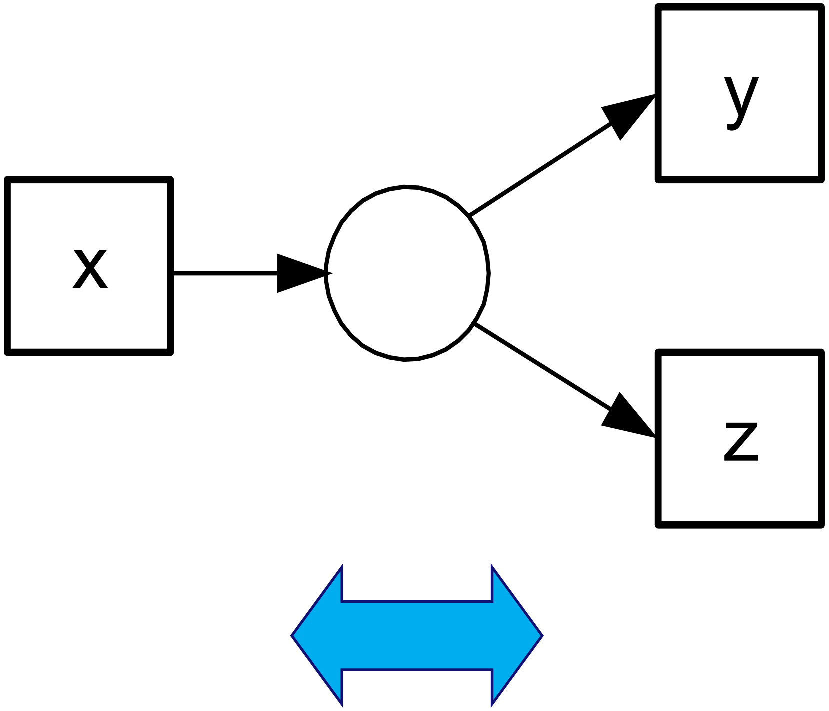
$$x \rightarrow y$$

Basic idea (2)



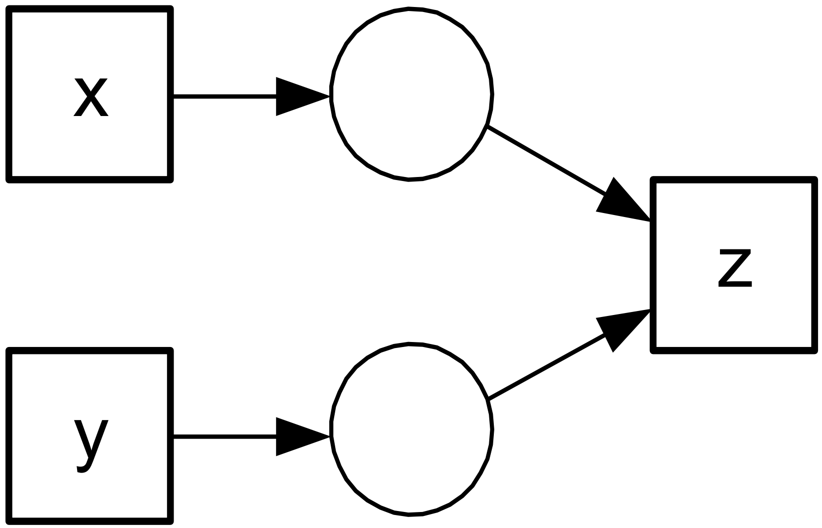
$x \to y$, $x \to z$, and $y || z$

Basic idea (3)



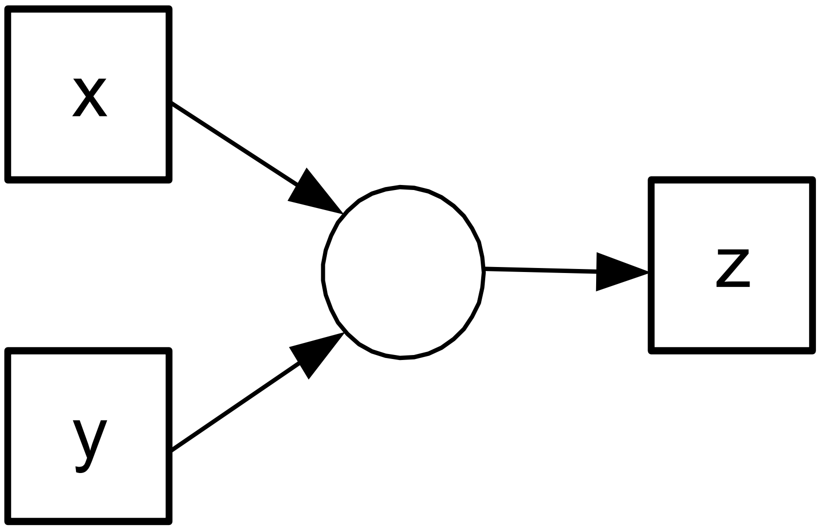
$x \to y$, $x \to z$, and $y \# z$

Basic idea (4)



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

Basic idea (5)



$x \rightarrow z$, $y \rightarrow z$, 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

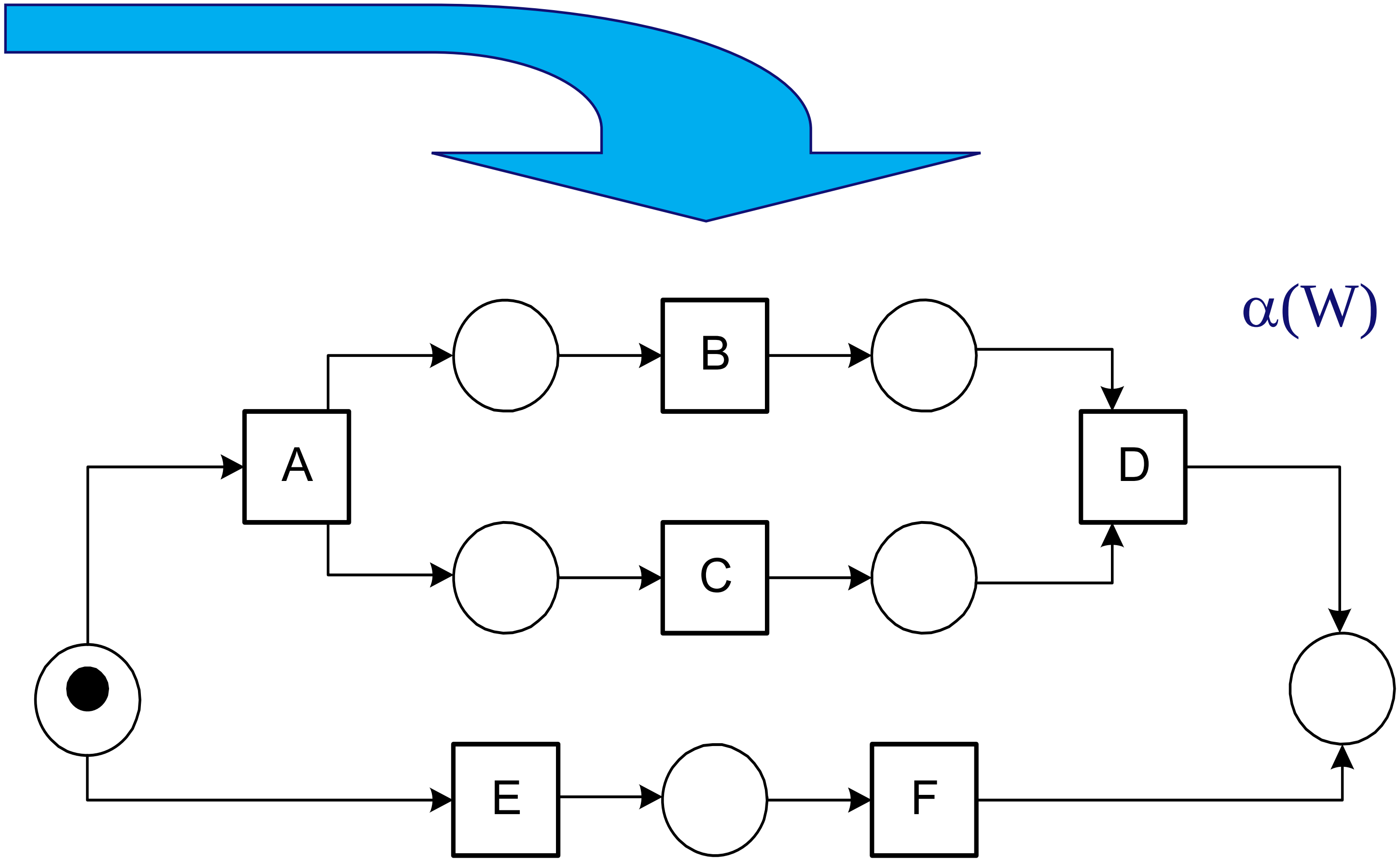
W:

```
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

$A \rightarrow B$
$A \rightarrow C$
$B \rightarrow D$
$C \rightarrow D$
$E \rightarrow F$

B||C
C||B



Exercise (1)

- **What does the alpha algorithm produce for a log consisting only of the following traces?**
  - **ABCD**
  - **ACBD**
  - **AED**

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. $\alpha(W) = (P_W, T_W, F_W)$.

- Direct succession: **$x>y$ iff for some case x is directly followed by y.**
- Causality: **$x \rightarrow y$ 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 $y>x$.**

Another example taken step-by-step ...

Definition 3.1 (Workflow trace, Workflow log). *Let T be a set of tasks. $\sigma \in T^*$ is a* workflow trace *and $W \in \mathcal{P}(T^*)$ is a* workflow log.[2]

The workflow trace of case 1 in Table 1 is $ABCD$. The workflow log corresponding to Table 1 is

$$\{ABCD, ACBD, AED\}.$$

ase identifier	task identifier
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 A
case 4	task C
case 1	task D
case 3	task C
case 3	task D
case 4	task B
case 5	task E
case 5	task D
case 4	task D

Taken from:

Wil M. P. van der Aalst, Ton Weijters, Laura Maruster: Workflow Mining: Discovering Process Models from Event Logs. IEEE Trans. Knowl. Data Eng. 16(9): 1128-1142 (2004)

Definition 3.2 (Log-based ordering relations). *Let W be a workflow log over T, i.e., $W \in \mathcal{P}(T^*)$. Let $a, b \in T$:*

- *$a >_W b$ iff there is a trace $\sigma = t_1 t_2 t_3 \ldots t_{n-1}$ and $i \in \{1, \ldots, n-2\}$ such that $\sigma \in W$ and $t_i = a$ and $t_{i+1} = b$,*
- *$a \rightarrow_W b$ iff $a >_W b$ and $b \not>_W a$,*
- *$a \#_W b$ iff $a \not>_W b$ and $b \not>_W a$, and*
- *$a \|_W b$ iff $a >_W b$ and $b >_W a$.*

Consider the workflow log $W = \{ABCD, ACBD, AED\}$ (i.e., the log shown in Table 1). Relation $>_W$ describes which tasks appeared in sequence (one directly following the other). Clearly, $A >_W B$, $A >_W C$, $A >_W E$, $B >_W C$, $B >_W D$, $C >_W B$, $C >_W D$, and $E >_W D$. Relation $\rightarrow_W$ can be computed from $>_W$ and is referred to as the *(direct) causal relation* derived from workflow log W. $A \rightarrow_W B$, $A \rightarrow_W C$, $A \rightarrow_W E$, $B \rightarrow_W D$, $C \rightarrow_W D$, and $E \rightarrow_W D$. Note that $B \not\rightarrow_W C$ because $C >_W B$. Relation $\|_W$ suggests potential parallelism. For log W, tasks B and C seem to be in parallel, i.e., $B \|_W C$ and $C \|_W B$. If two tasks can follow each other directly in any order, then all possible interleavings are present and,

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

A→B
A→C
A→E
B→D
C→D
E→D

B||C
C||B

Definition 4.3 (Mining algorithm α). *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. $T_I = \{t \in T \mid \exists_{\sigma \in W} t = first(\sigma)\}$,
3. $T_O = \{t \in T \mid \exists_{\sigma \in W} t = last(\sigma)\}$,
4.

$$X_W = \{(A, B)$$

$$\wedge\ \forall_{a \in A} \forall$$

$$\wedge\ \forall_{b_1, b_2 \in}$$

5.

$$Y_W = \{(A, B$$

$$\wedge B \subseteq B' \Longrightarrow (A, B) = (A', B')\},$$

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. $\alpha(W) = (P_W, T_W, F_W)$.

$W = \{ABCD, ACBD, AED\}$

1. $T_W = \{A, B, C, D, E\}$,

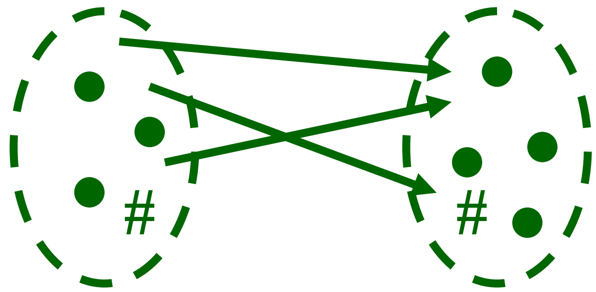
2. $T_I = \{A\}$,

3. $T_O = \{D\}$,

Definition 4.3 (Mining algorithm α). *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. $T_I = \{t \in T \mid \exists_{\sigma \in W} t = first(\sigma)\}$,
3. $T_O = \{t \in T \mid \exists_{\sigma \in W} t = last(\sigma)\}$,
4. $$X_W = \{(A, B) \mid A \subseteq T_W \ \wedge B \subseteq T_W \ \wedge \ \forall_{a \in A} \forall_{b \in B} a \rightarrow_W b \ \wedge \ \forall_{a_1, a_2 \in A} a_1 \#_W a_2 \ \wedge \ \forall_{b_1, b_2 \in B} b_1 \#_W b_2\},$$
5.

$\boxed{W = \{ABCD, ACBD, AED\}}$



A and B need to be non-empty.

4.

$$X_W = \{(\{A\},\{B\}),(\{A\},\{C\}),(\{A\},\{E\}), (\{B\},\{D\}),(\{C\},\{D\}),(\{E\},\{D\}), (\{A\},\{B,E\}),(\{A\},\{C,E\}),(\{B,E\},\{D\}), (\{C,E\},\{D\})\},$$

and

8. $\alpha(W) = (P_W, T_W, F_W)$.

4.

$$X_W = \{(\{A\},\{B\}),(\{A\},\{C\}),(\{A\},\{E\}),\\ (\{B\},\{D\}),(\{C\},\{D\}),(\{E\},\{D\}),\\ (\{A\},\{B,E\}),(\{A\},\{C,E\}),(\{B,E\},\{D\}),\\ (\{C,E\},\{D\})\},$$

5.

$$Y_W = \{(A,B) \in X_W \mid \forall_{(A',B')\in X_W} A \subseteq A' \\ \wedge B \subseteq B' \Longrightarrow (A,B) = (A',B')\},$$

$$W = \{ABCD, ACBD, AED\}$$

5.

$$Y_W = \{(\{A\},\{B,E\}),(\{A\},\{C,E\}),(\{B,E\},\\ \{D\}),(\{C,E\},\{D\})\},$$

and

8. $\alpha(W) = (P_W, T_W, F_W)$.

5.

$$Y_W = \{(\{A\},\{B,E\}),(\{A\},\{C,E\}),(\{B,E\},\{D\}),(\{C,E\},\{D\})\},$$

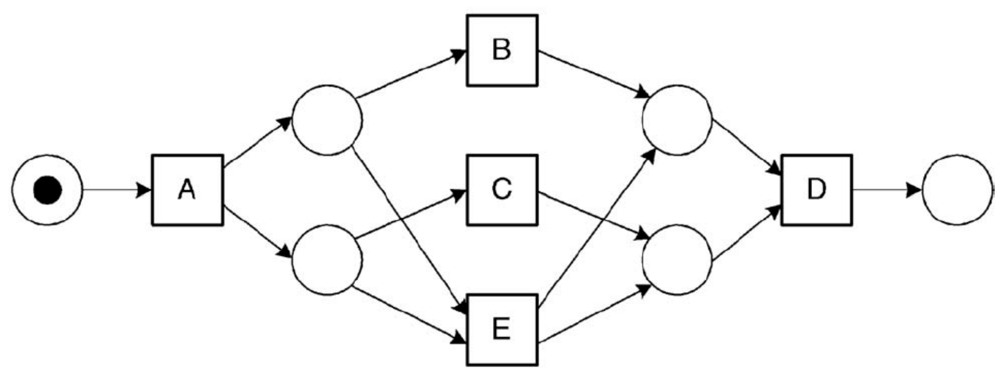
$\wedge\ \forall_{a\in A}\forall_{b\in B} a \rightarrow_W b\ \wedge\ \forall_{a_1,a_2\in A} a_1 \#_W a_2$

$\wedge\ \forall_{b_1,b_2\in B} b_1 \#_W b_2\},$

$W = \{ABCD, ACBD, AED\}$

$$P_W = \{i_W, o_W, p_{(\{A\},\{B,E\})}, p_{\{A\},\{C,E\})}, p_{(\{B,E\},\{D\})}, p_{(\{C,E\},\{D\})}\},$$

$$F_W = \{(i_W, A), (A, p_{(\{A\},\{B,E\})}), (p_{(\{A\},\{B,E\})}, B), \ldots, (D, o_W)\},$$



✓ 6. $P_W = \{p_{(A,B)} \mid (A,B) \in Y_W\} \cup$

✓ 7.

$F_W = \{(a, p_{(A,B)}) \mid (A,B) \in Y_W \wedge$

$\cup\ \{(p_{(A,B)}, b) \mid (A,B) \in Y_W$

$\cup \{(i_W, t) \mid t \in T_I\} \cup \{(t, o$

and

✓ 8. $\alpha(W) = (P_W, T_W, F_W)$.

Exercise (2)

- **What does the alpha algorithm produce for a log consisting only of the following traces?**
  - **ACD**
  - **BCE**

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. $\alpha(W) = (P_W, T_W, F_W)$.

- Direct succession: **$x>y$ iff for some case x is directly followed by y.**
- Causality: **$x \rightarrow y$ 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 $y>x$.**

Exercise (3)

- **What does the alpha algorithm produce for a log consisting only of the following traces?**
  - **ACEG**
  - **AECG**
  - **BDFG**
  - **BFDG**

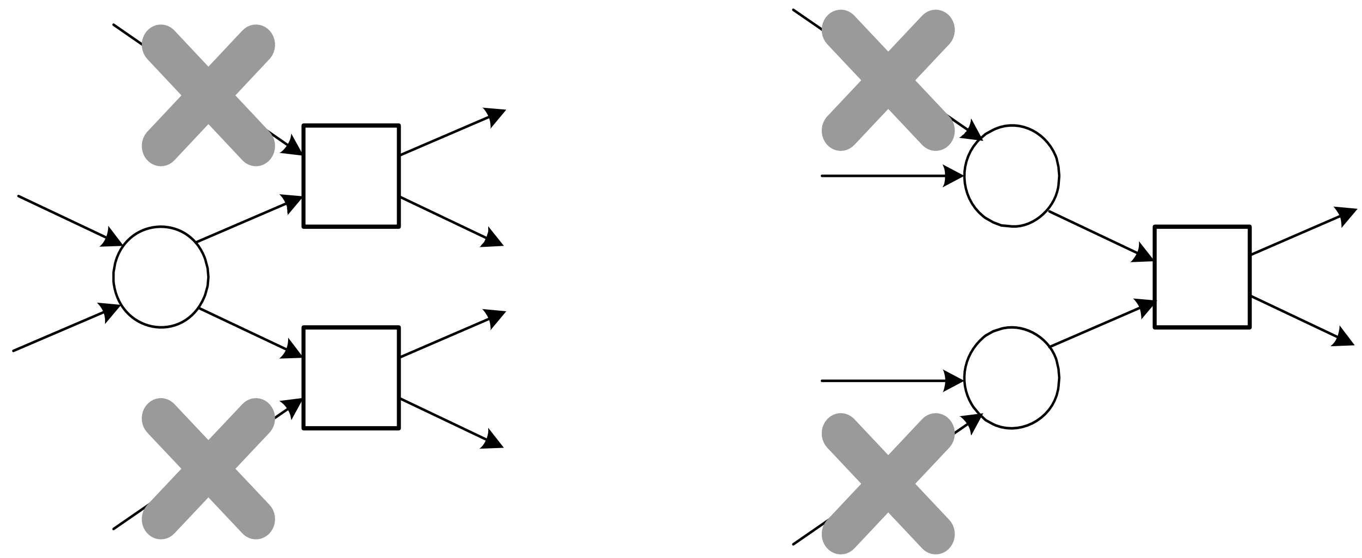
Let W be a workflow log over T. $\alpha(W)$ is defined as follows.

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. $\alpha(W) = (P_W, T_W, F_W)$.

- 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.**

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:**

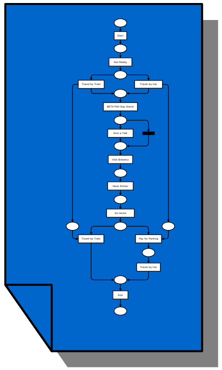


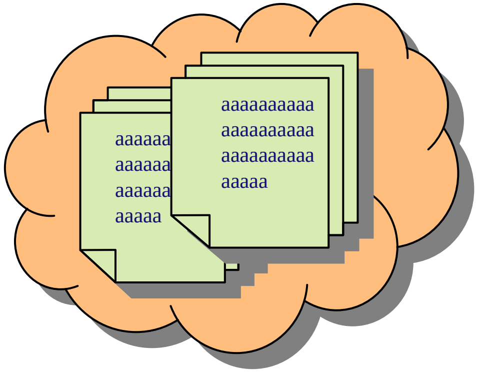
(Short loops require some refinement but not a problem.)

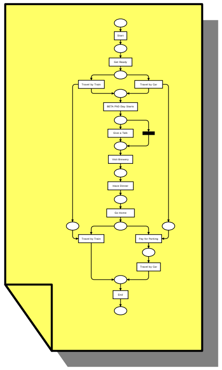
Alpha algorithm

- 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!)

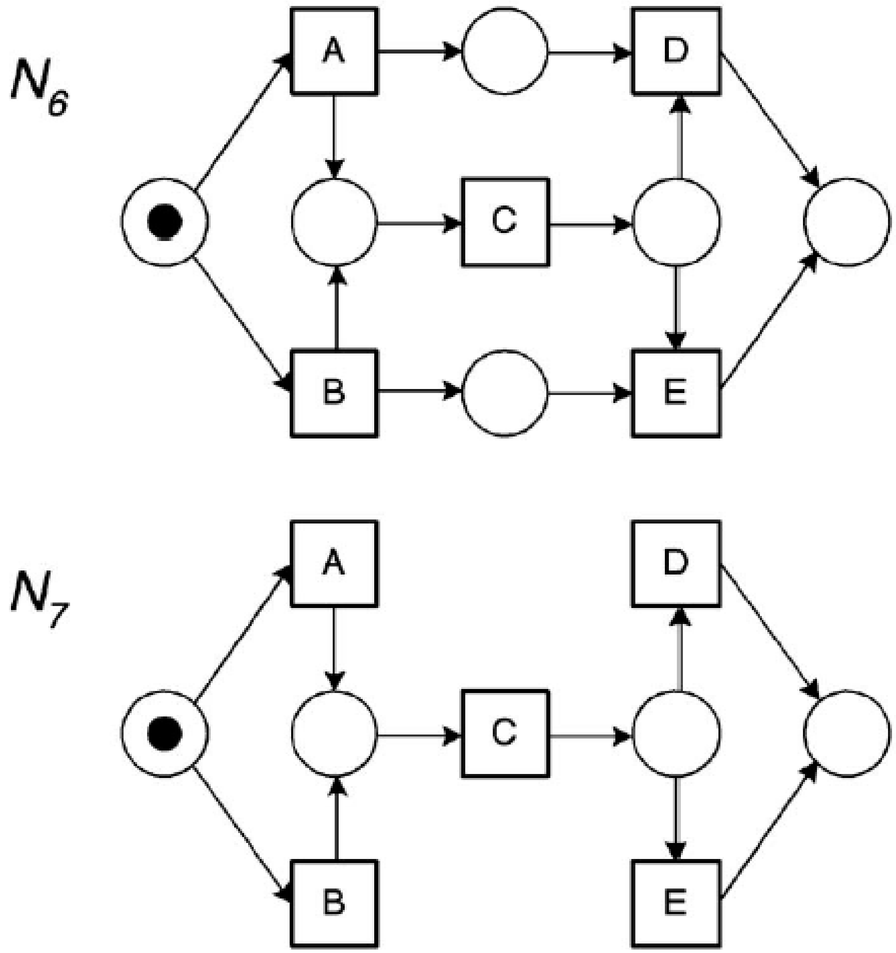


Fig. 7. The nonfree-choice WF-net N_6 cannot be rediscovered by the α algorithm.

Log only contains information about behavior and not structure

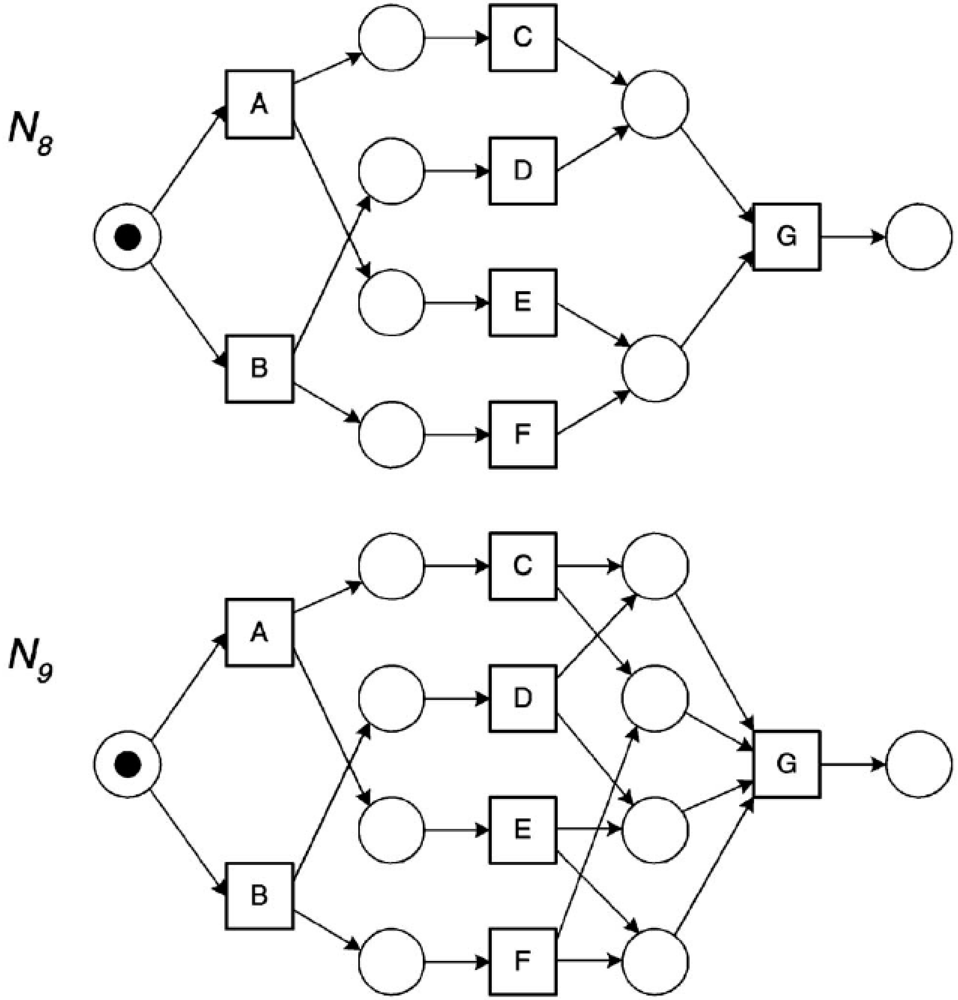


Fig. 8. WF-net N_8 cannot be rediscovered by the α algorithm. Nevertheless, α returns a WF-net which is behavioral equivalent.

Completeness notion may be too crude in some cases

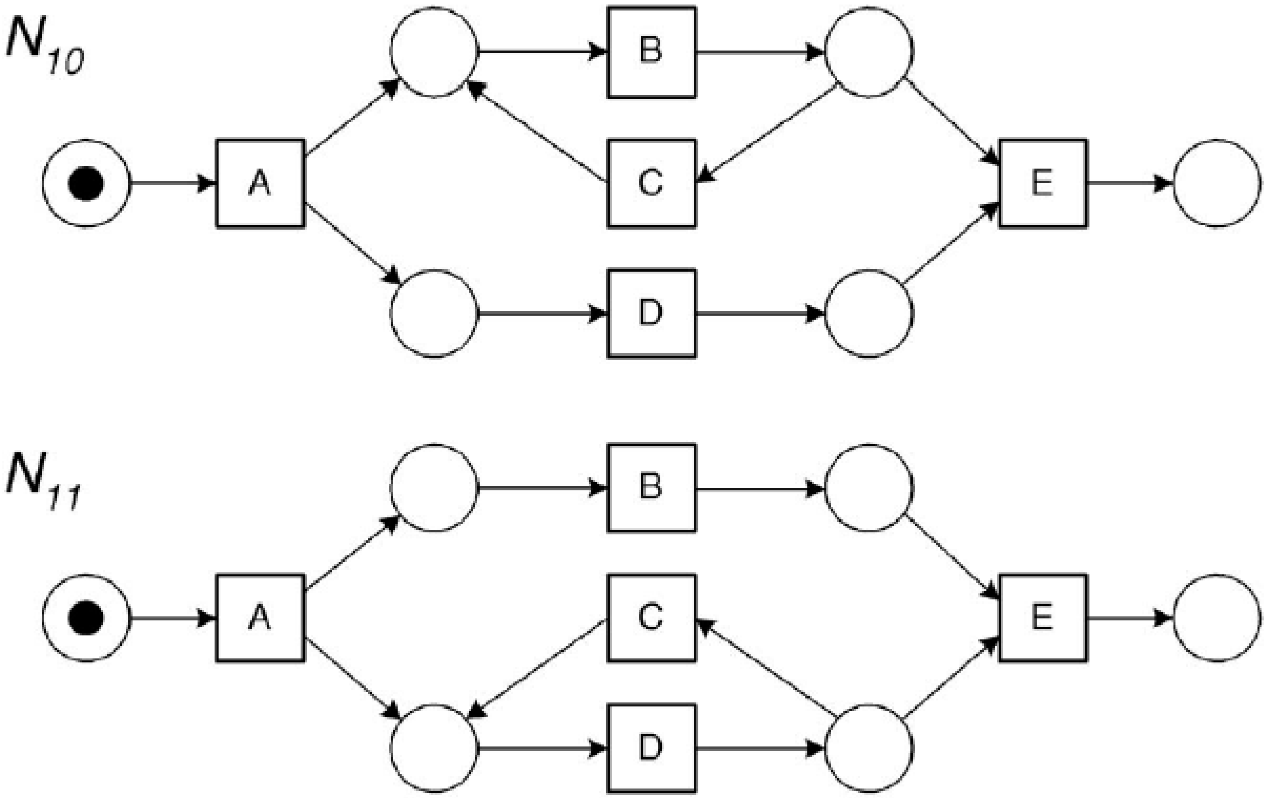


Fig. 9. Although both WF-nets are not behavioral equivalent they are identical with respect to $>$.

Another example of behaviorally equivalent SWF-nets

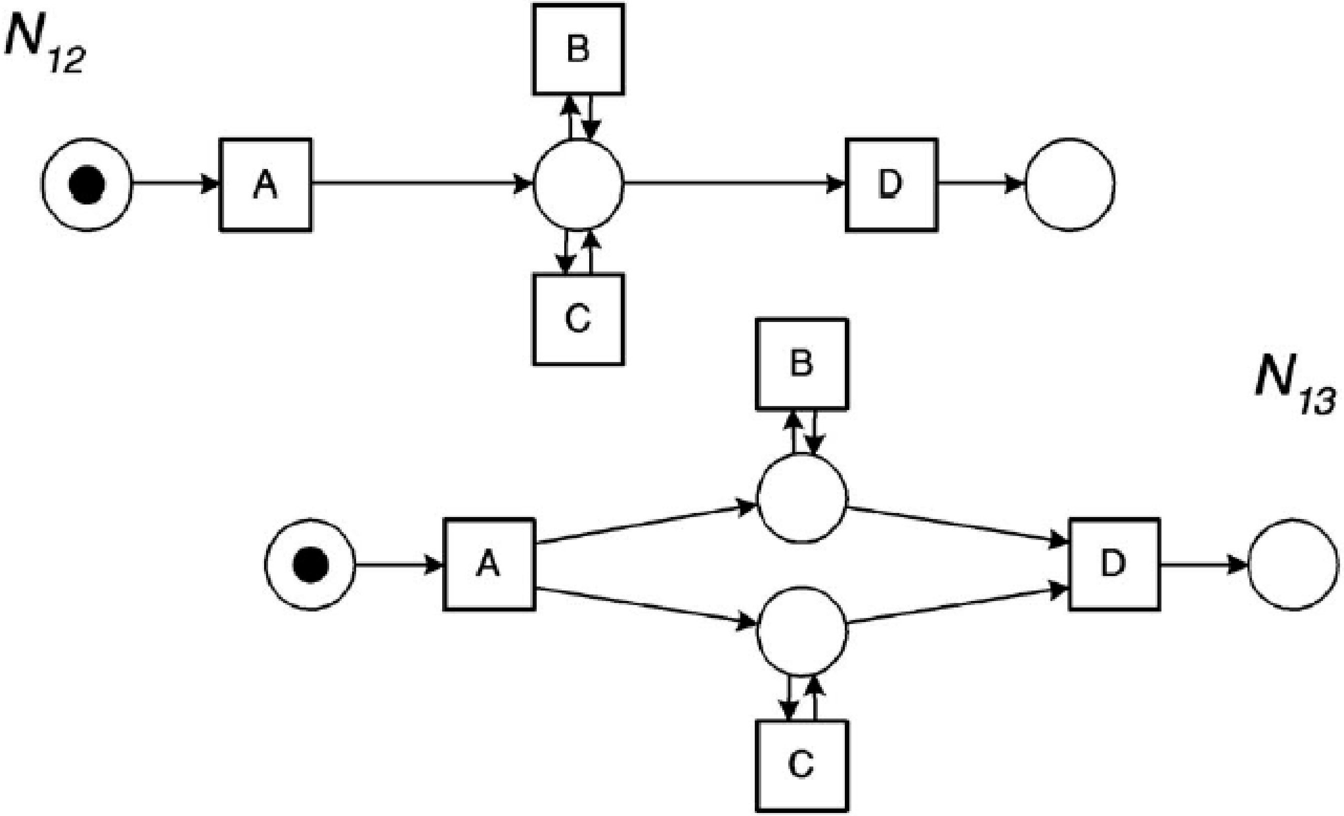
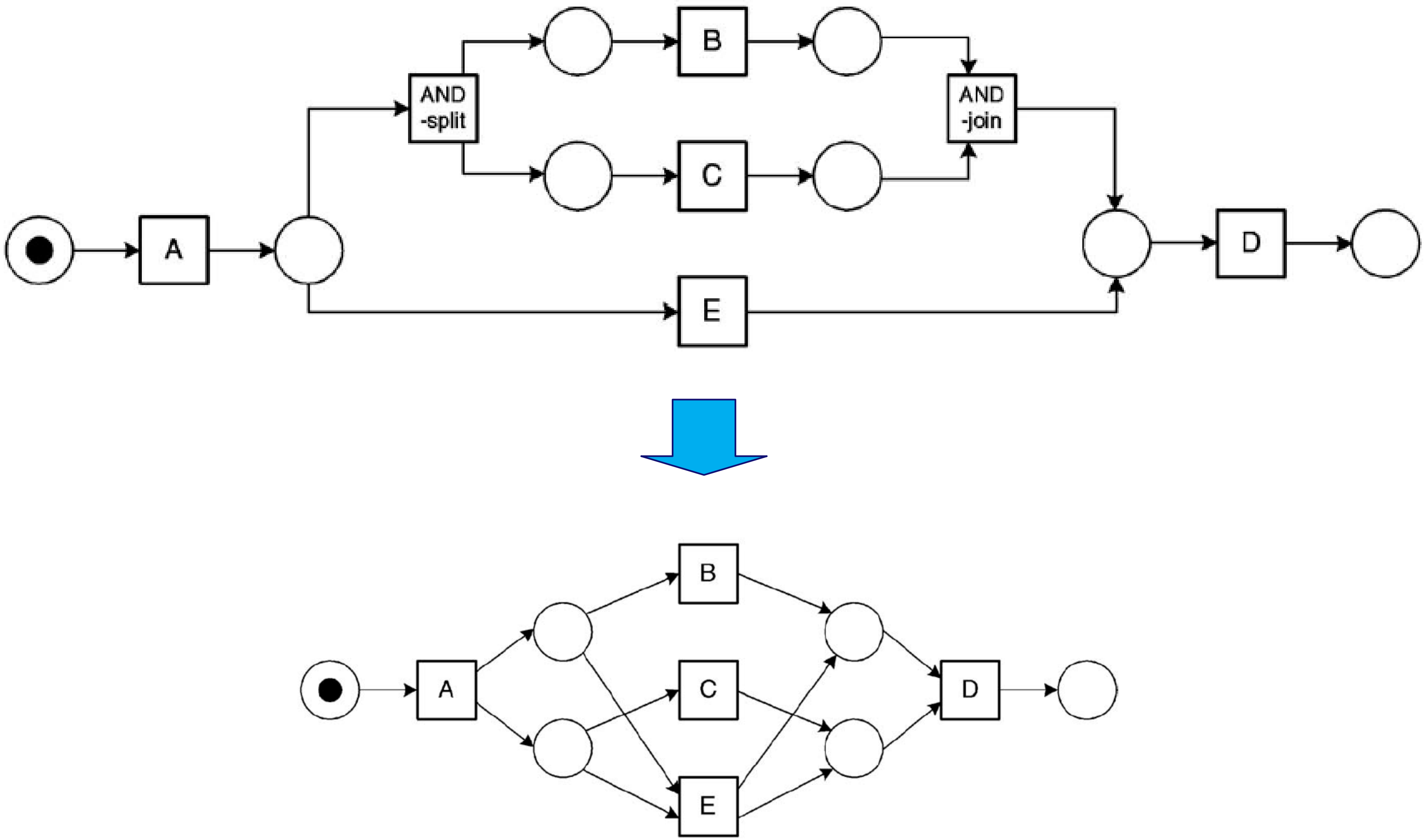
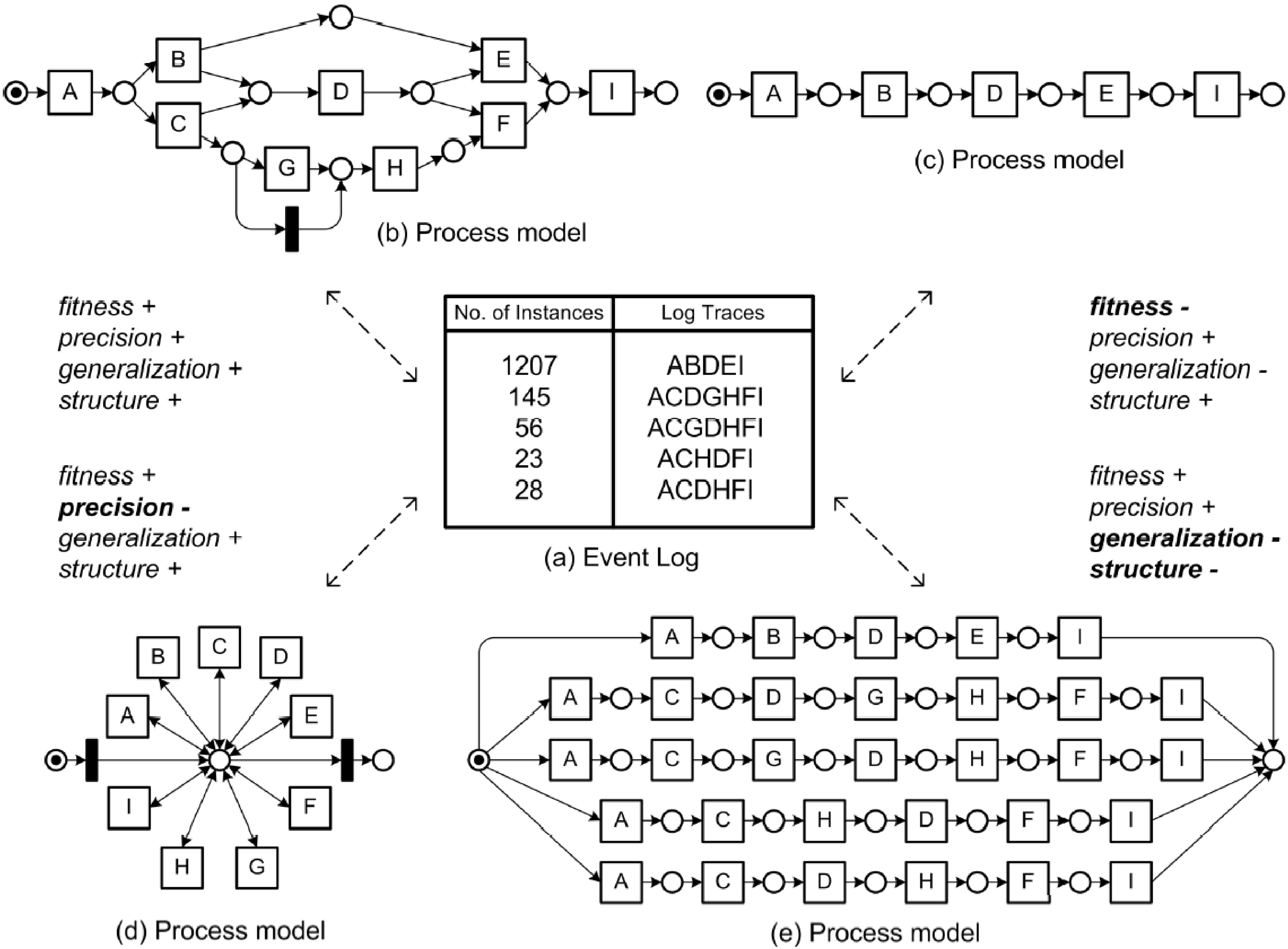


Fig. 10. Both SWF-nets are behavioral equivalent and, therefore, any algorithm will be unable to distinguish N_{12} from N_{13} (assuming a notion of completeness based on $>$).

Silent steps (and duplicate steps) cannot be discovered





No. of Instances	Log Traces
1207	ABDEI
145	ACDGHFI
56	ACGDHFI
23	ACHDFI
28	ACDHFI

(a) Event Log

(b) Process model

(c) Process model

(d) Process model

(e) Process model

Simple process mining algorithms tend to:

- Have problems with complex control-flow constructs. **For example, many process mining algorithms are unable to deal with non-free-choice constructs and complex nested loops.**
- Not allow for duplicates. **In the event log it is not possible to distinguish between activities that are logged in a similar way, i.e., there are multiple activities that have the same "footprint" in the log. As a result, most algorithms map these different activities onto a simple activity thus making the model incorrect or counter-intuitive.**
- Silent steps. **Things that are not recorded cannot be discovered.**
- Underfit (i.e., overgeneralize) or overfit. **Many algorithms have a tendency to overgeneralize, i.e., the discovered model allows for much more behavior than actually recorded in the log. In some circumstances this may be desirable. However, there seems to be a need to flexibly balance between "overfitting" and "underfitting".**
- Yield inconsistent models. **For more complicated processes many algorithms have a tendency to produce models that may have deadlocks and/or livelocks. It seems vital that the generated models satisfy some soundness requirements (e.g., the soundness property).**

Other Process Discovery Techniques

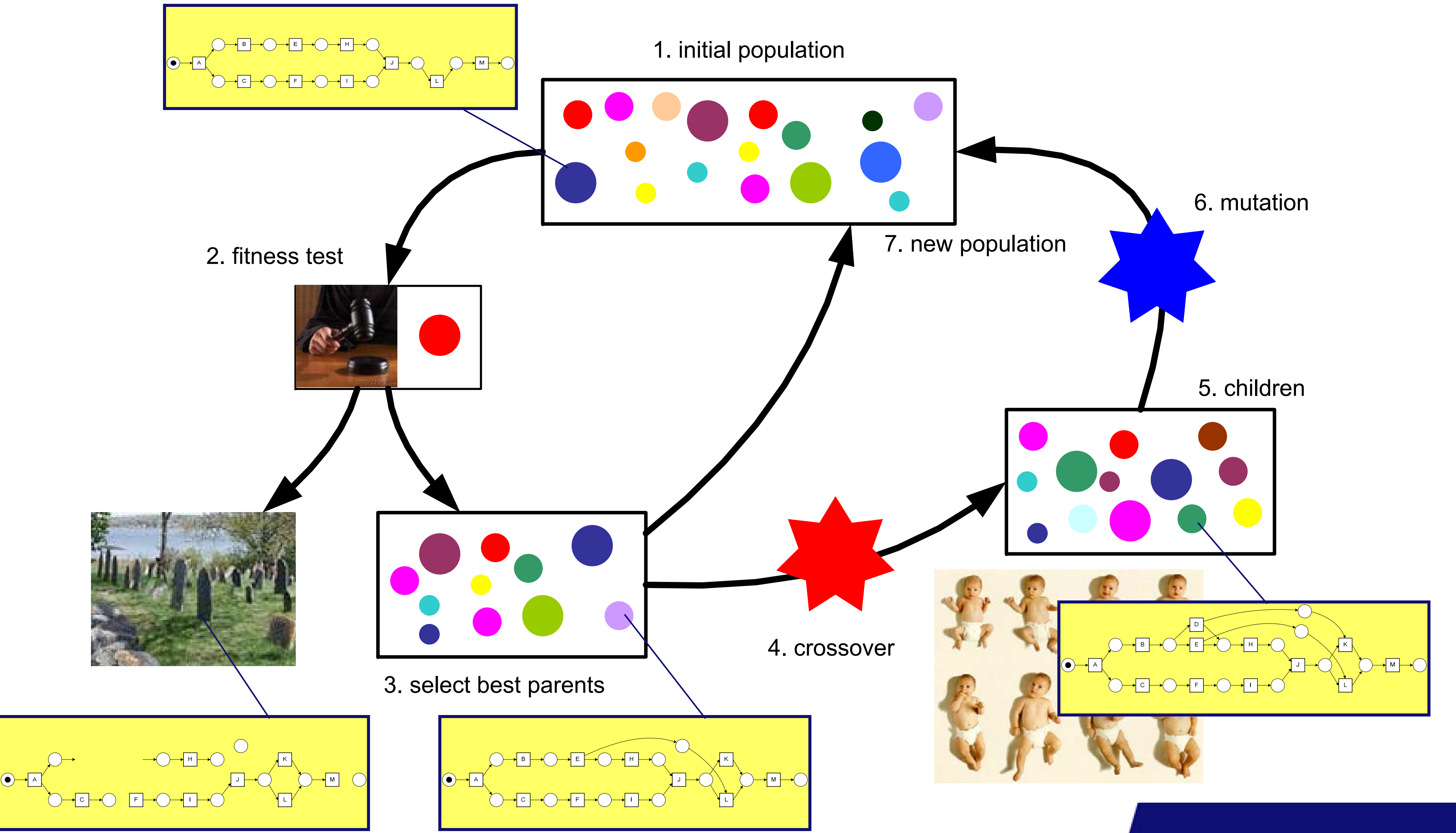
Technische Universiteit
Eindhoven
University of Technology

Where innovation starts

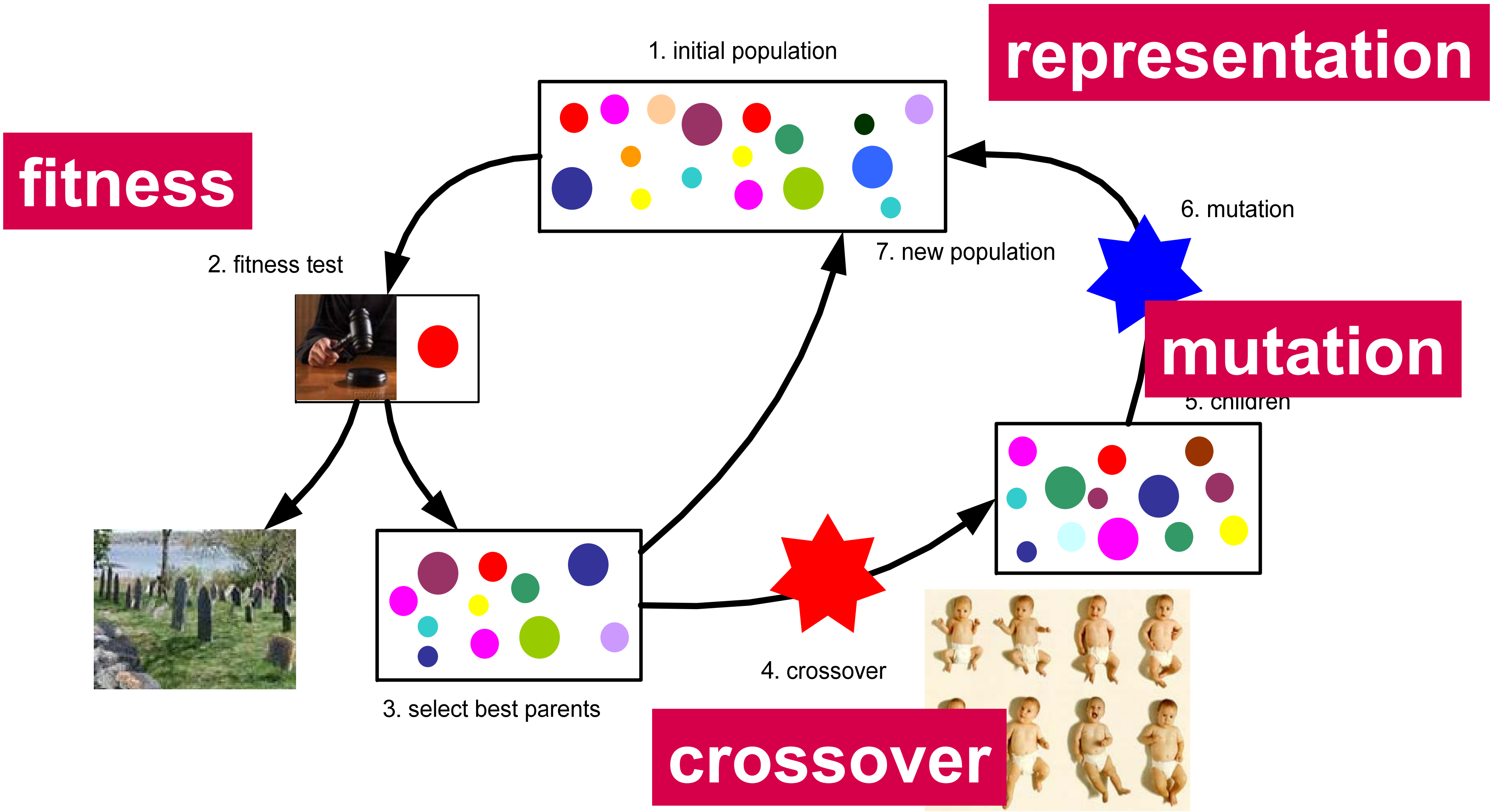
Overview of process discovery techniques

- 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

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

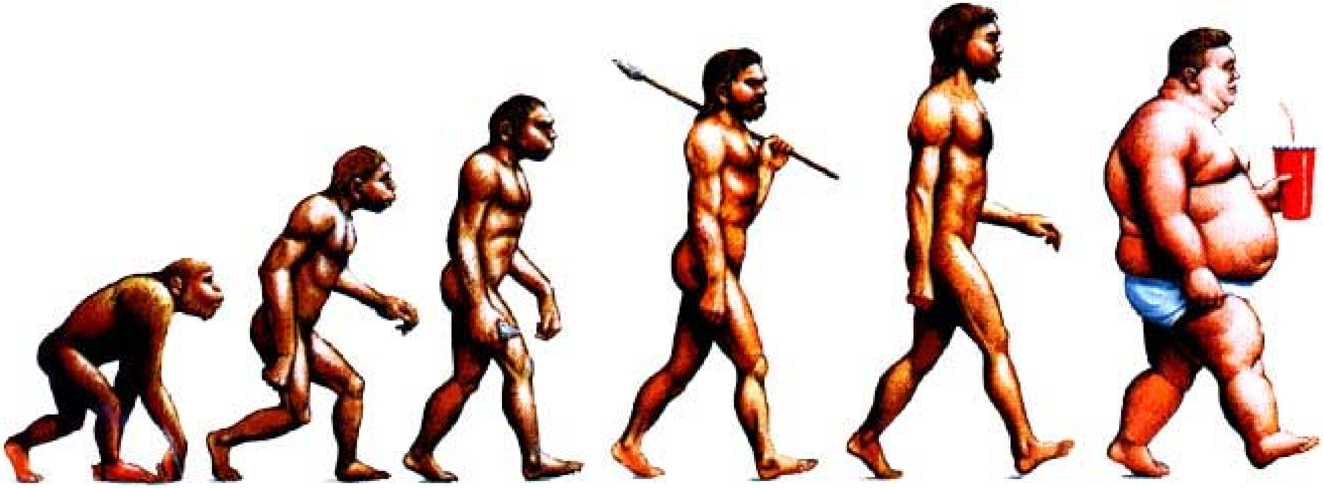


Design choices



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.).



Balancing Between Overfitting and Underfitting

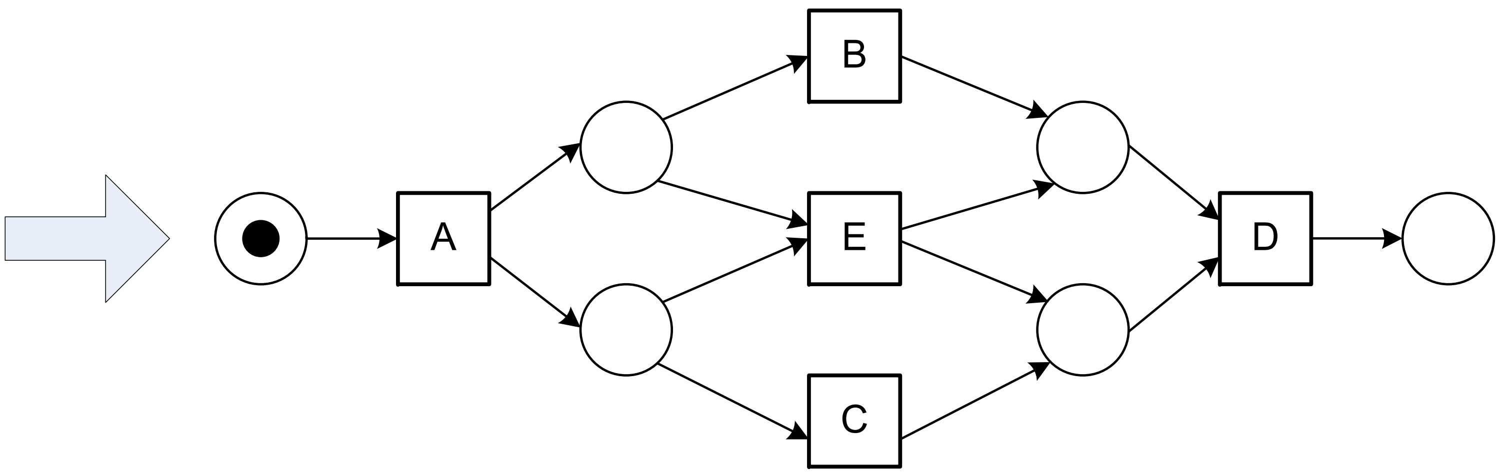
Technische Universiteit
Eindhoven
University of Technology

Where innovation starts

Challenge: Balancing Between Underfitting and Overfitting

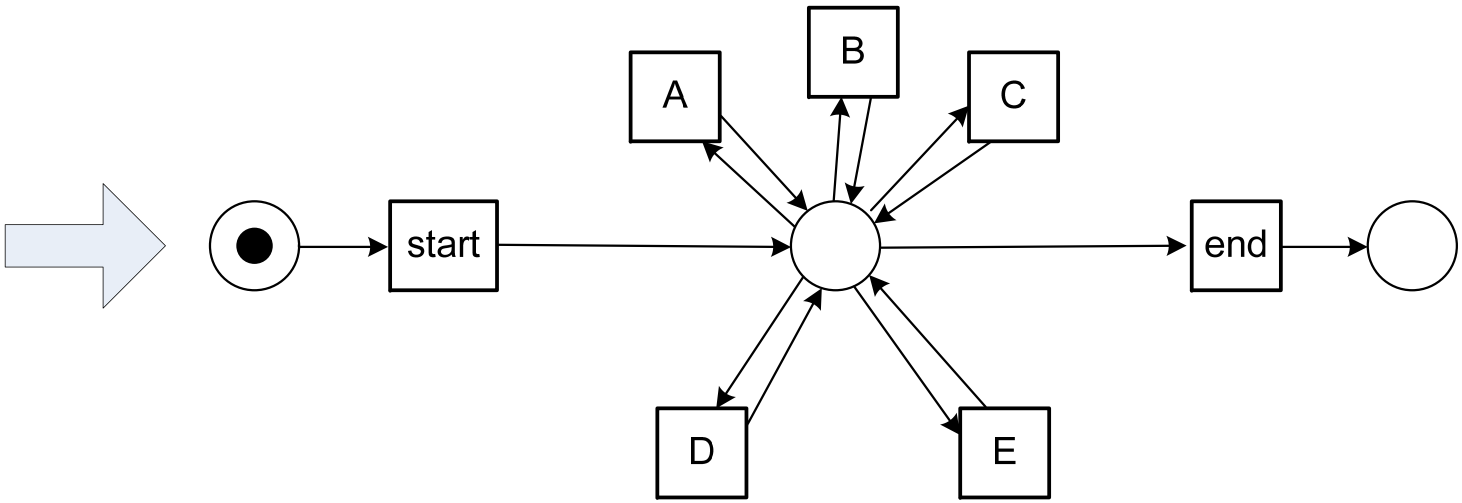
The essence

ABCD
ACBD
AED
ABCD
ABCD
AED
ACBD
...

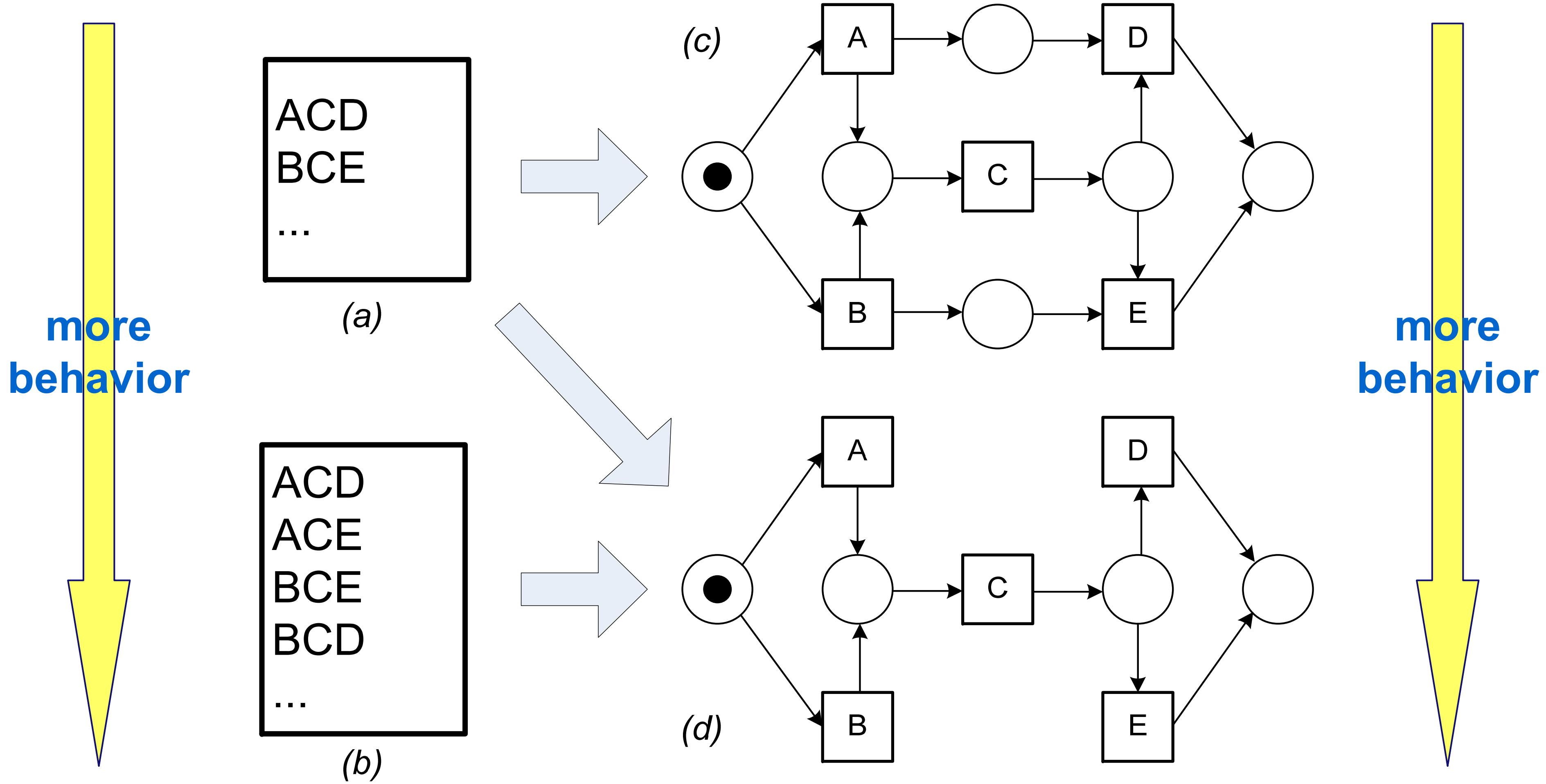


But ...

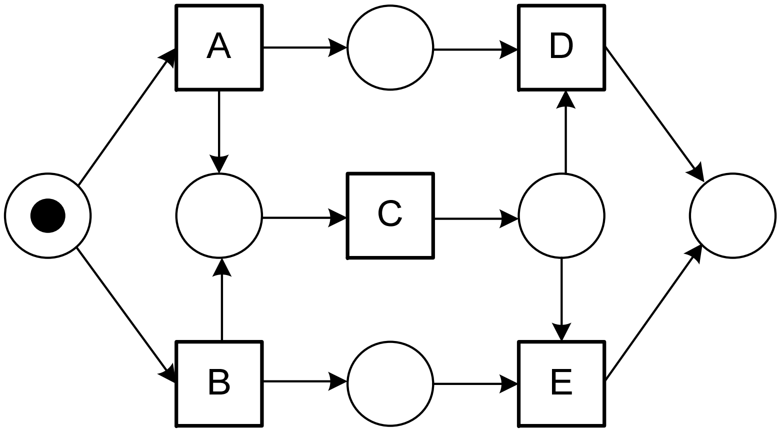
Any log containg activities A, B, C, D, and E.

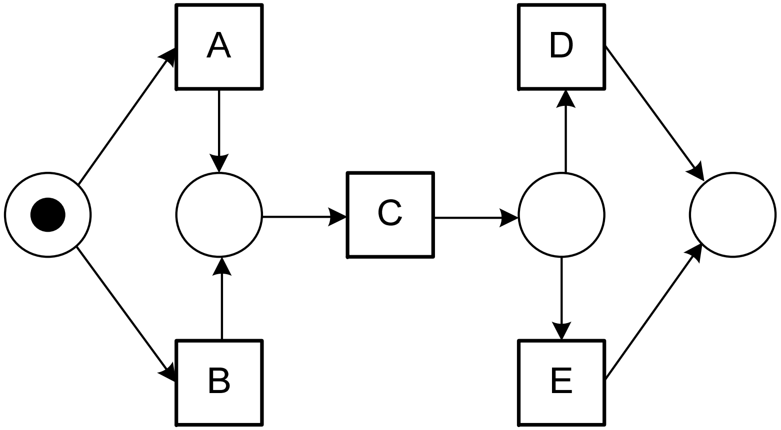


Finding a balance

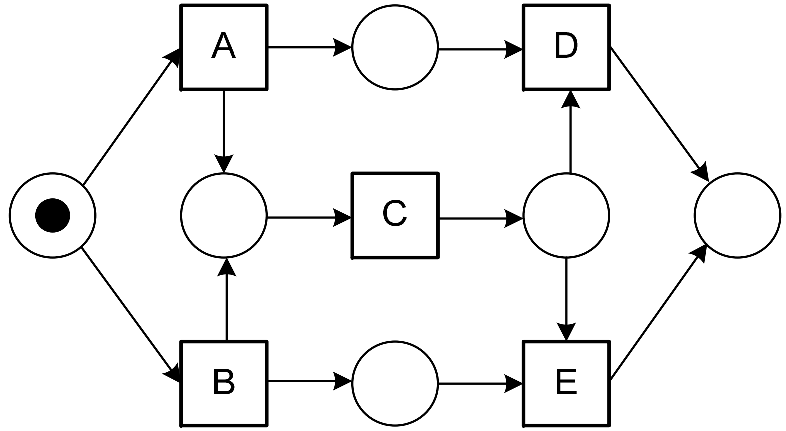


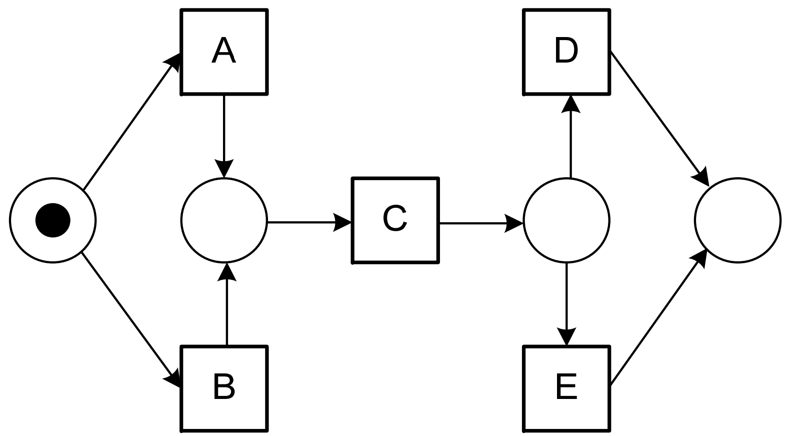
ACD **99**
ACE **0**
BCE **85**
BCD **0**



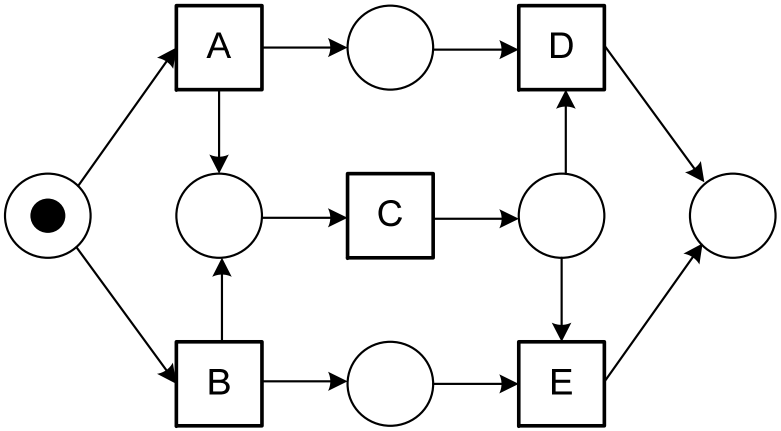


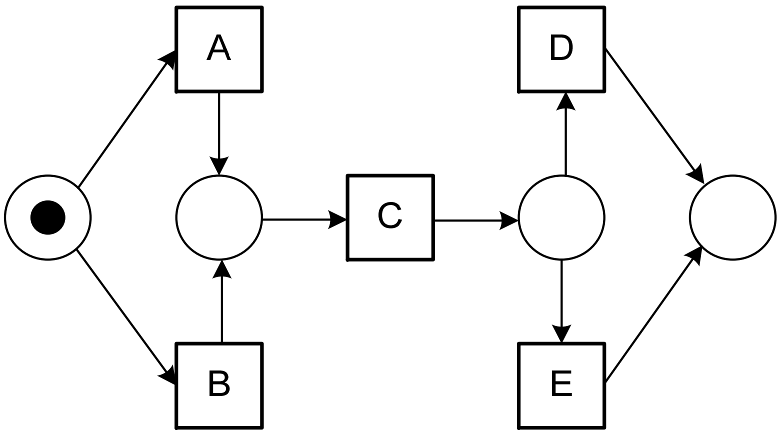
ACD **99**
ACE **88**
BCE **85**
BCD **78**





ACD	**99**
ACE	**2**
BCE	**85**
BCD	**3**





Important observations

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

Relevance

See examples in Part I !!!!!

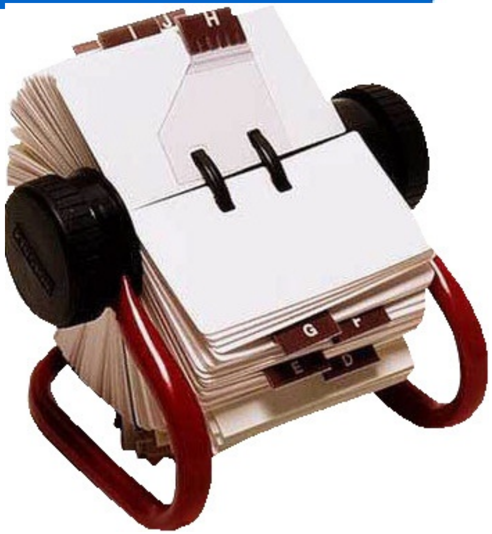
Discovering Other Perspectives

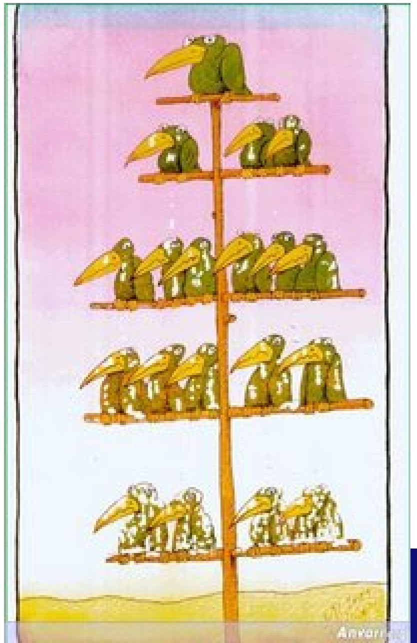
TU/e Technische Universiteit Eindhoven University of Technology

Where innovation starts

Perspectives

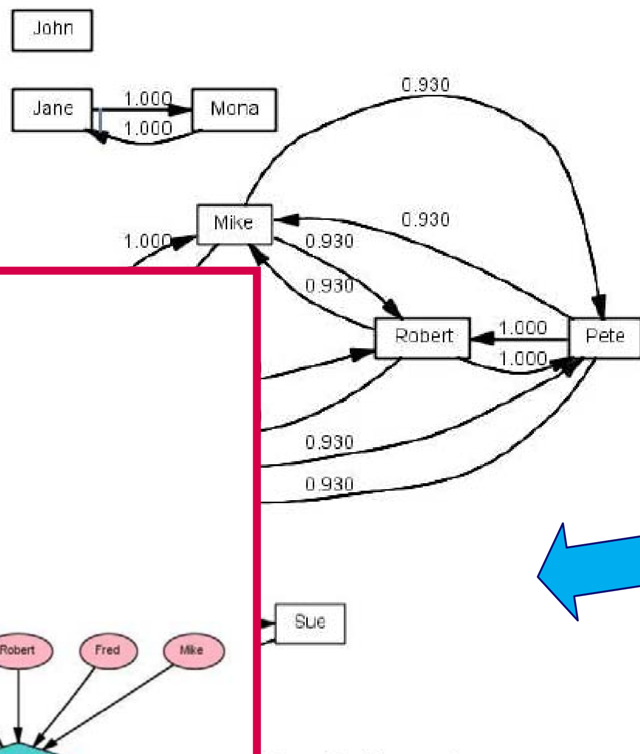
- **Control-flow perspective**
  - **As before ...**
- **Data perspective**
  - **How does data flow from one task to another?**
  - **What data is influencing decisions?**
  - **What are the (data-driven) business rules?**
- **Organizational perspective**
  - **Who is doing what?**
  - **Who is working with who?**
  - **What are the (real) roles in an organization?**
- **...**



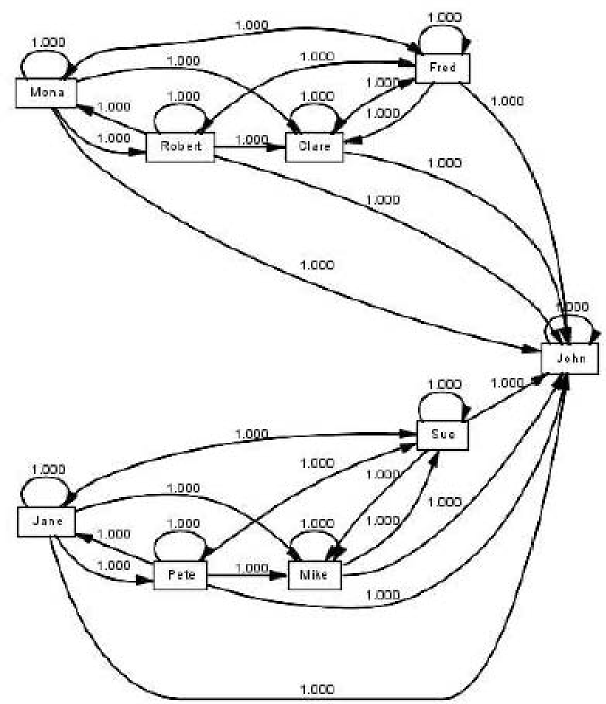


Examples of social network mining

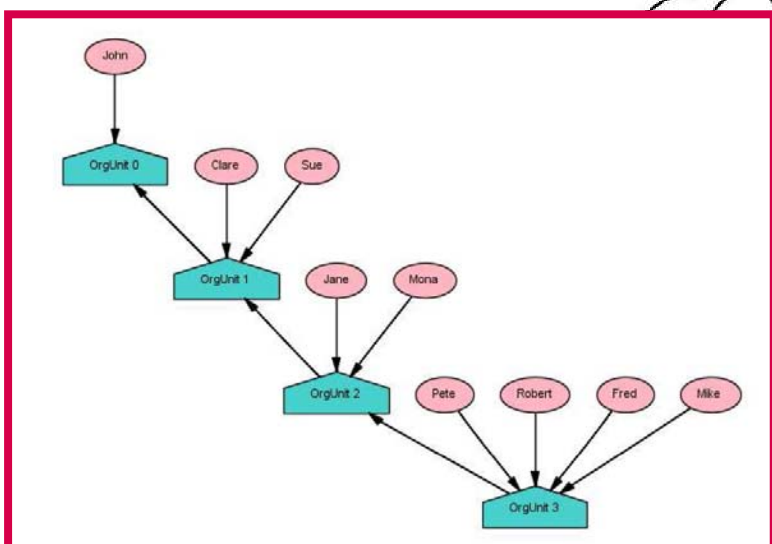
(Minseok Song et al.)



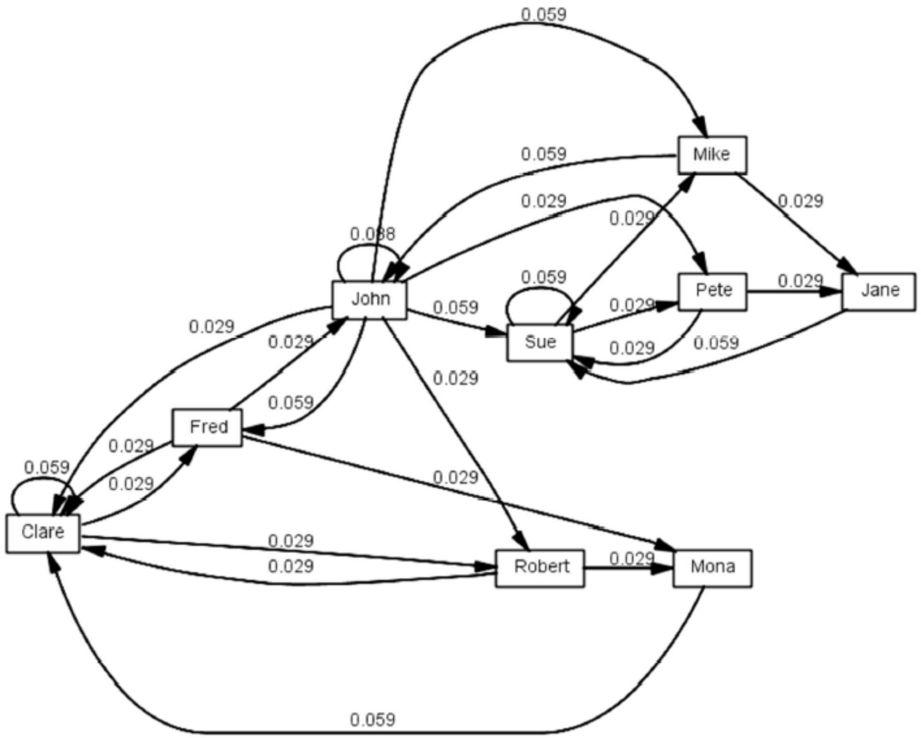
ing similar task

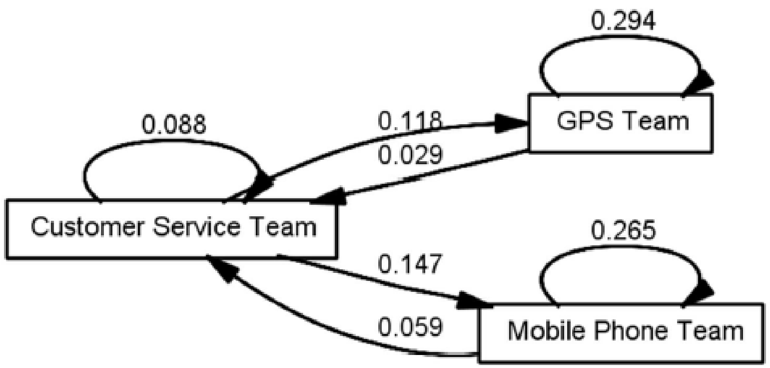


(b) working together

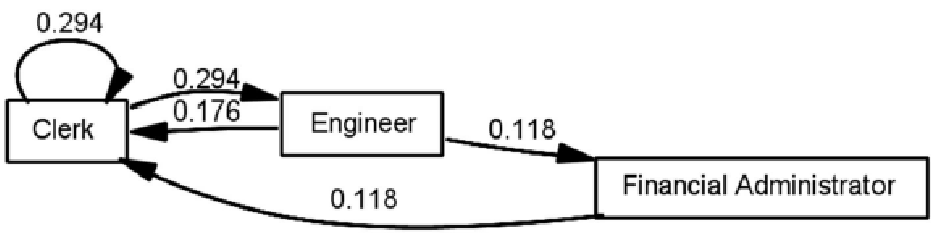


Social networks based on hand-over of work



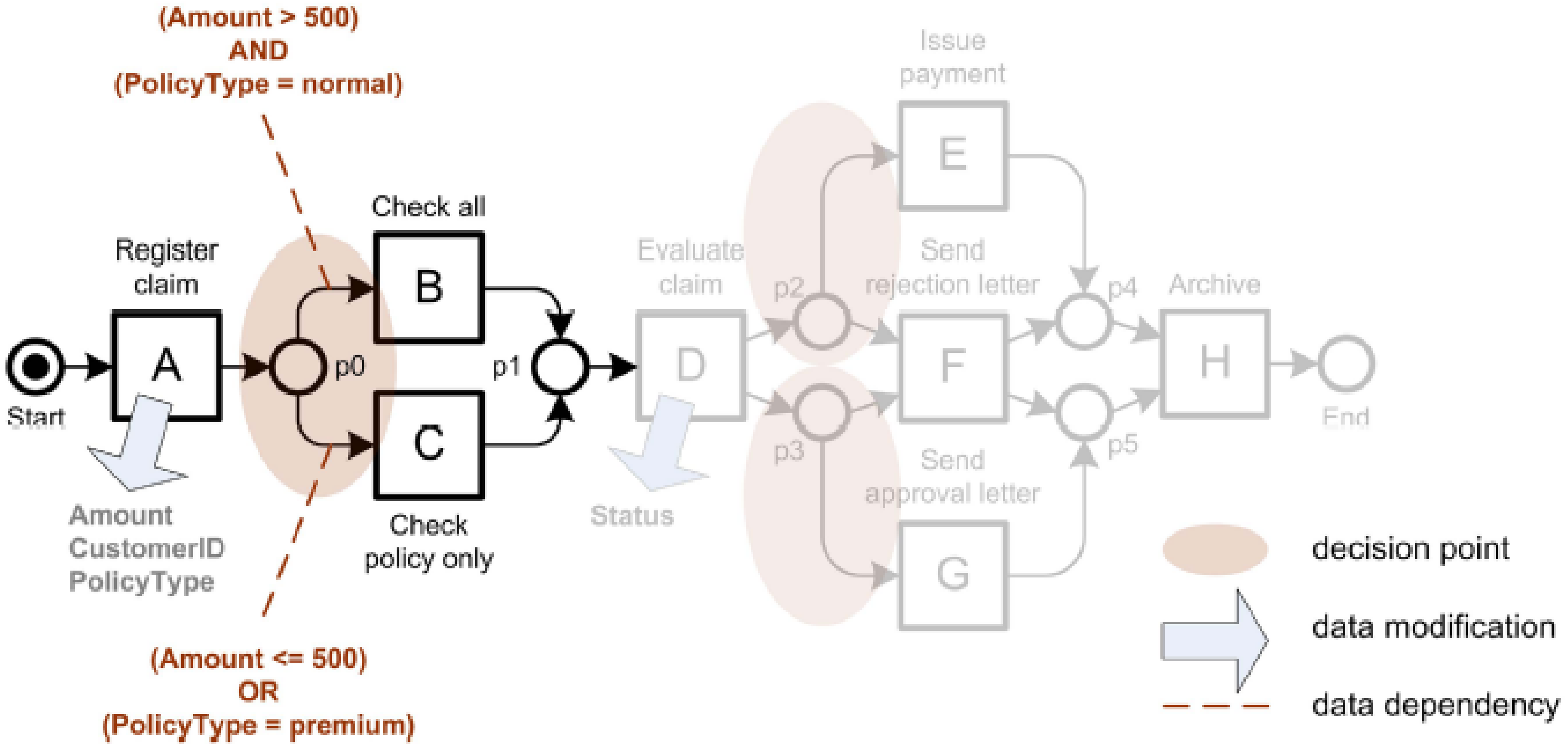


(a) social network for organizational units



(b) social network for roles

Decision mining
(Anne Rozinat et al.)

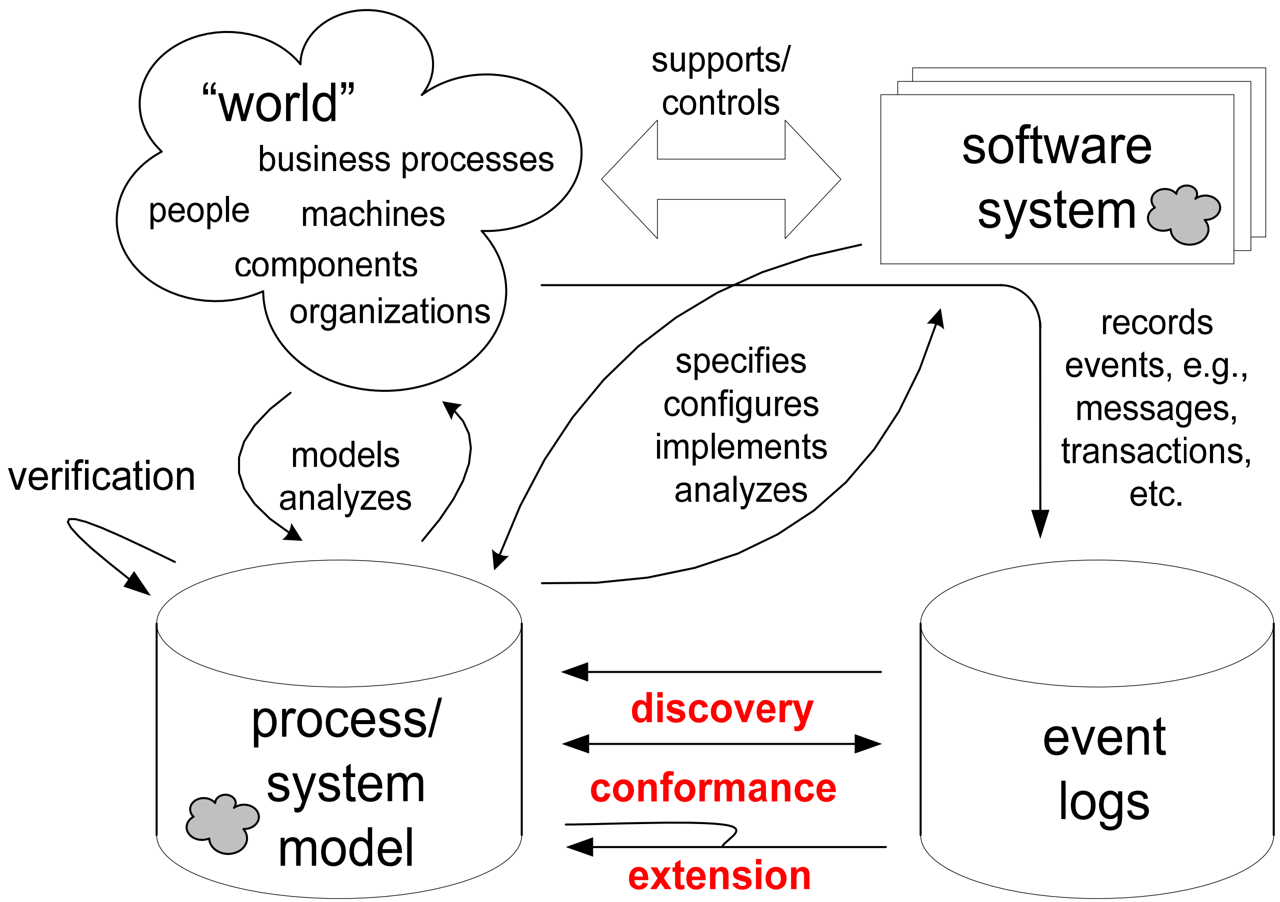


Conformance Checking and Extension

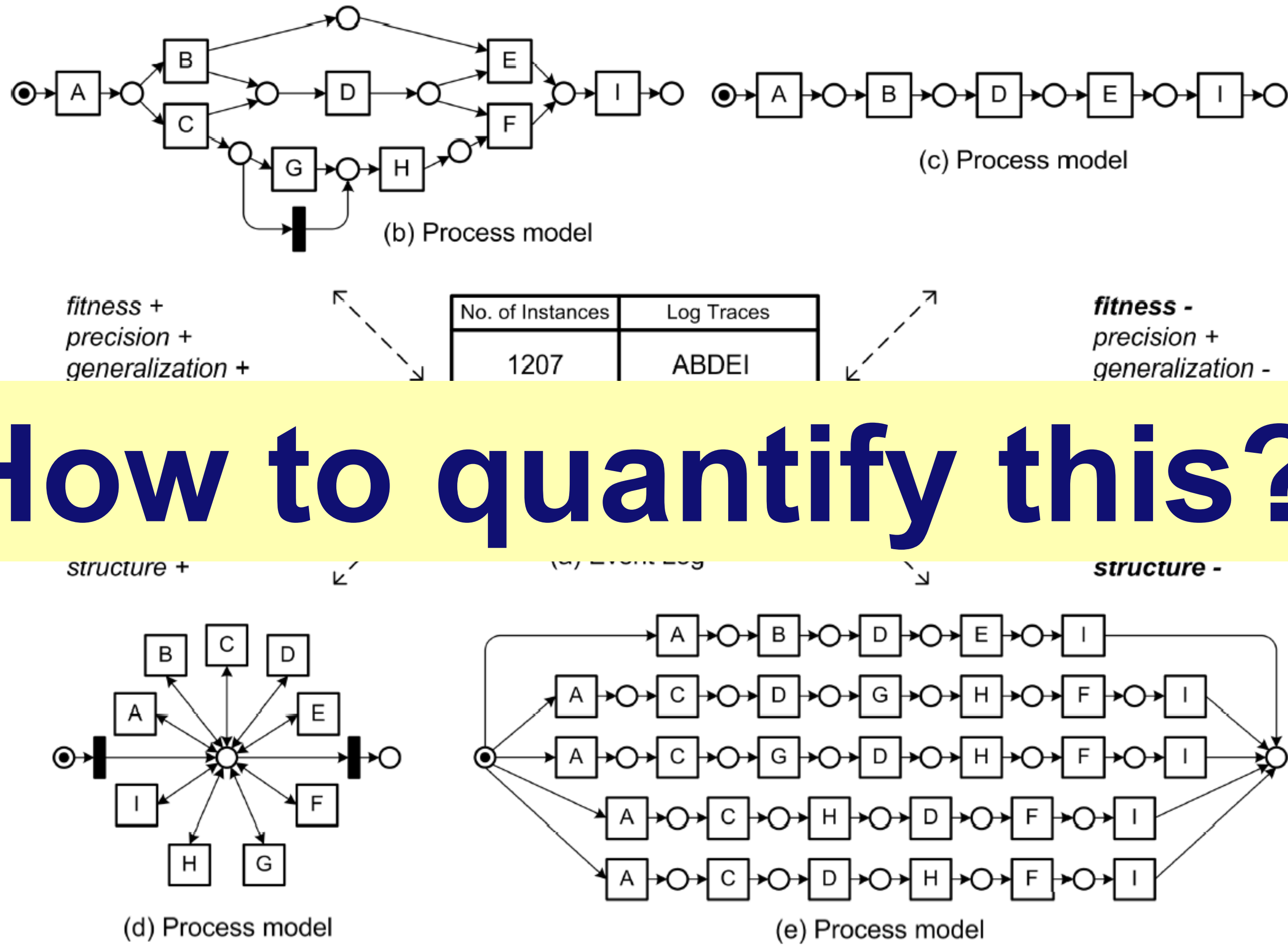
Technische Universiteit
Eindhoven
University of Technology

Where innovation starts

Conformance and Extension



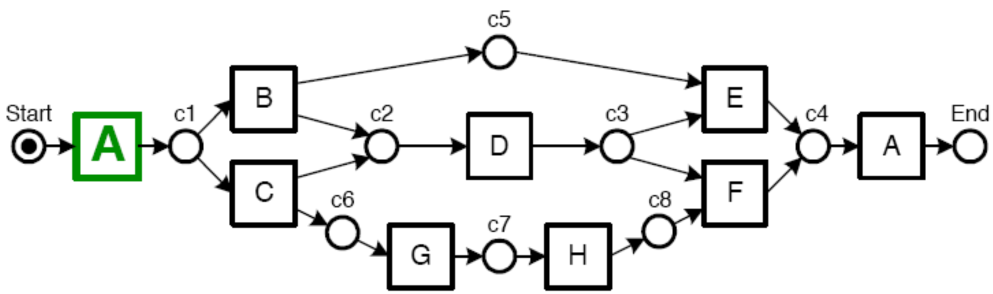
Conformance checker
(Anne Rozinat et al.)



How to quantify this?

Fitness by replay

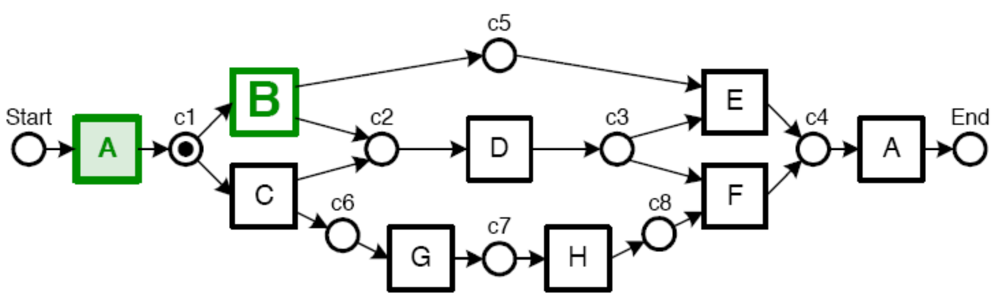
(a)



m = 0
r = 0
c = 0
p = 1

No. of Instances	Log Traces
1207	ABDEA
145	ACDGHFA
56	ACGDHFA
23	ACHDFA
28	ACDHFA

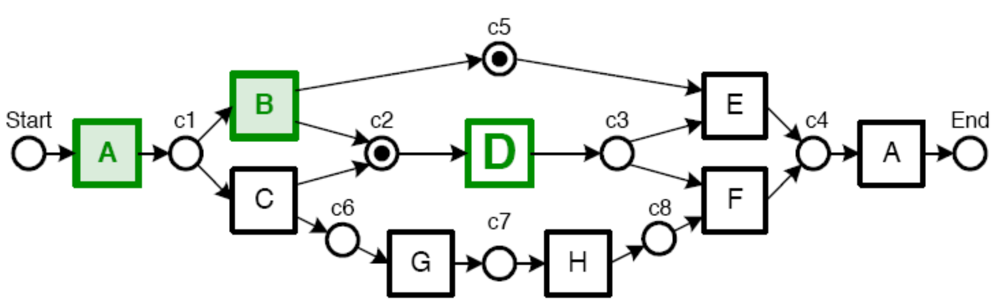
(b)



m = 0
r = 0
c = 1
p = 2

No. of Instances	Log Traces
1207	ABDEA
145	ACDGHFA
56	ACGDHFA
23	ACHDFA
28	ACDHFA

(c)



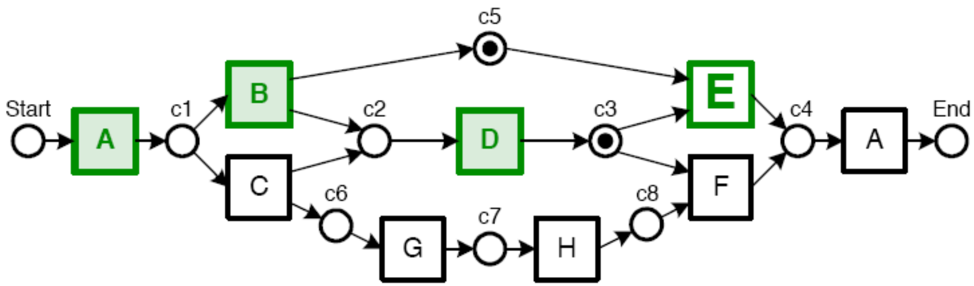
m = 0
r = 0
c = 2
p = 4

No. of Instances	Log Traces
1207	ABDEA
145	ACDGHFA
56	ACGDHFA
23	ACHDFA
28	ACDHFA

m=missing,r=remaining,c=consumed,p=produced

No problem (m=0, r=0)

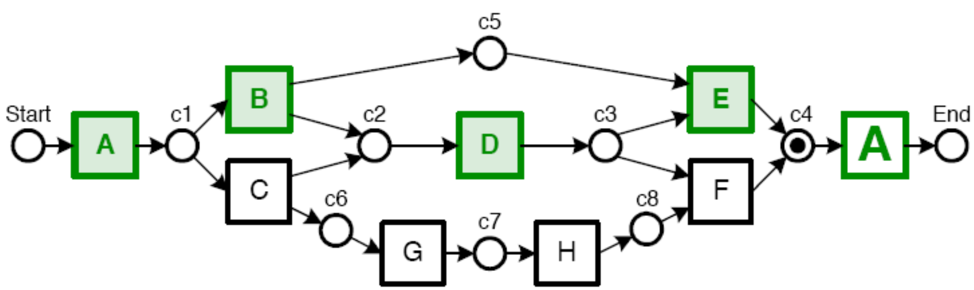
(d)



m = 0
r = 0
c = 3
p = 5

No. of Instances	Log Traces
1207	→ ABDEA
145	ACDGHFA
56	ACGDHFA
23	ACHDFA
28	ACDHFA

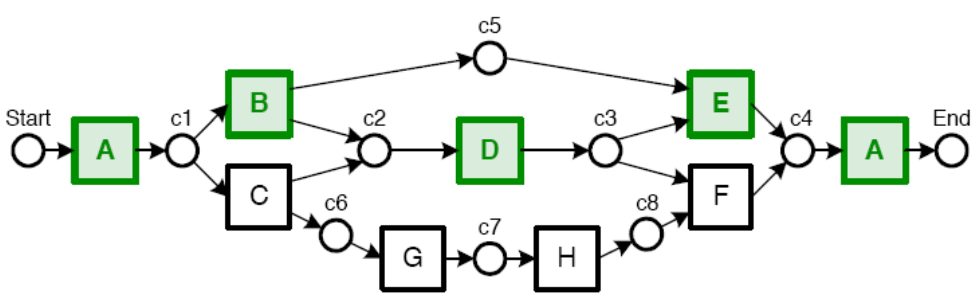
(e)



m = 0
r = 0
c = 5
p = 6

No. of Instances	Log Traces
1207	→ ABDEA
145	ACDGHFA
56	ACGDHFA
23	ACHDFA
28	ACDHFA

(f)

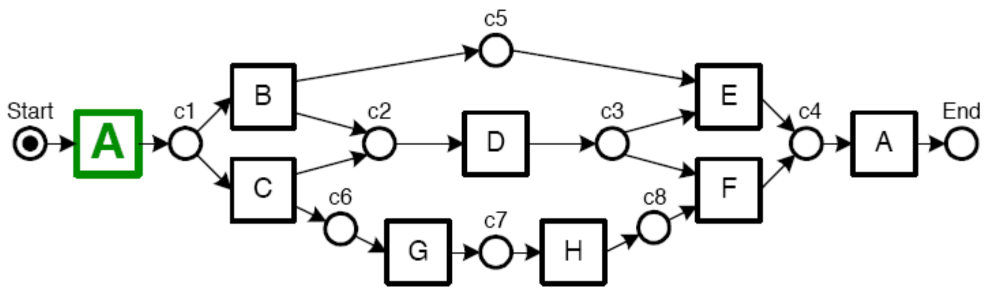


m = 0
r = 0
c = 7
p = 7

No. of Instances	Log Traces
1207	→ ABDEA
145	ACDGHFA
56	ACGDHFA
23	ACHDFA
28	ACDHFA

Another (impossible) trace

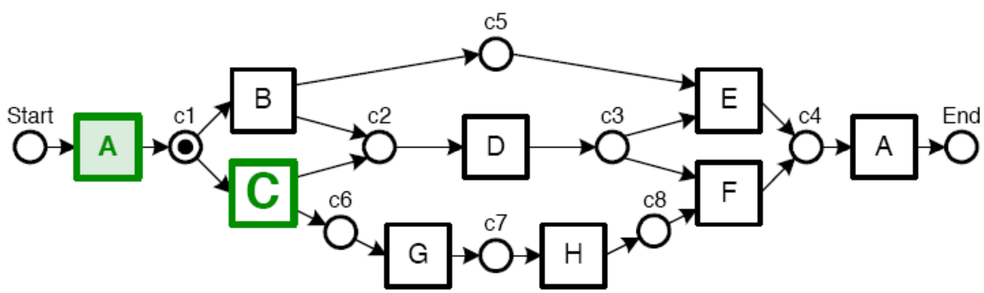
(a)



m = 0
r = 0
c = 0
p = 1

No. of Instances	Log Traces
1207	ABDEA
145	ACDGHFA
56	ACGDHFA
23	ACHDFA
28	ACDHFA

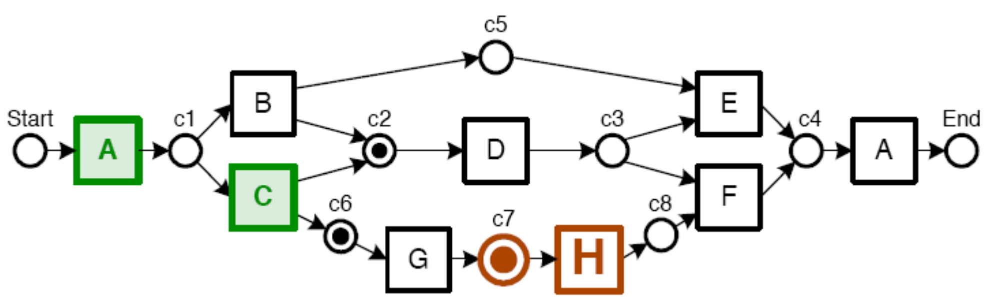
(b)



m = 0
r = 0
c = 1
p = 2

No. of Instances	Log Traces
1207	ABDEA
145	ACDGHFA
56	ACGDHFA
23	ACHDFA
28	ACDHFA

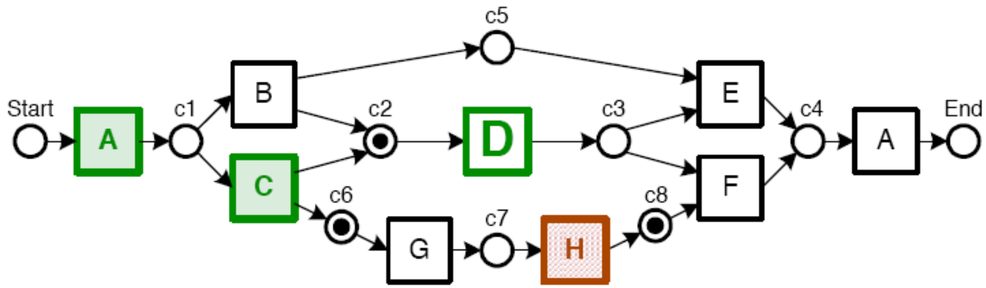
(c)



m = 1
r = 0
c = 2
p = 4

No. of Instances	Log Traces
1207	ABDEA
145	ACDGHFA
56	ACGDHFA
23	ACHDFA
28	ACDHFA

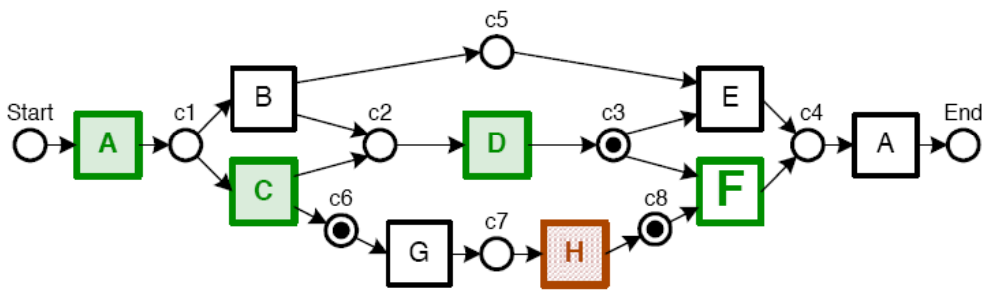
(d)



m = 1
r = 0
c = 3
p = 5

No. of Instances	Log Traces
1207	ABDEA
145	ACDGHFA
56	ACGDHFA
23	→ ACHDFA
28	ACDHFA

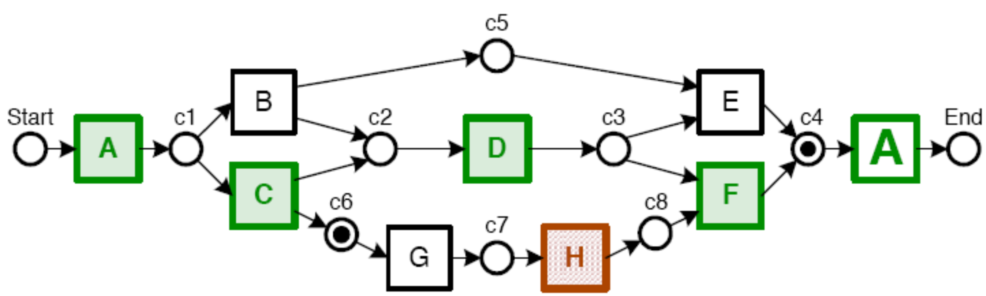
(e)



m = 1
r = 0
c = 4
p = 6

No. of Instances	Log Traces
1207	ABDEA
145	ACDGHFA
56	ACGDHFA
23	→ ACHDFA
28	ACDHFA

(f)

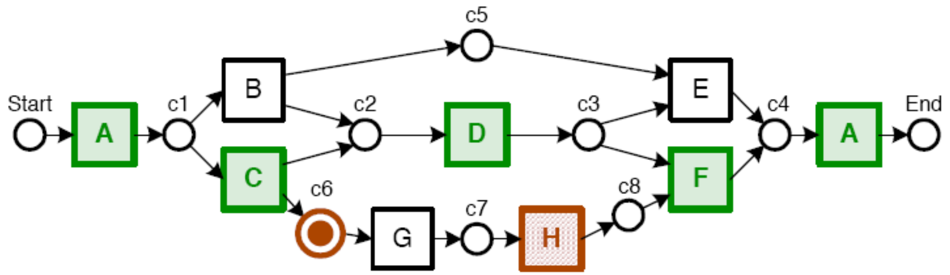


m = 1
r = 0
c = 6
p = 7

No. of Instances	Log Traces
1207	ABDEA
145	ACDGHFA
56	ACGDHFA
23	→ ACHDFA
28	ACDHFA

Fitness calculation

(g)



m = 1
r = 1
c = 8
p = 8

No. of Instances	Log Traces
1207	ABDEA
145	ACDGHFA
56	ACGDHFA
23	ACHDFA
28	ACDHFA

$$f = \frac{1}{2}\left(1 - \frac{\sum_{i=1}^{k} n_i m_i}{\sum_{i=1}^{k} n_i c_i}\right) + \frac{1}{2}\left(1 - \frac{\sum_{i=1}^{k} n_i r_i}{\sum_{i=1}^{k} n_i p_i}\right)$$

$$f(M1, L2) = \frac{1}{2}\left(1 - \frac{51}{10666}\right) + \frac{1}{2}\left(1 - \frac{51}{10666}\right) \approx 0.995$$

Examples

No. of Instances	Log Traces
4070 245 56	ABDEA ACDGHFA ACGDHFA

(a) Event Log L1

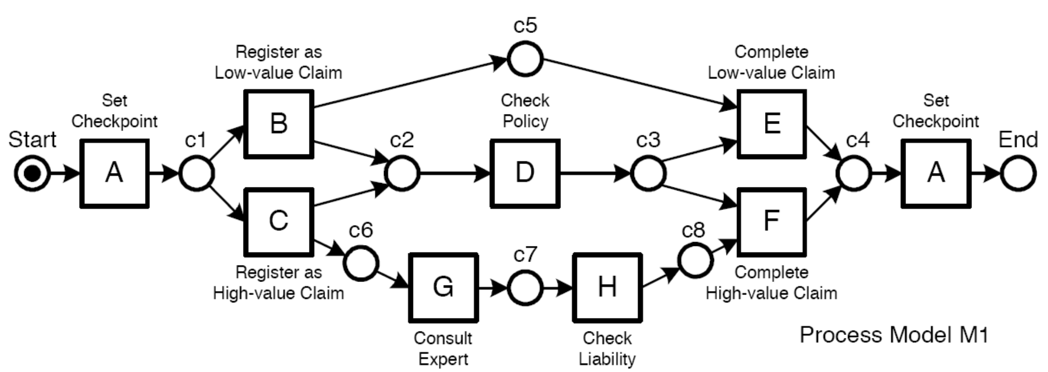
No. of Instances	Log Traces
1207 145 56 23 28	ABDEA ACDGHFA ACGDHFA ACHDFA ACDHFA

(b) Event Log L2

No. of Instances	Log Traces
24 7 15 6 1 8	BDE AABHF CHF ADBE ACBGDFAA ABEDA

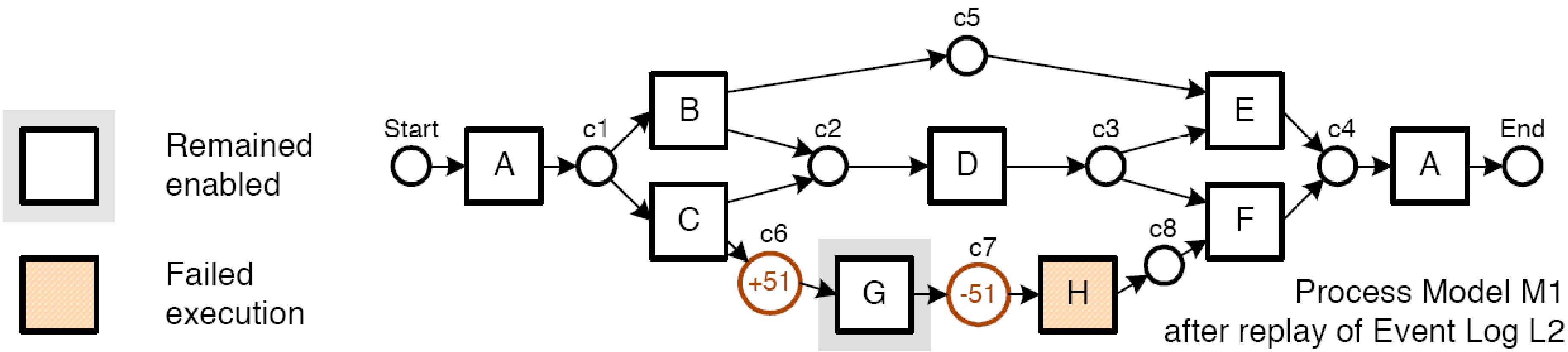
(c) Event Log L3

f=1.000 **f=0.995** **f=0.540**



Process Model M1

Diagnostics

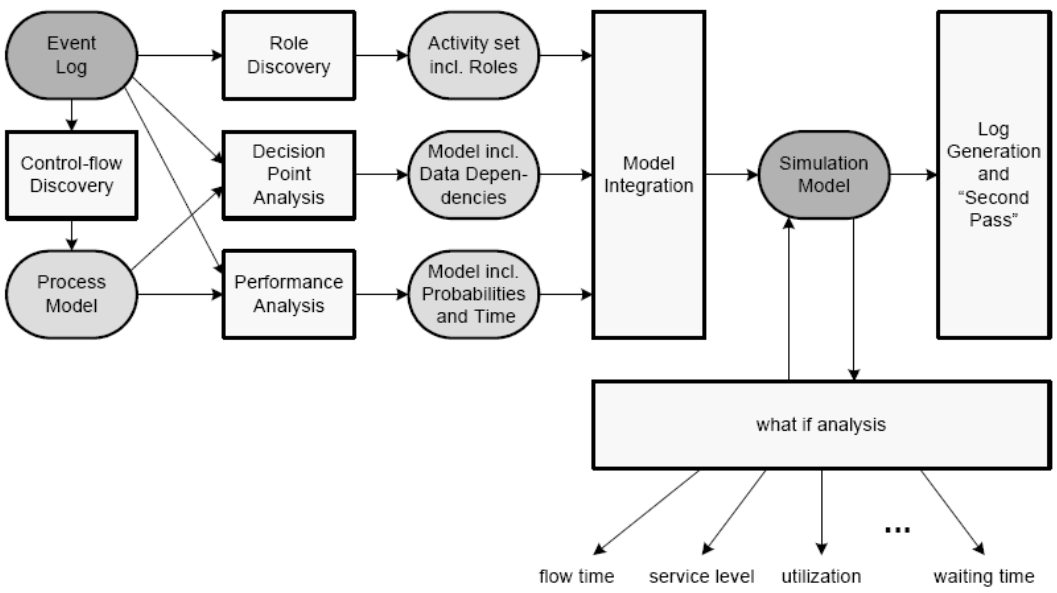


Other Metrics

- **Fitness is not sufficient: hence other metrics are needed such as behavioral and structural appropriateness, etc.**
- **These metrics cover aspects such as:**
  - **Punishing for "too much" behavior.**
  - **Punishing for "overly complex" models.**

Extension

- Existing models can be enriched by logs analysis (e.g., indicating bottlenecks, etc.).
- Process mining results can be combined.
- Can be used to create comprehensive simulation models and export them to e.g. CPN Tools:



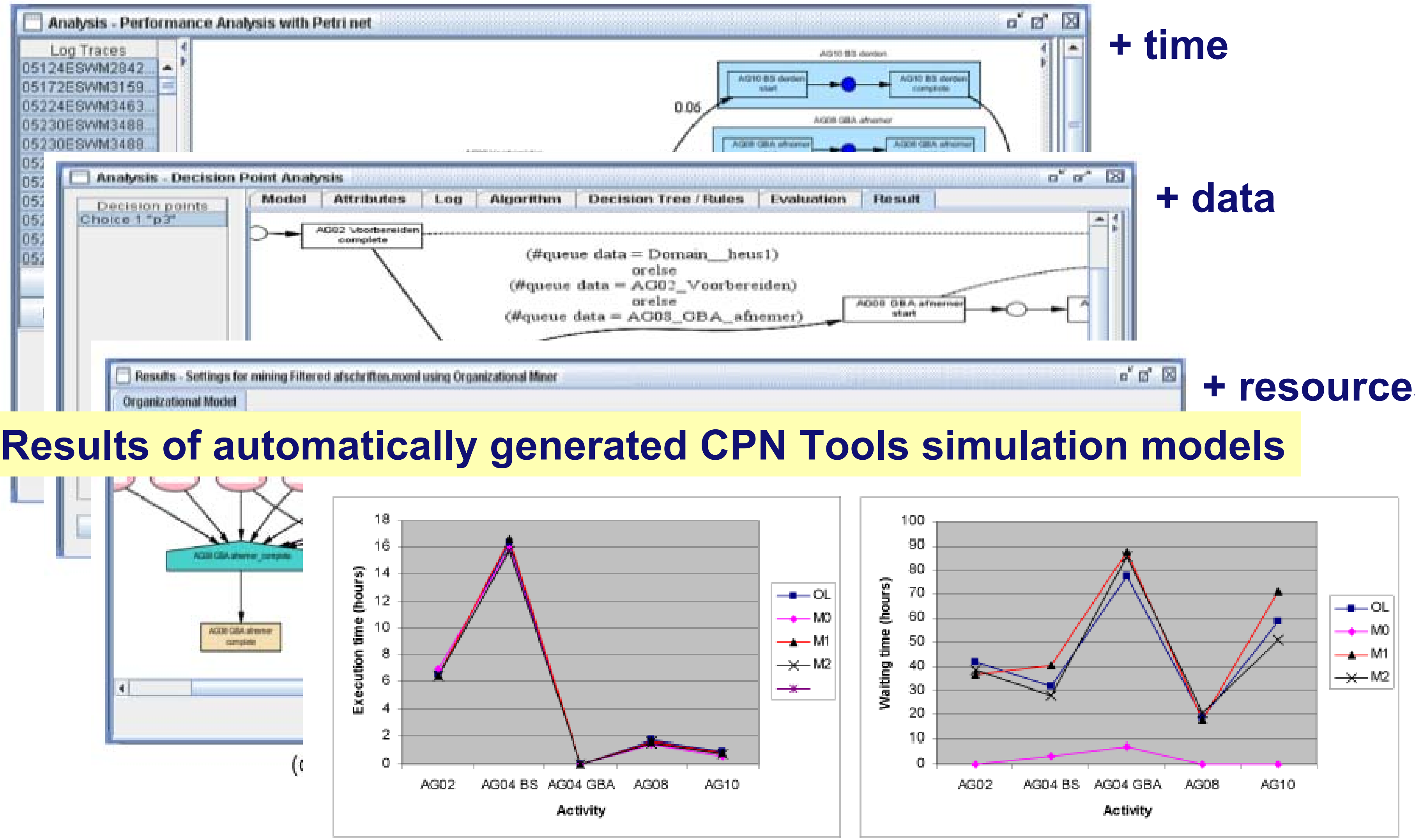
Example: Log from Dutch municipality

+ time

+ data

+ resources

Results of automatically generated CPN Tools simulation models



(a) execution time

(b) waiting time

Part III

Hands-on with ProM

www.processmining.org

ProM

Technische Universiteit **Eindhoven** University of Technology

Where innovation starts

Download/install ProM 5.2

Exercise (4)

- Consider the following log:
  - a b c d f
  - a c b d f
  - a b d c f
  - a c d b f
  - a d e f
  - a e d f
- Use the Alpha algorithm/ProM to discover the corresponding process model (Filename: exercise4.mxml)

Exercise (5)

- **Consider the following log:**
  - **a b c d e f b d c e g**
  - **a b d c e g**
  - **a b c d e f b c d e f b d c e g**
- **Use the Alpha algorithm/ProM to discover the corresponding process model (Filename: exercise5.mxml)**

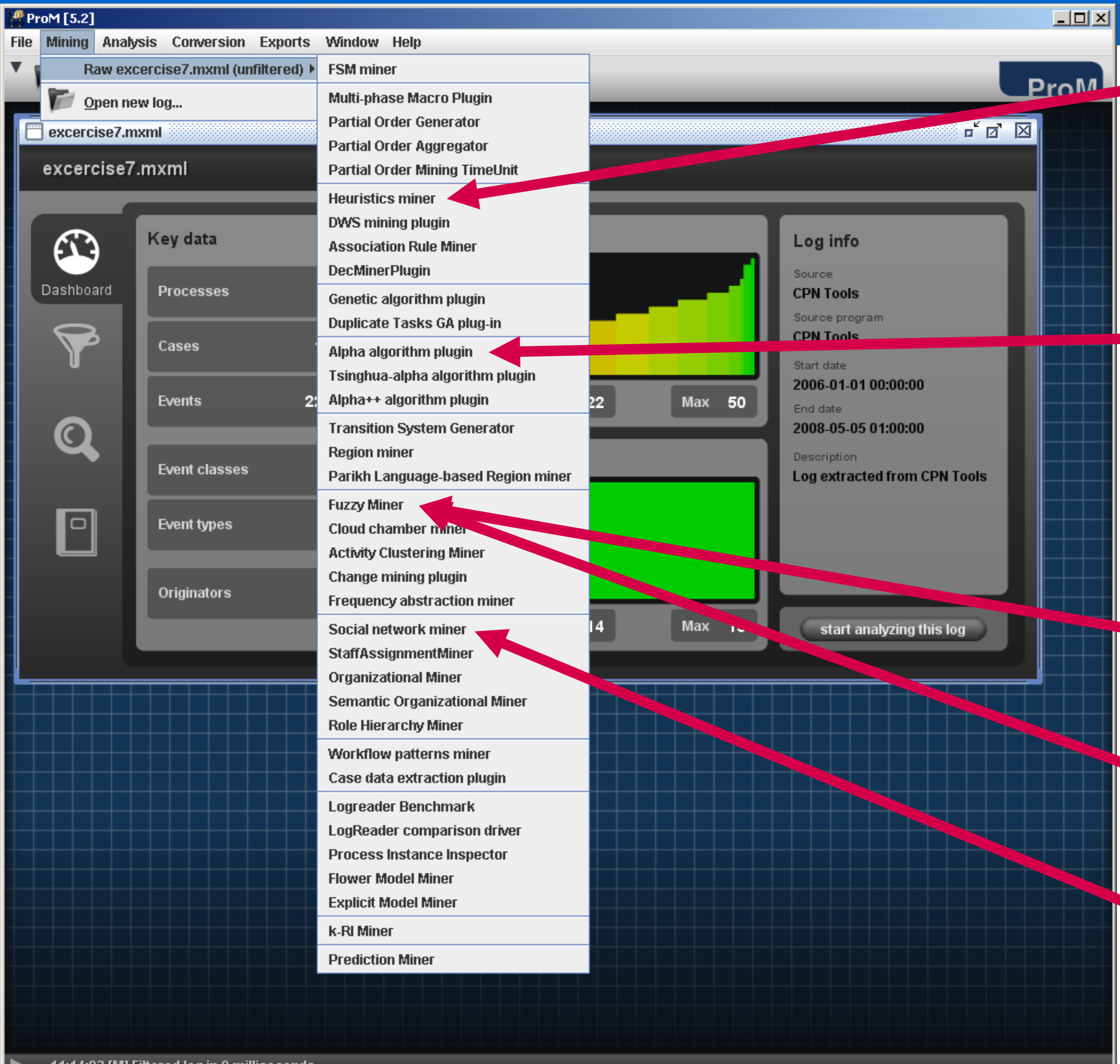
Exercise (6)

- **Consider the following log:**
  - **a b e f**
  - **a b e c d b f**
  - **a b c e d b f**
  - **a b c d e b f**
  - **a e b c d b f**
- **Use the Alpha algorithm/ProM to discover the corresponding process model (Filename: exercise6.mxml)**

Exercise (7)

- Load exercise7.mxml
- Inspect the log (e.g. using the internet explorer)
- Use the Alpha algorithm to discover the corresponding process model.

Explore ProM using exercise7.mxml



[Heuristics miner]
discover process model and play with settings

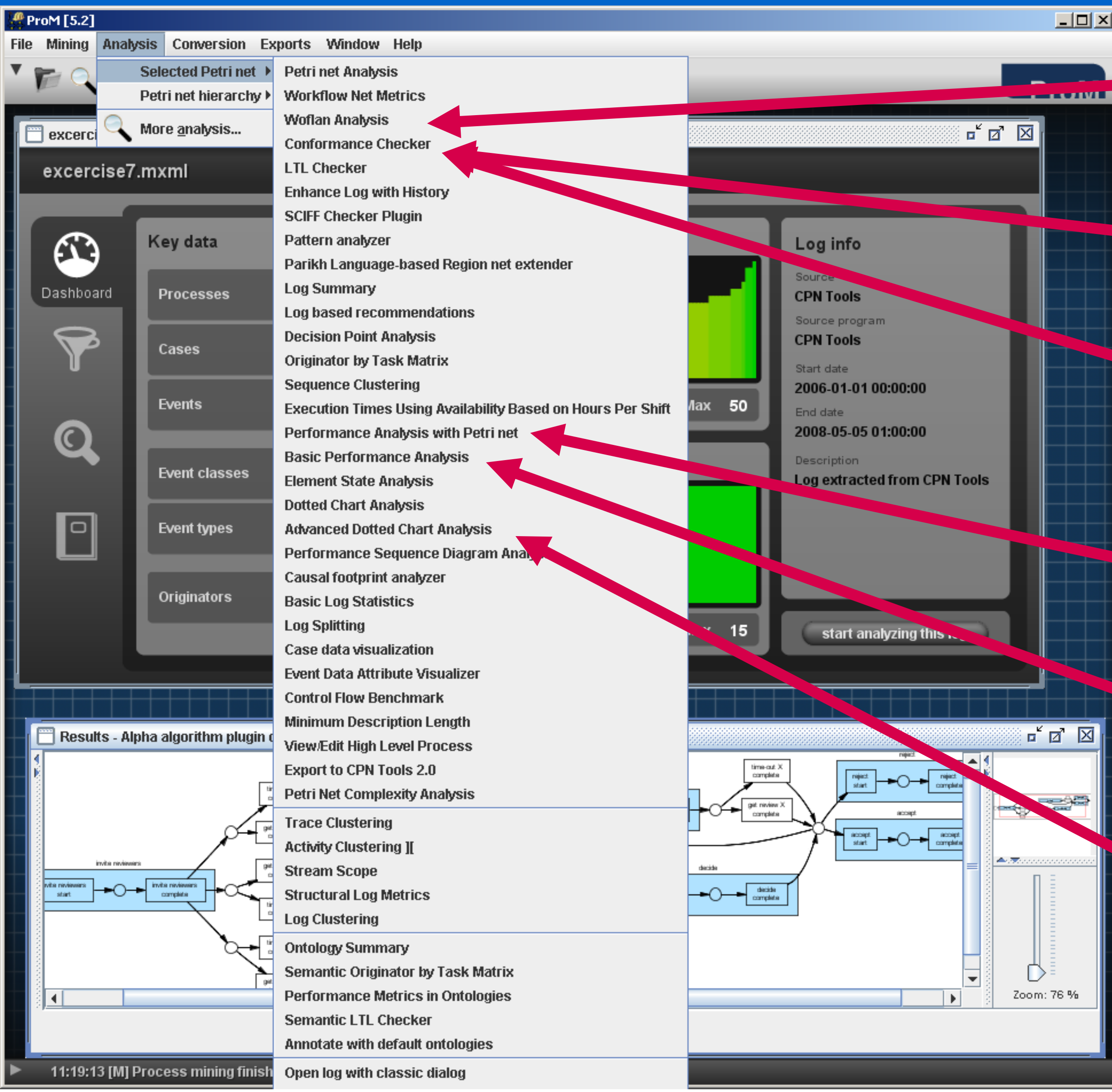
[Alpha algorithm plugin]
discover process model and play with settings

[Fuzzy Miner]
discover process model and explore the various views

[Fuzzy Miner]
animate the model

[Social Network Miner]
discover the social network

Explore ProM using exercise7.mxml (2)



[Woflan Analysis]
verify correctness of model

[Conformance Checker] check the conformance of the mined model

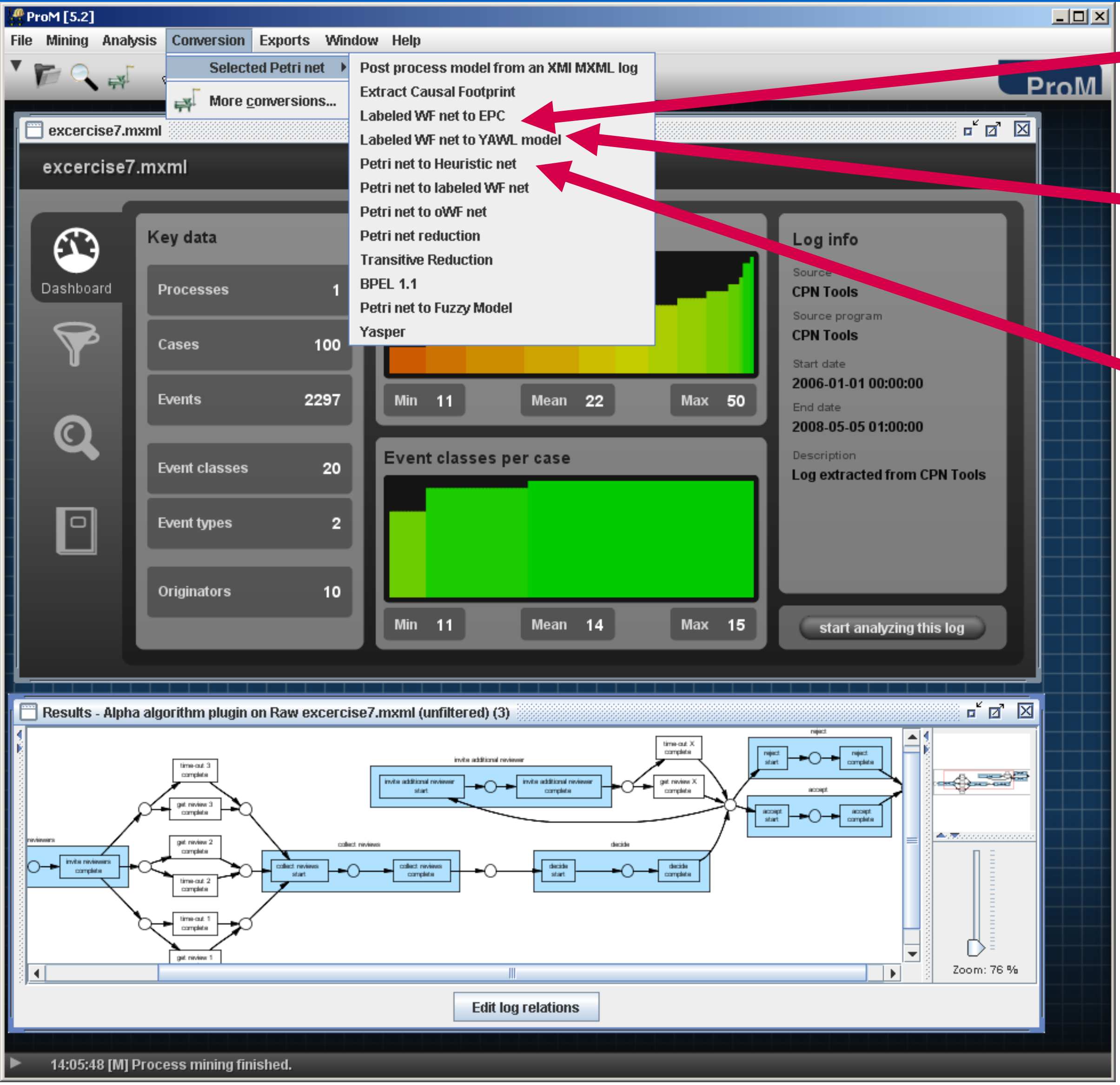
[Conformance Checker] modify the log (delete and insert events) and the check the conformance

[Performance Analysis with Petri net]
analyze the performance (where are the bottlenecks)

[Basic Performance Analysis] analyze the performance (explore all options)

[(Advanced) Dotted Chart Analysis]
construct dotted charts and explore all options

Explore ProM using exercise7.mxml (3)



convert discovered Petri net into EPC model

convert discovered Petri net into YAWL model

convert discovered Petri net into heuristic net

If time is left, ...

- Repeat the process using exercise8.mxml
- Repeat the process using repairexample.zip and repairexamplesample2.zip
- Use the above two files to follow the steps described in the ProM Framework Tutorial

Conclusion

Technische Universiteit
Eindhoven
University of Technology

Where innovation starts

Conclusion

- The abundance of event data enables a wide variety of process mining techniques ranging from process discovery to conformance checking.
- A reality check for people that are involved in process modeling.
- Great application possibilities!
- Good research/PhD topic!
- TomTom functionality is already possible today!
- Check out ProM with its 250+ plug-ins

Thanks! cf. *www.processmining.org*

- 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
- Nikola Trcka
- 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
- Kenny van Uden
- ...

Relevant WWW sites

http://www.senternovem.nl/innovatievouchers
MKB 2.500 – 7.500 euro

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

Technische Universiteit
Eindhoven
University of Technology