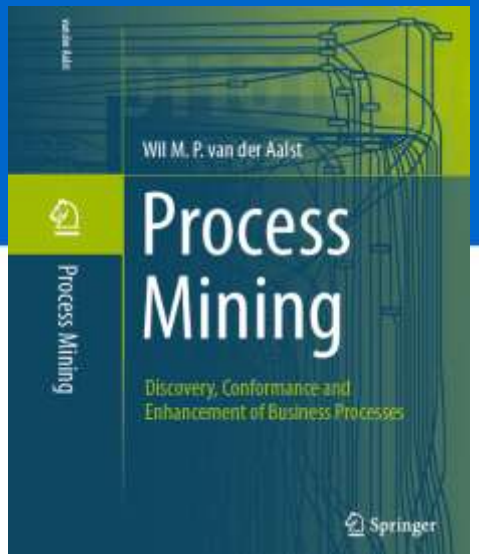
# Process Cubes

## Slicing, Dicing, Rolling Up and Drilling Down Event Data for Process Mining

**prof.dr.ir. Wil van der Aalst**
**www.processmining.org**




TU/e Technische Universiteit Eindhoven University of Technology

Where innovation starts


小草对您微微笑
请您把路绕一绕
KEEP OFF GRASS
绿色大学办公室
修缮中心园林科


小草对您微微笑
请您把路绕一绕
KEEP OFF GRASS
绿色大学办公室
绿缘中心园林科

# Positioning Process Mining

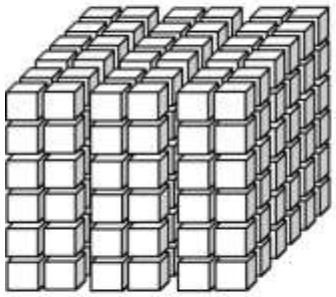
**process model analysis**

(simulation, verification, optimization, gaming, etc.)

**performance-oriented** questions, problems and solutions

process mining

**compliance-oriented** questions, problems and solutions



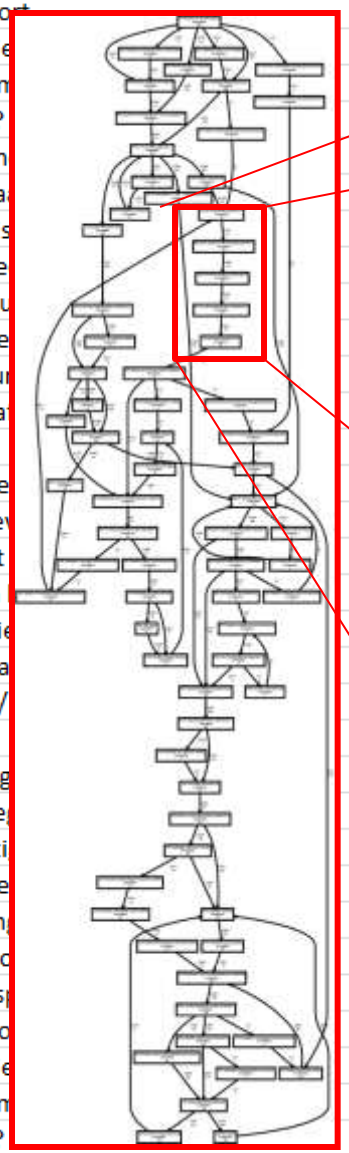
**data-oriented analysis**

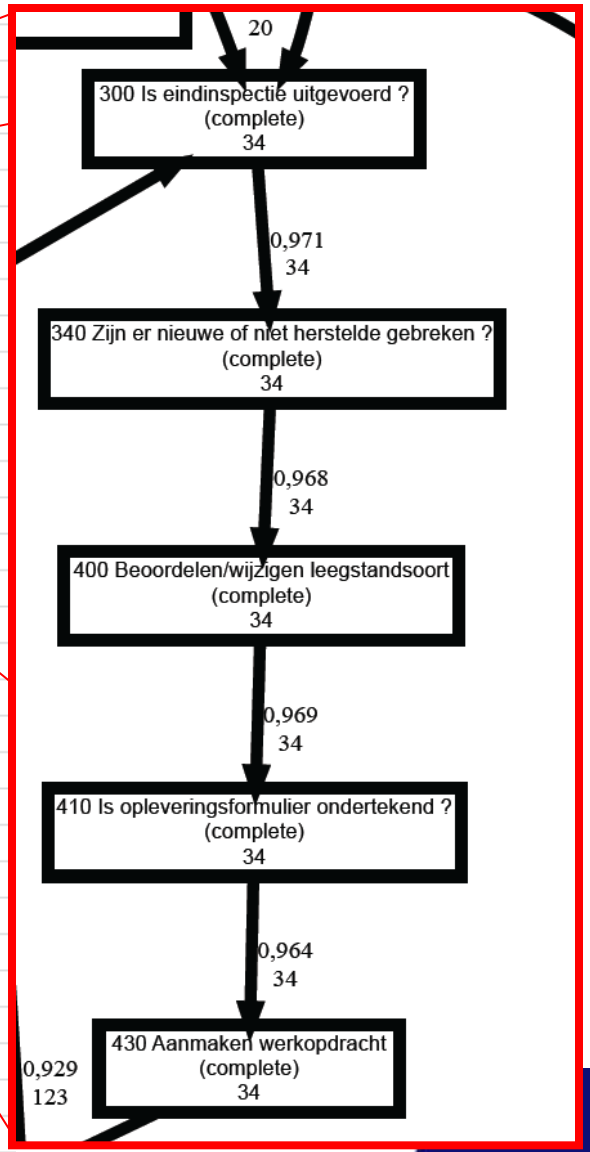
(data mining, machine learning, business intelligence)

# Example Process Discovery

## (Vestia, Dutch housing agency, 208 cases, 5987 events)

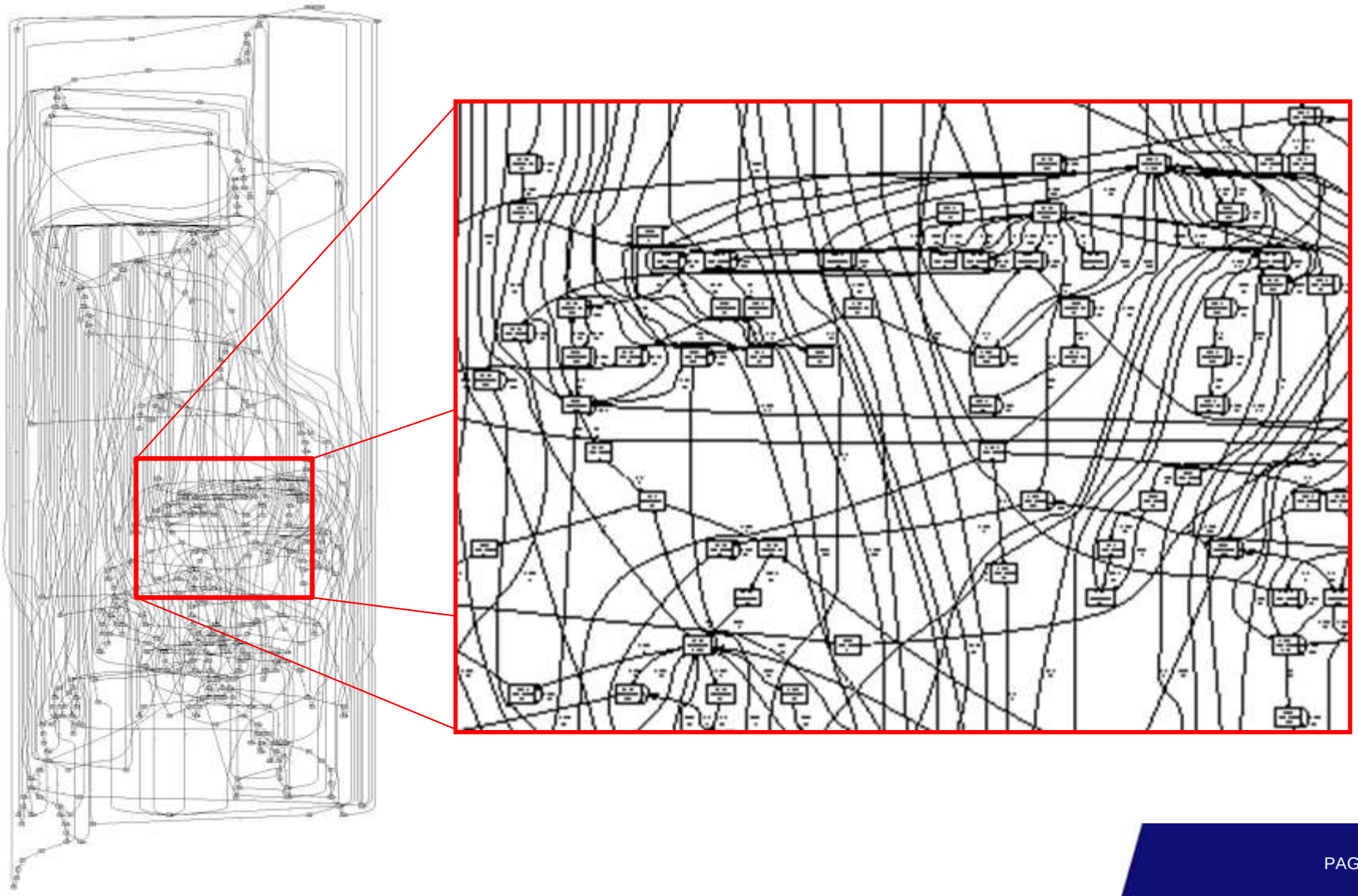
| | | |
|---|---|---|
| 117315 | 110 Bepalen leegstandsoo | 16.05.2007 14:06:23 |
| 117315 | 120 Plannen eindinspectie | 16.05.2007 14:36:01 |
| 117315 | 130 Is het opleveringsform | 23.05.2007 09:41:40 |
| 117315 | 150 Is er sprake van ZAV ? | 23.05.2007 09:41:51 |
| 117315 | 170 Aanpassen plattegron | 23.05.2007 11:57:18 |
| 117315 | 180 Aanpassen woningwa | 23.05.2007 09:42:37 |
| 117315 | 190 Actualiseren huurprijs | 23.05.2007 09:48:23 |
| 117315 | 200 Toewijzen woning/be | 23.05.2007 09:48:29 |
| 117315 | 210 Registreren voorl. huu | 10.09.2007 16:24:36 |
| 117315 | 220 Is contract getekend e | 11.09.2007 14:56:18 |
| 117315 | 240 Definitief maken Huur | 31.03.2008 16:17:12 |
| 117315 | 250 Aanpassen factureera | 09.09.2008 15:39:59 |
| 117315 | 260 After sales | 09.09.2008 16:51:24 |
| 117315 | 270 Archiveren nieuwe ve | 10.09.2008 07:52:08 |
| 117315 | 300 Is eindinspectie uitgev | 07.06.2007 14:47:04 |
| 117315 | 340 Zijn er nieuwe of niet | 07.06.2007 14:47:06 |
| 117315 | 400 Beoordelen/wijzigen l | 07.06.2007 14:51:16 |
| 117315 | 410 Is opleveringsformulie | 07.06.2007 14:51:26 |
| 117315 | 430 Aanmaken werkopdra | 11.06.2007 09:21:39 |
| 117315 | 440 Worden er bonussen/ | 11.06.2007 09:21:49 |
| 117315 | 460 Opstellen eindnota | 08.08.2007 16:18:26 |
| 117315 | 470 Archiveren huuropzeg | 09.08.2007 14:42:23 |
| 119763 | 010 Registreren huuropze | 09.05.2007 11:19:14 |
| 119763 | 030 Vastleggen toekomsti | 09.05.2007 12:25:01 |
| 119763 | 050 Inplannen afspraak 1e | 09.05.2007 11:59:52 |
| 119763 | 060 Aanmaken bevestigin | 09.05.2007 12:31:57 |
| 119763 | 070 Is 1e inspectie uitgevo | 16.05.2007 13:04:26 |
| 119763 | 100 Gereedmelden 1e insp | 16.05.2007 13:43:39 |
| 119763 | 110 Bepalen leegstandsoo | 16.05.2007 13:43:28 |
| 119763 | 120 Plannen eindinspectie | 16.05.2007 13:42:58 |
| 119763 | 130 Is het opleveringsform | 16.05.2007 13:34:49 |
| 119763 | 150 Is er sprake van ZAV ? | 16.05.2007 13:34:56 |





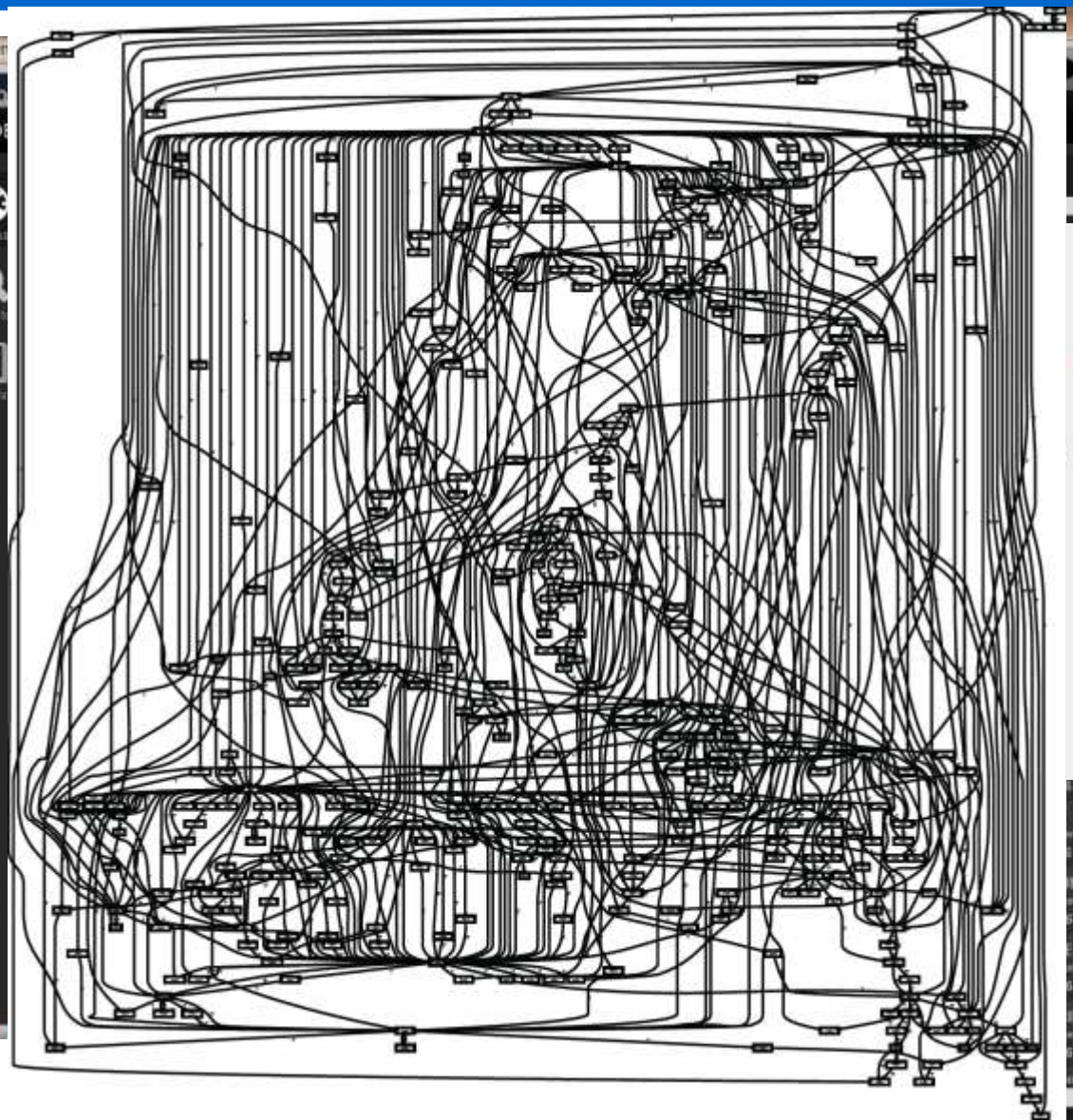
# Example Process Discovery

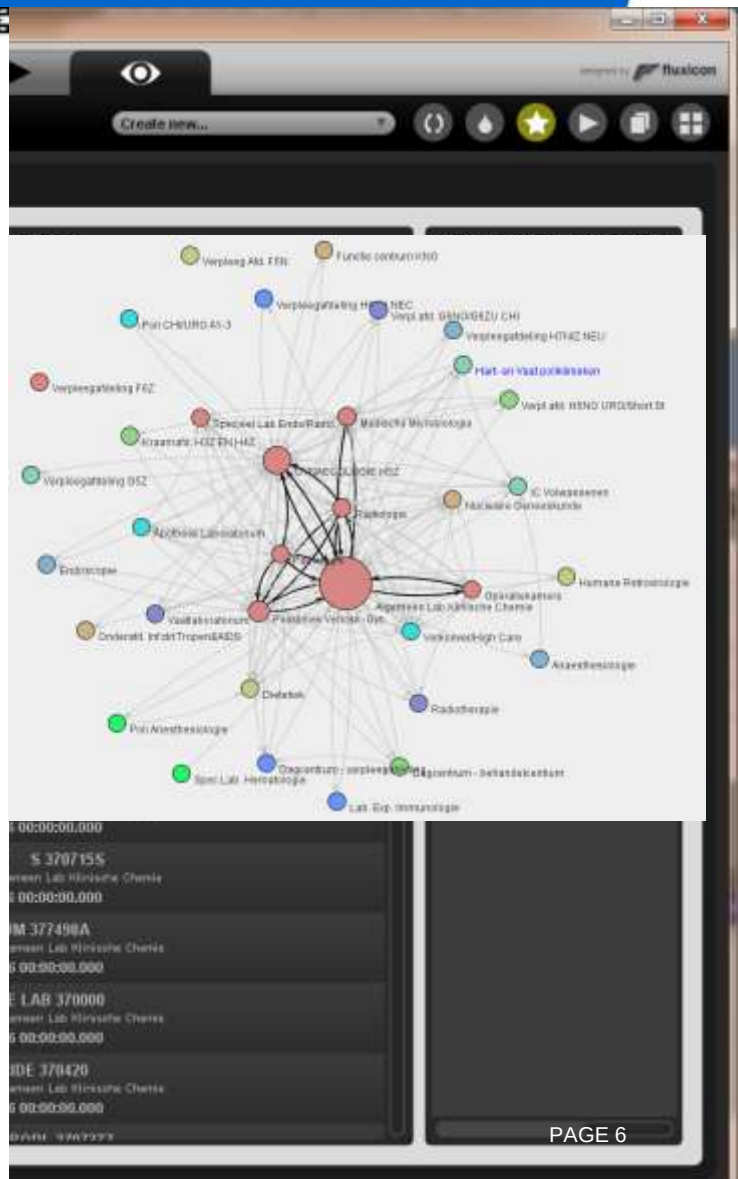
## (ASML, test process lithography systems, 154966 events)



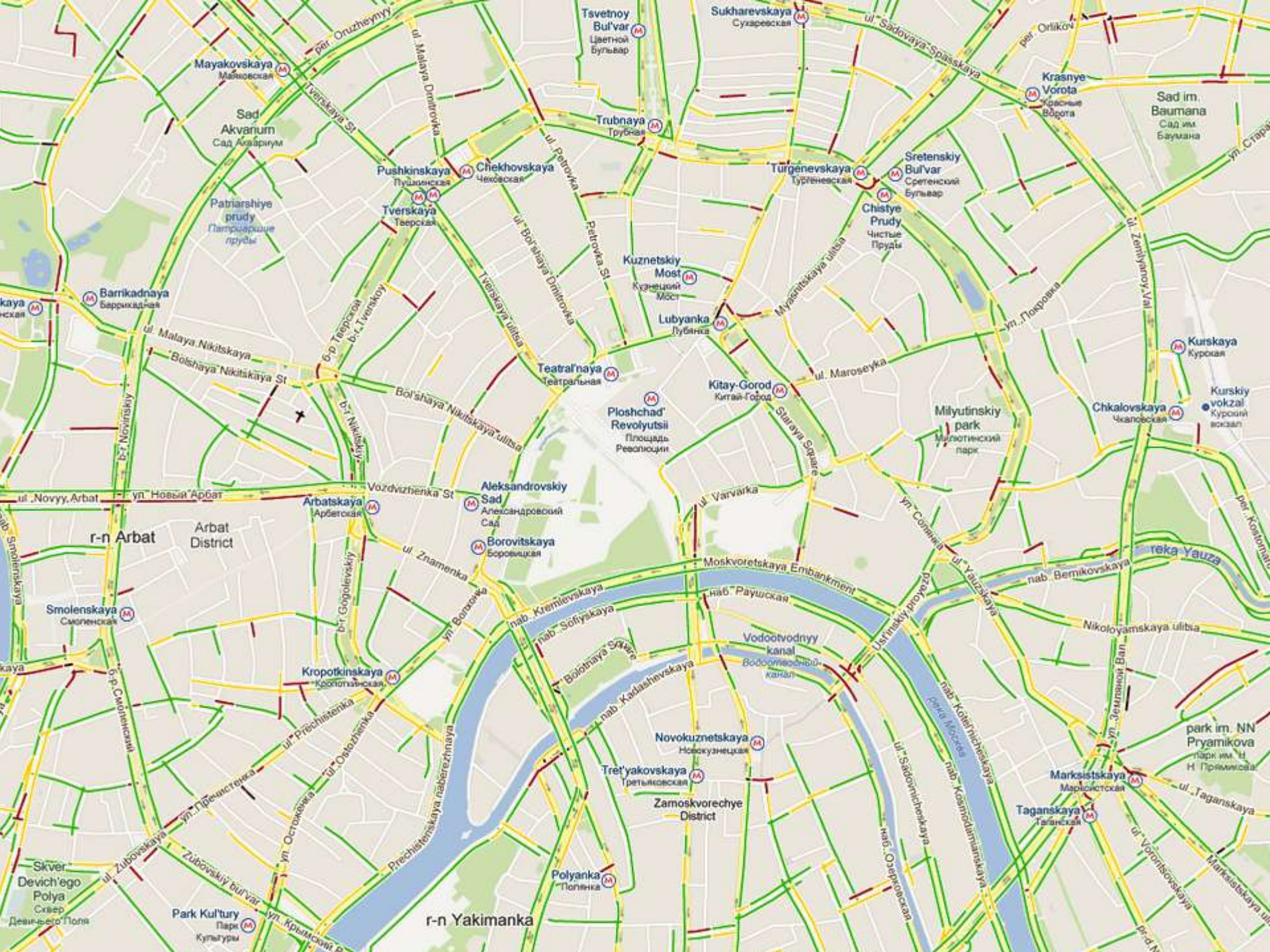
# Example Process Discovery

## (AMC, 627 gynecological oncology patients, 24331 events)

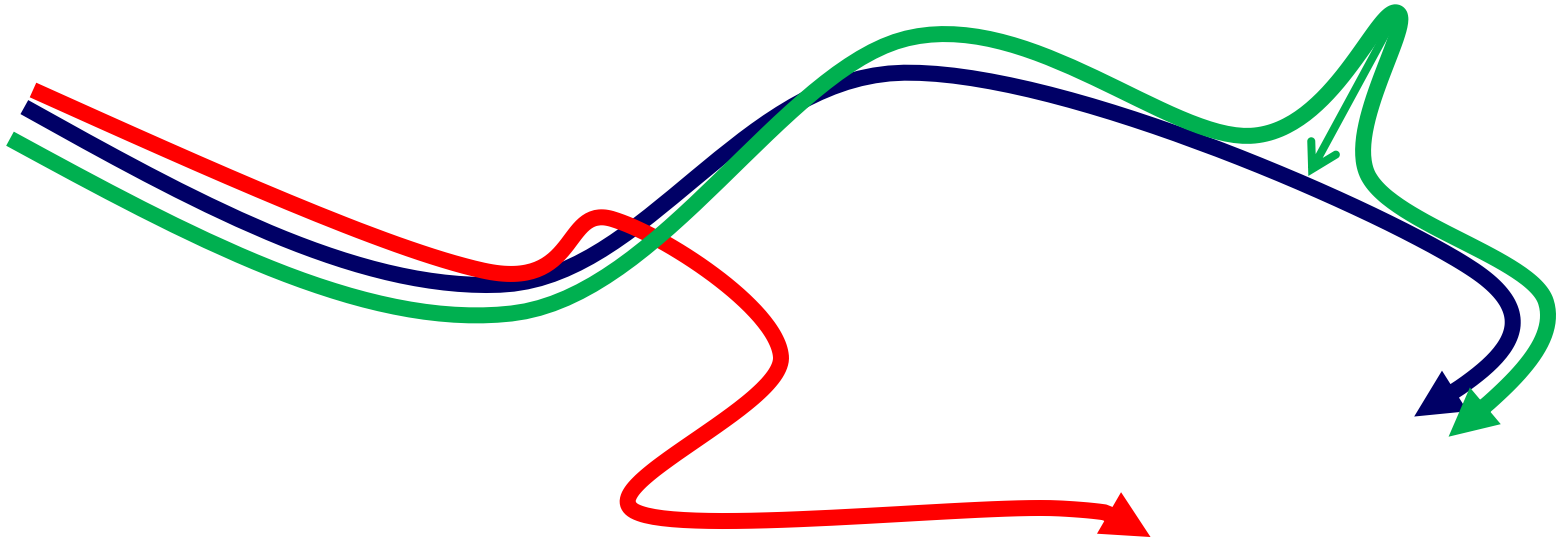




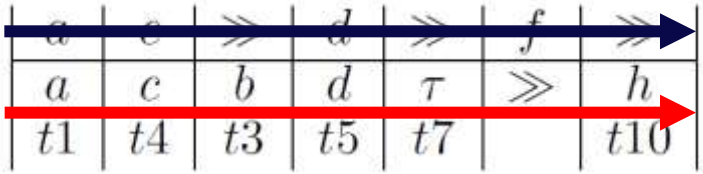
# Models are like the glasses required to see and understand event data!

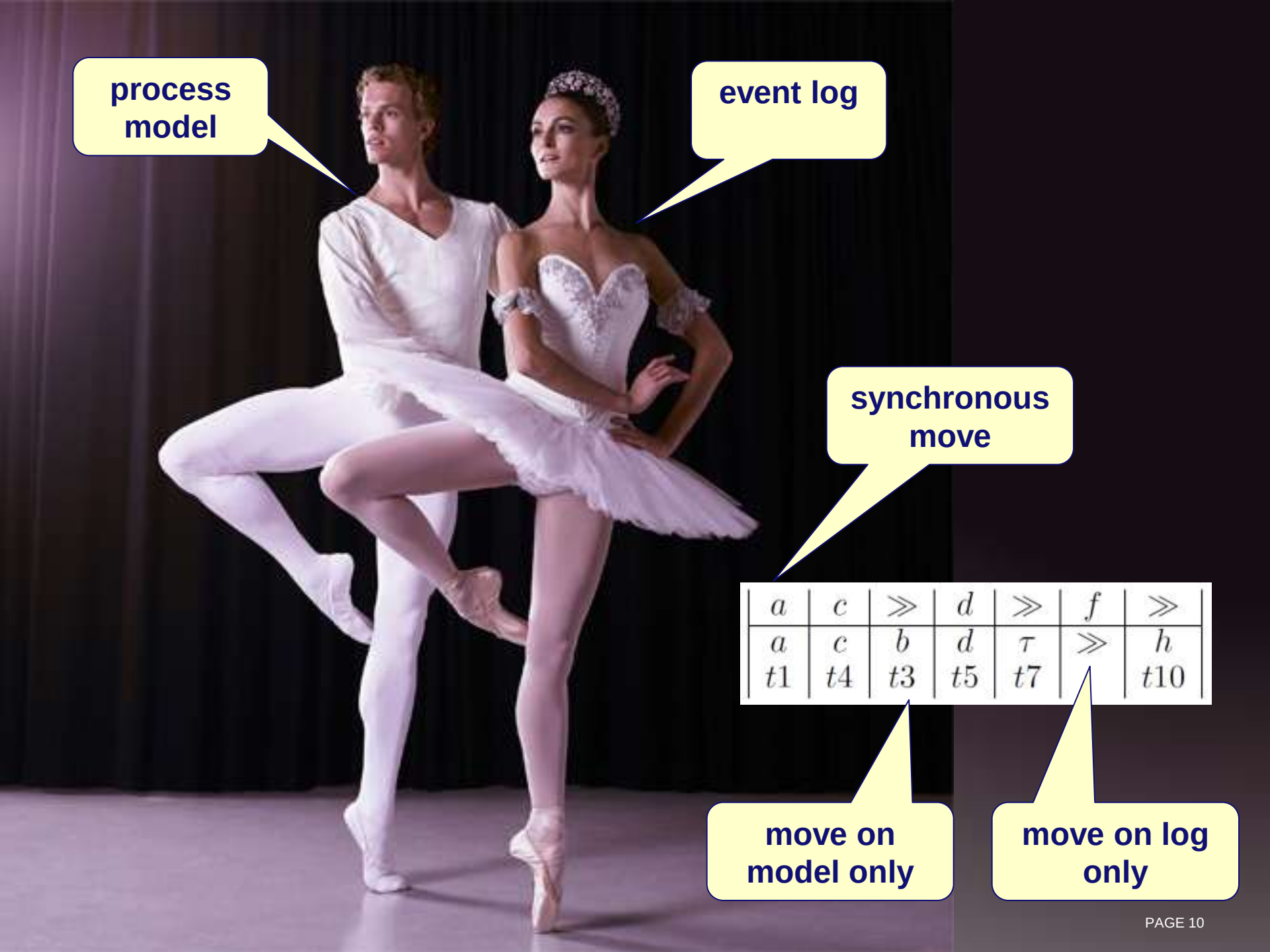

Tsvetnoy Bul'var
Цветной бульвар
Sukharevskaya
Сухаревская
ul. Sadovaya-Spasskaya
per. Orlikov
Mayakovskaya
Маяковская
per. Oruzheynyy
ul. Malaya Dmitrovka
Tverskaya St
Sad Akvarium
Сад Аквариум
Trubnaya
Трубная
Krasnye Vorota
Красные Ворота
Sad im. Baumana
Сад им Баумана
Pushkinskaya
Пушкинская
Chekhovskaya
Чеховская
ul. Petrovka
Turgenevskaya
Тургеневская
Sretenskiy Bul'var
Сретенский бульвар
Patriarshiye prudy
Патриаршие пруды
Tverskaya
Тверская
Chistye Prudy
Чистые Пруды
ul. Bol'shaya Dmitrovka
Petrovka St
Kuznetskiy Most
Кузнецкий Мост
Myasnitskaya ulitsa
Barrikadnaya
Баррикадная
Lubyanka
Лубянка
ul. Malaya Nikitskaya
Bol'shaya Nikitskaya St
b-r Tverskoy
Teatral'naya
Театральная
Kitay-Gorod
Китай-город
ul. Maroseyka
Kurskaya
Курская
Ploshchad' Revolyutsii
Площадь Революции
Milyutinskiy park
Милютинский парк
Chkalovskaya
Чкаловская
Kurskiy vokzal
Курский вокзал
b-r Novinskiy
b-r Nikitskiy
ul. Novyy Arbat
ул. Новый Арбат
Vozdvizhenka St
Aleksandrovskiy Sad
Александровский Сад
Arbatskaya
Арбатская
ul. Varvarka
Staraya Square
r-n Arbat
Arbat District
Borovitskaya
Боровицкая
ul. Znamenka
Moskvoretskaya Embankment
nab. Raushskaya
reka Yauza
nab. Bernikovskaya
Smolenskaya
Смоленская
nab. Kremlevskaya
nab. Sofiyskaya
Vodootvodnyy kanal
Водоотводный канал
Nikoloyamskaya ulitsa
Bolotnaya Square
Kropotkinskaya
Кропоткинская
nab. Kadashevskaya
Ust'inskiy proyezd
ul. Prechistenka
ul. Ostozhenka
Novokuznetskaya
Новокузнецкая
park im. NN Pryamikova
Tret'yakovskaya
Третьяковская
Zamoskvorechye District
Marksistskaya
Марксистская
Taganskaya
Таганская
Skver Devich'ego Polya
Сквер Девичьего Поля
Zubovskiy bul'var
Prechistenskaya naberezhnaya
Polyanka
Поляна
Park Kul'tury
Парк Культуры
r-n Yakimanka

# Alignments are essential!



- conformance checking to diagnose deviations
- squeezing reality into the model to do model-based analysis





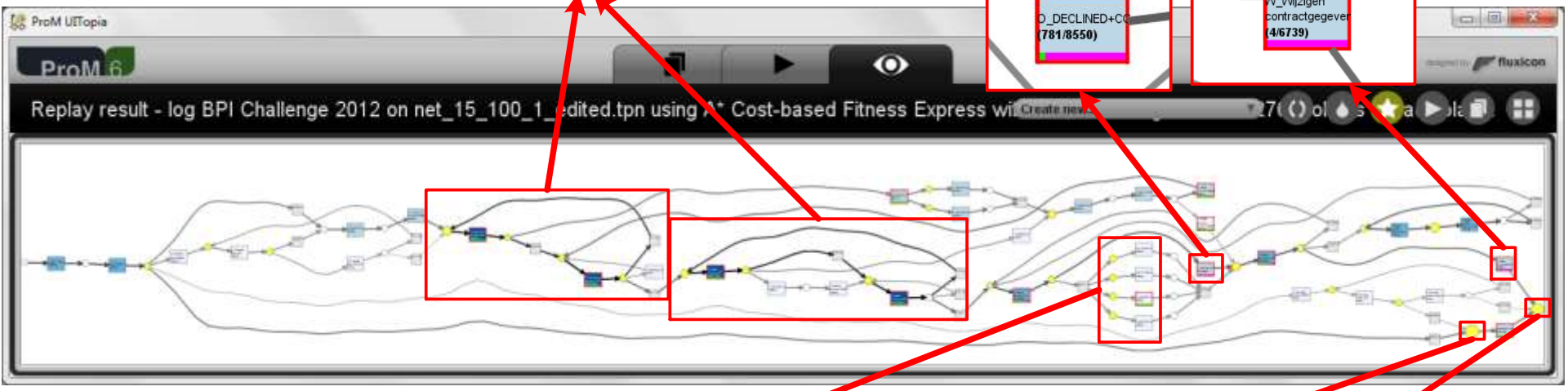

| $a$ | $c$ | $\gg$ | $d$ | $\gg$ | $f$ | $\gg$ |
|---|---|---|---|---|---|---|
| $a$ | $c$ | $b$ | $d$ | $\tau$ | $\gg$ | $h$ |
| $t1$ | $t4$ | $t3$ | $t5$ | $t7$ | | $t10$ |

# Example: BPI Challenge 2012

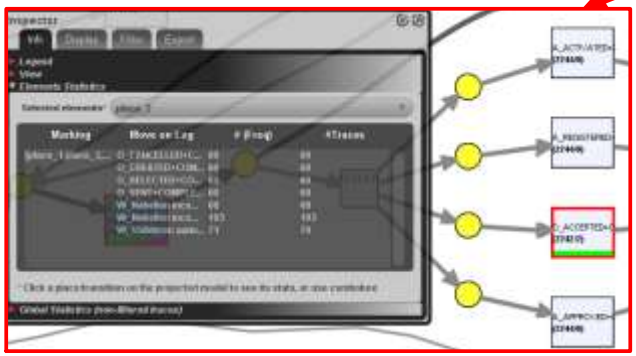
## (Dutch financial institute, doi:10.4121/uuid:3926db30-f712-4394-aebc-75976070e91f)

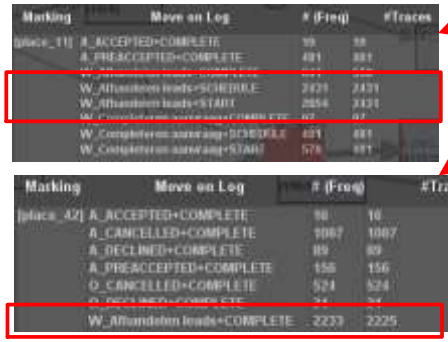
Loops of “W_Completeren aanvraag” and “W_Nabellen offertes” are often performed

“O_DECLINED” and “W_Wijzigen contractgegevens” are often skipped




Many moves on log of “O_CANCELLED”, ”O_CREATED”, ”O_SELECTED”, “O_SENT” occurred with the same frequency value (i.e. 60) before parallel branch



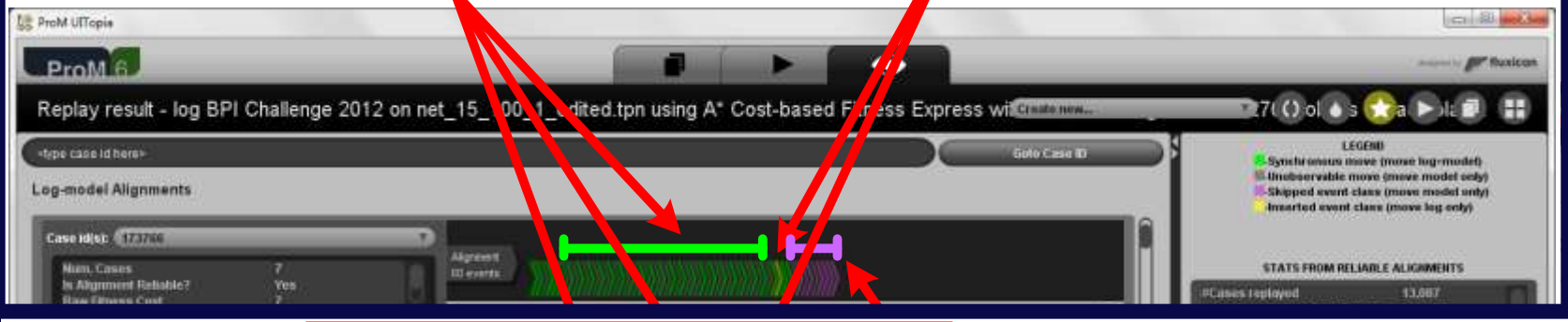



Many moves on log of “W_Afhandelen leads” ( > 2200 times) occurred in the end of traces

**Work of Arya Adriansyah (Replay project)**

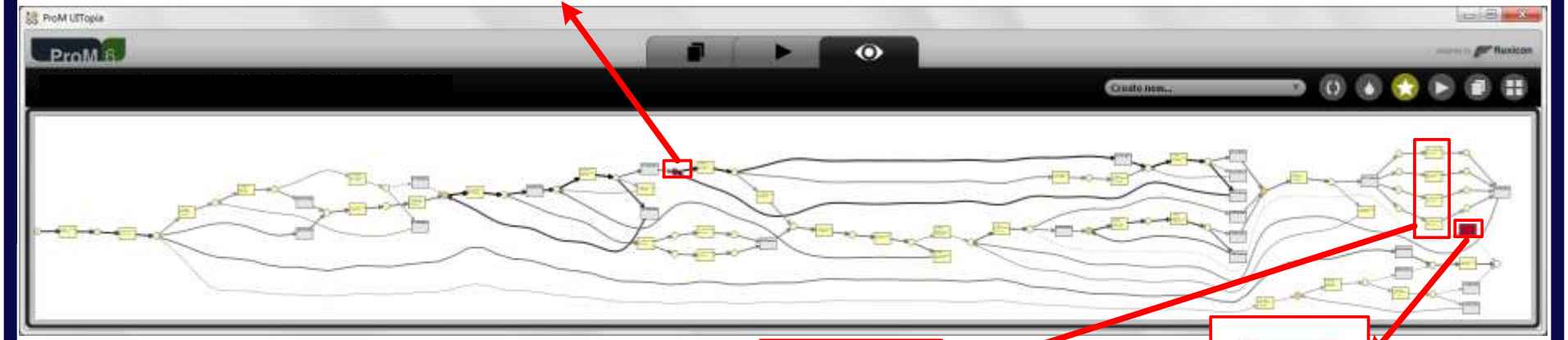
Synchronous moves of "Completeren aanvraag"

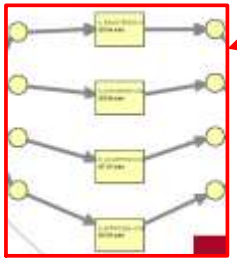
Move on log of "Completeren aanvraag"




| Property | Min. | Max. | Avg. | Std. Dev | Freq. |
|---|---|---|---|---|---|
| Waiting time | 0.00 ms | 29.78 days | 2.83 days | 3.30 days | 24,229 |
| Synchronization time | 0.00 ms | 0.00 ms | 0.00 ms | 0.00 ms | 24,229 |
| Sojourn time | 0.00 ms | 29.78 days | 2.83 days | 3.30 days | 24,229 |

The average waiting time for the input place of "W_Nabellen offertes+START" is very long (2.83 days) compares to the average waiting time of other places



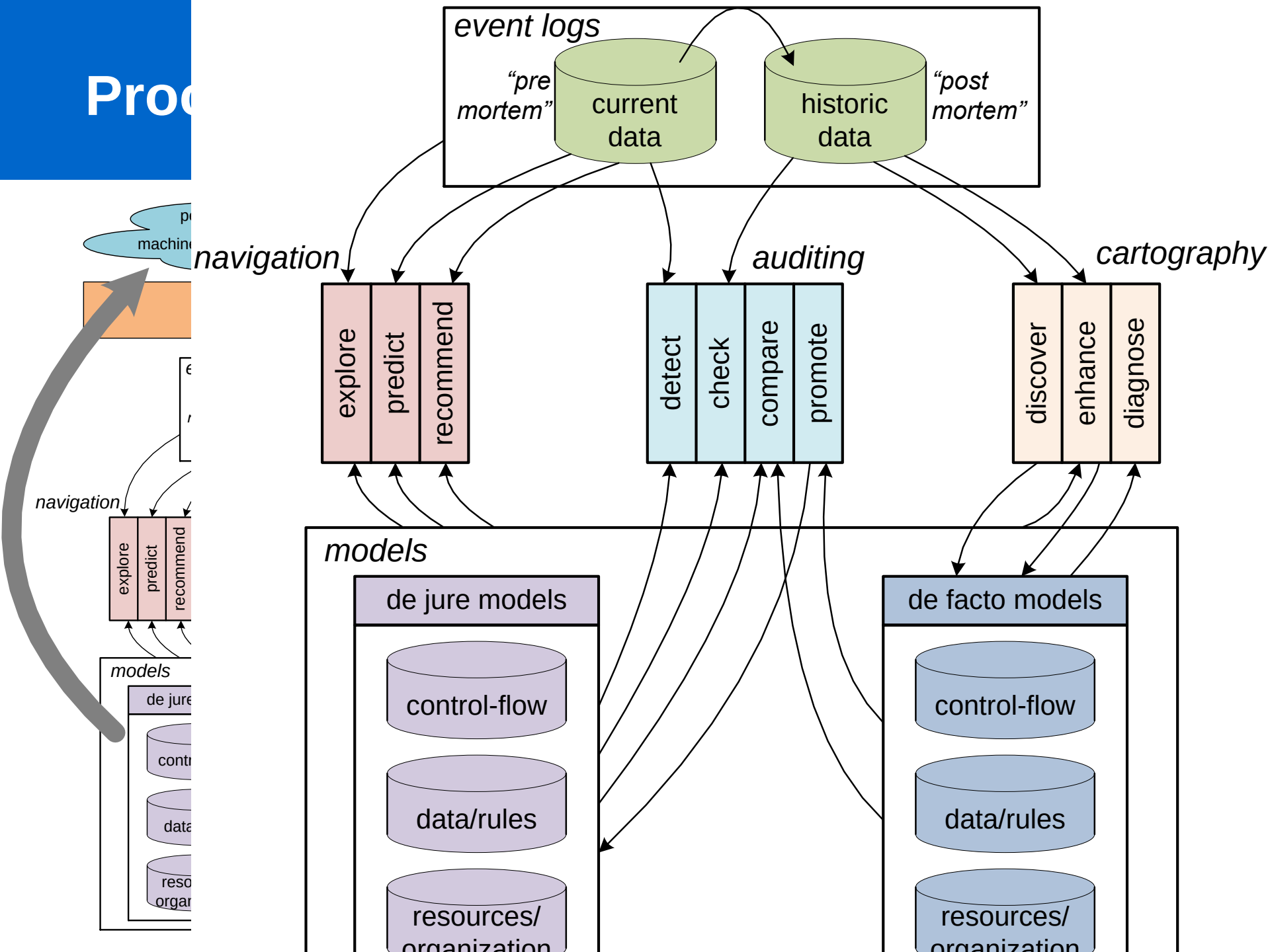



"O_ACCEPTED" has average sojourn time of 27.07 minutes, while "A_REGISTERED", "A_ACTIVATED", and "A_APPROVED" have average sojourn time of 29.56 minutes

| Property | Min. | Max. | Avg. | Std. Dev | Freq. |
|---|---|---|---|---|---|
| Throughput time | 0.00 ms | 0.00 ms | 0.00 ms | 0.00 ms | 4 |
| Waiting time | 1.55 hours | 3.43 months | 1.14 months | 1.55 months | 4 |
| Sojourn time | 1.55 hours | 3.43 months | 1.14 months | 1.55 months | 4 |
| #Unique cases ... | 4 | | | | |

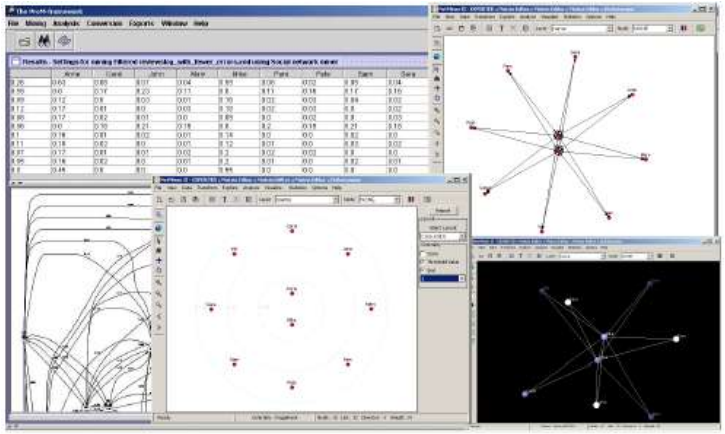
Activity "W_Wijzigen contractgegevens" is the bottleneck, but it occured rarely (only 4 times)

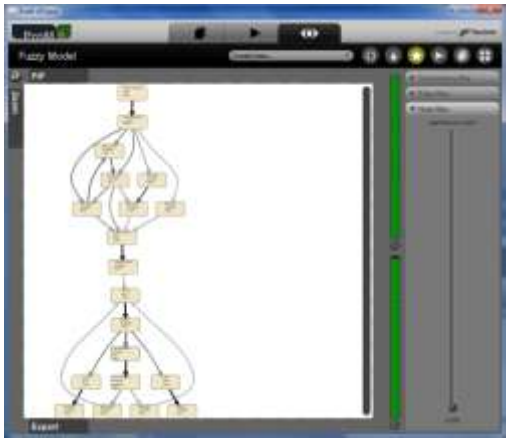
# Proc


event logs
"pre mortem"
current data
historic data
"post mortem"
navigation
auditing
cartography
explore
predict
recommend
detect
check
compare
promote
discover
enhance
diagnose
models
de jure models
de facto models
control-flow
data/rules
resources/ organization
control-flow
data/rules
resources/ organization

# ProM (2004 – now)

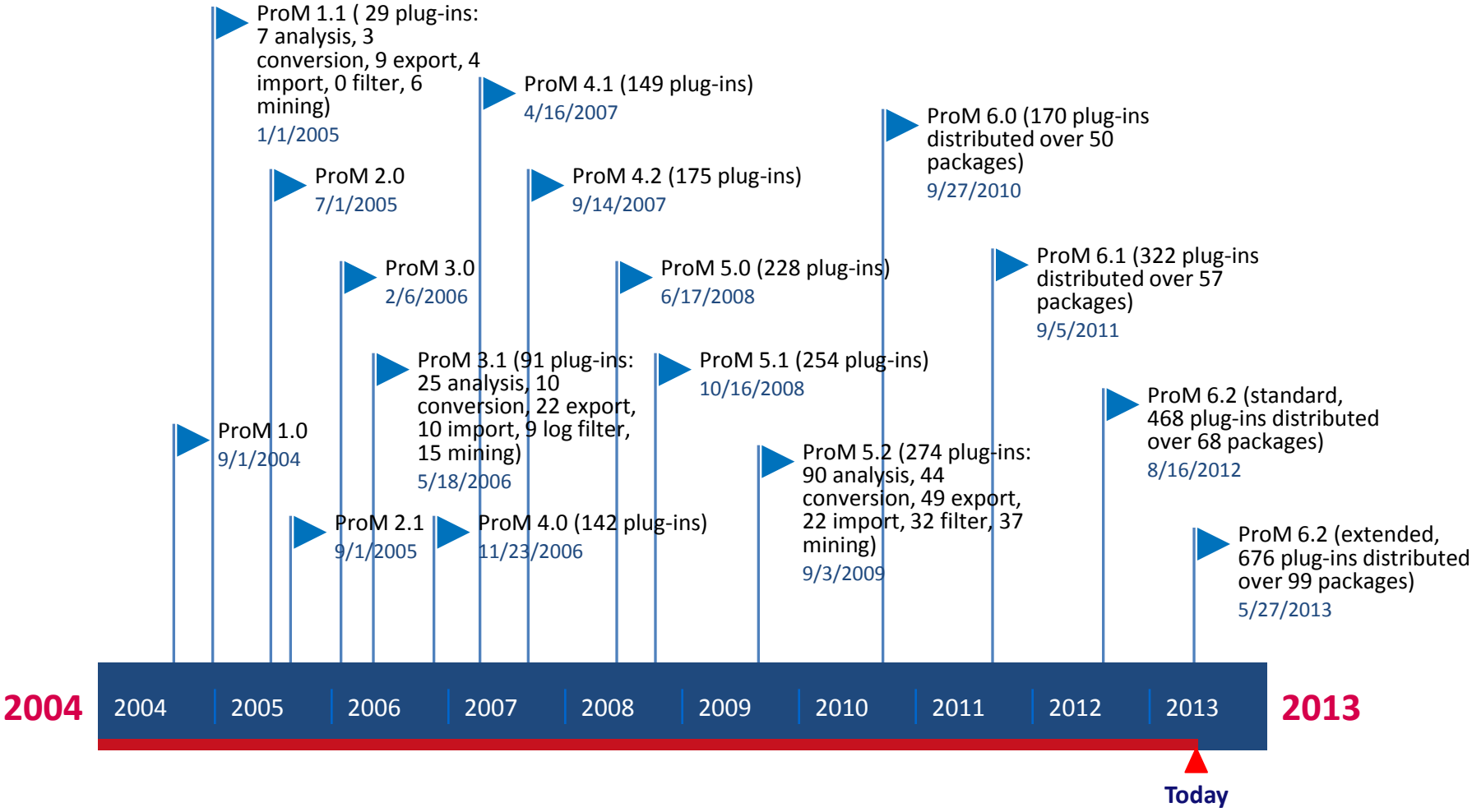
ProM
process mining workbench





**See www.processmining.org**

# Overview of ProM releases



# ProM 1.1

**ProM 1.1 ( 29 plug-ins: 7 analysis, 3 conversion, 9 export, 4 import, 0 filter, 6 mining)**
1/1/2005

ProM 4.1 (149 plug-ins)
4/16/2007

ProM 2.0
7/1/2005

ProM 4.2 (175
9/14/2007

ProM 3.0
2/6/2006

ProM
6/17

ProM 3.1 (91 plug-ins: 25 analysis, 10 conversion, 22 export, 10 import, 9 log filter, 15 mining)
5/18/2006

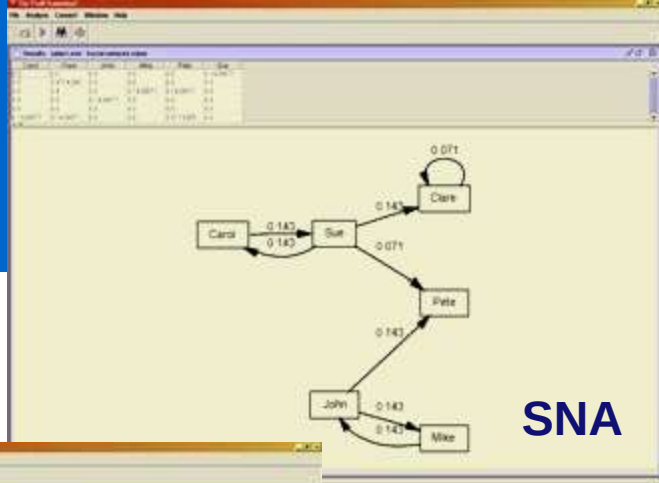
ProM 1.0
9/1/2004

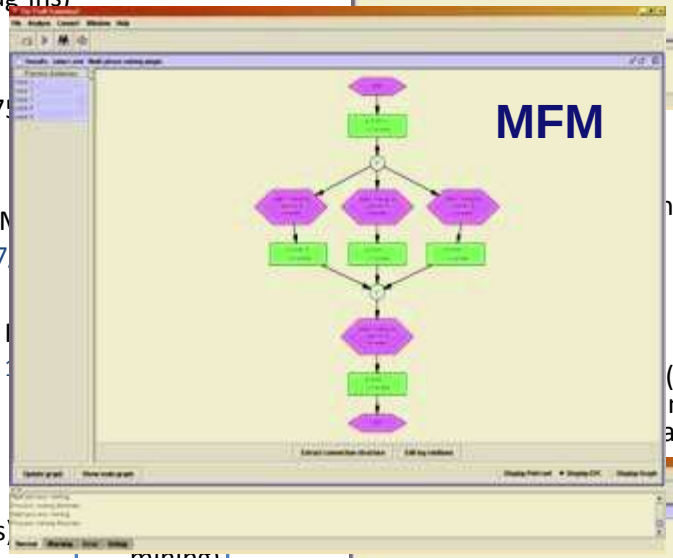
ProM 2.1
9/1/2005

ProM 4.0 (142 plug-ins)
11/23/2006

mining)
9/3/2009

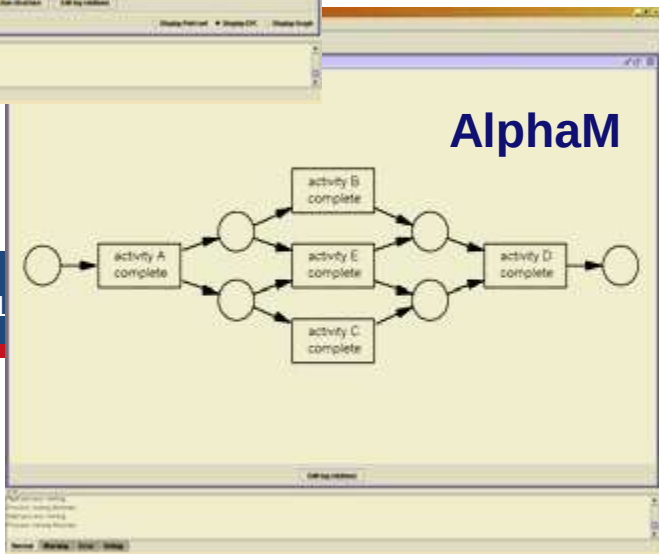
2004 | 2004 | 2005 | 2006 | 2007 | 2008 | 2009 | 2010 | 2011



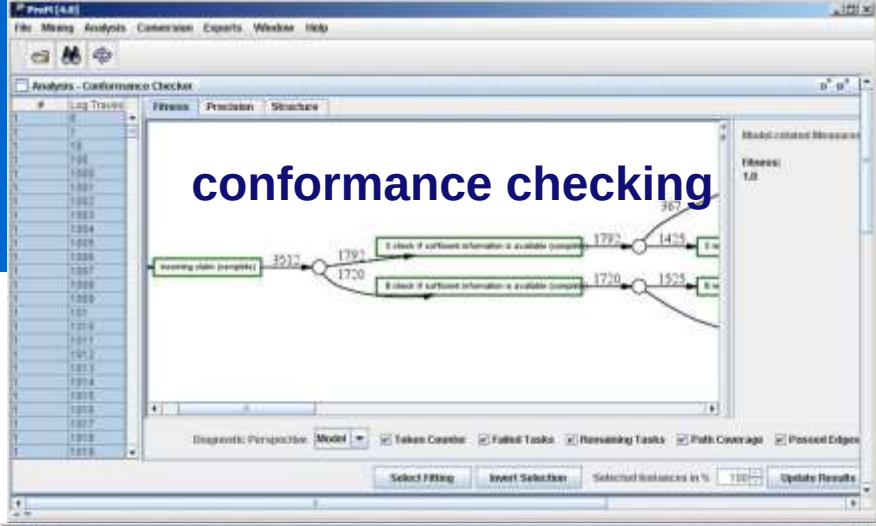



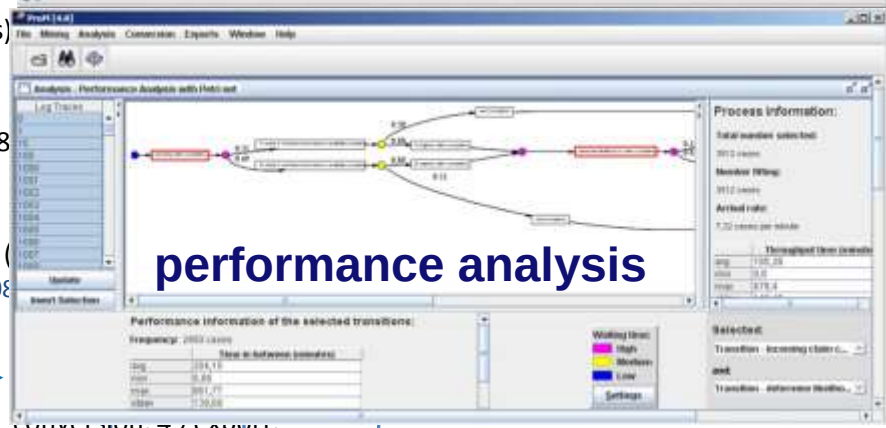

ns

(standard,
ns distributed
ackages)



# ProM 4.0




ProM 1.0
9/1/2004

ProM 1.1 ( 29 plug-ins: 7 analysis, 3 conversion, 9 export, 4 import, 0 filter, 6 mining)
1/1/2005

ProM 2.0
7/1/2005

ProM 2.1
9/1/2005

ProM 3.0
2/6/2006

ProM 3.1 (91 plug-ins: 25 analysis, 10 conversion, 22 export, 10 import, 9 log filter, 15 mining)
5/18/2006

**ProM 4.0 (142 plug-ins)**
11/23/2006

ProM 4.1 (149 plug-ins)
4/16/2007

ProM 4.2 (175 plug-ins)
9/14/2007

ProM 5.0 (228
6/17/2008

ProM 5.1 (
10/16/2008

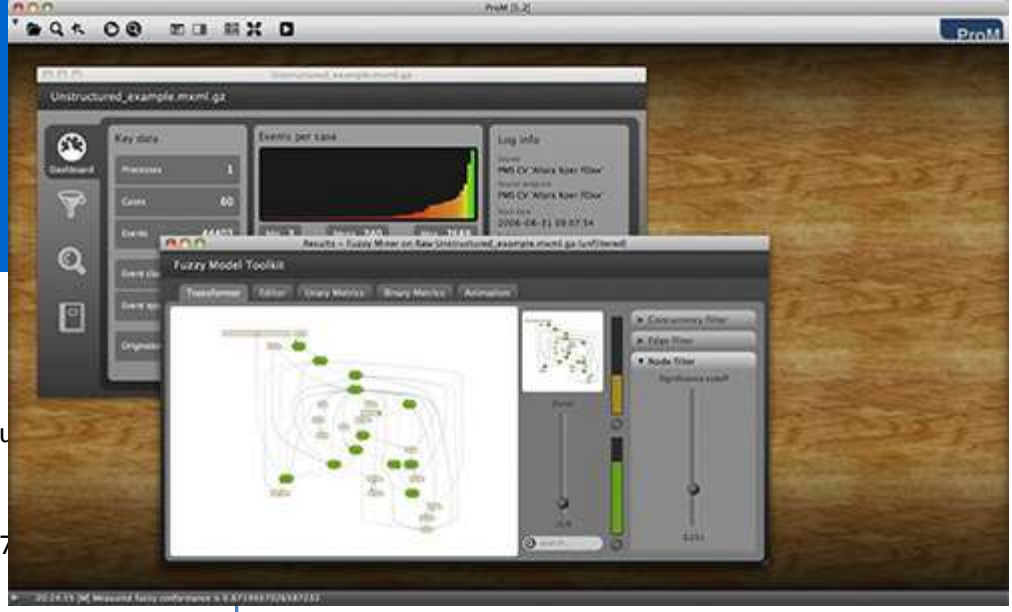
conversion, 49 export, 22 import, 32 filter, 37 mining)
9/3/2009

ProM 6.2 (extended, 676 plug-ins distributed over 99 packages)
5/27/2013

2004 2004 2005 2006 2007 2008 2009 2010 2011 2012 2013 2013

Today

# ProM 5.2



ProM 1.1 ( 29 plug-ins: 7 analysis, 3 conversion, 9 export, 4 import, 0 filter, 6 mining)
1/1/2005

ProM 2.0
7/1/2005

ProM 3.0
2/6/2006

ProM 3.1 (91 plug-ins: 25 analysis, 10 conversion, 22 export, 10 import, 9 log filter, 15 mining)
5/18/2006

ProM 1.0
9/1/2004

ProM 2.1
9/1/2005

ProM 4.0 (142 plug-ins)
11/23/2006

ProM 4.1 (149 plu
4/16/2007

ProM 4.2 (17
9/14/2007

ProM 5.0 (228 plug-ins)
6/17/2008

ProM 5.1 (254 plug-ins)
10/16/2008

**ProM 5.2 (274 plug-ins: 90 analysis, 44 conversion, 49 export, 22 import, 32 filter, 37 mining)**
9/3/2009

ProM 6.1 (322 plug-ins distributed over 57 packages)
9/5/2011

ProM 6.2 (standard, 468 plug-ins distributed over 68 packages)
8/16/2012

ProM 6.2 (extended, 676 plug-ins distributed over 99 packages)
5/27/2013

2004 | 2004 | 2005 | 2006 | 2007 | 2008 | 2009 | 2010 | 2011 | 2012 | 2013 | 2013

Today

# ProM 6.0: A new start …

ProM 1.1 ( 29 plug-ins: 7 analysis, 3 conversion, 9 export, 4 import, 0 filter, 6 mining)
1/1/2005

ProM 2.0
7/1/2005

ProM 3.0
2/6/2006

ProM 3.1 (91 plug-ins: 25 analysis, 10 conversion, 22 export,
filter,

ProM 4.1 (149 plug-ins)
4/16/2007

ProM 4.2 (175 plug-ins)
9/14/2007

ProM 5.0 (228 plug-ins)
6/17/2008

ProM 5.1 (254 plug-ins)
16/2008

(142 plug-i
6

ProM 5.2 (274 plug-ins: 90 analysis, 44 conversion, 49 export, 22 import, 32 filter, 37 mining)
9/3/2009

**ProM 6.0 (170 plug-ins distributed over 50 packages)**
9/27/2010

ProM 6.1 (322 plug-ins distributed over 57 packages)
9/5/2011

ProM 6.2 (standard, 468 plug-ins distributed over 68 packages)
8/16/2012

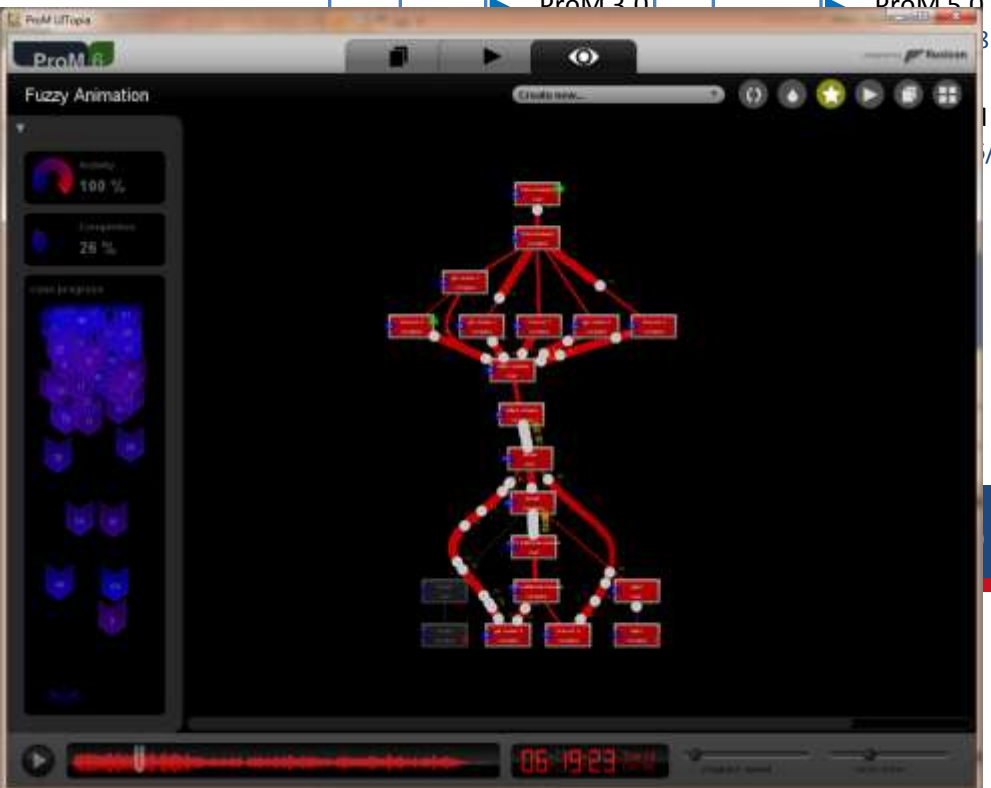
ProM 6.2 (extended, 676 plug-ins distributed over 99 packages)
5/27/2013

2008 | 2009 | 2010 | 2011 | 2012 | 2013

**2013**

**Today**

# ProM Today




ProM 1.1 ( 29 plug-ins: 7 analysis, 3 conversion, 9 export, 4 import, 0 filter, 6 mining)
1/1/2005

ProM 2.0
7/1/2005

ProM 4.1 (149 plug-ins)
4/16/2007

ProM 4.2 (175 plug-ins)
9/14/2007

ProM 5.2 (274 plug-ins: 90 analysis, 44 conversion, 49 export, 22 import, 32 filter, 37 mining)
9/3/2009

ProM 6.0 (170 plug-ins distributed over 50 packages)
9/27/2010

ProM 6.1 (322 plug-ins distributed over 57 packages)
9/5/2011

ProM 6.2 (standard, 468 plug-ins distributed over 68 packages)
8/16/2012

**ProM 6.2 (extended, 676 plug-ins distributed over 99 packages)**
5/27/2013

2010 | 2011 | 2012 | 2013

**2013**

**Today**

# ProM 6.3

## ProM

The process mining framework ProM provides a versatile and extendible environment for process mining. It provides plug-ins to extract different types of models from event logs, e.g., the construction of a process and organizational models. Moreover, it supports the conversion and analysis of models. Using conformance checking techniques models can also be compared with reality and existing models can be enhanced with additional information, e.g., indicating bottlenecks in a process.

## Improved features in ProM 6.3

- Improved the data-aware replayer
- Improved event-transition conformance
- Improved use of classifiers for heuristics miner
- Improved PNML import w.r.t. CPN Tools models
- Improved alignments between traces and Petri nets
- Improved import/export of transition systems
- Improved model repair
- General bug fixing

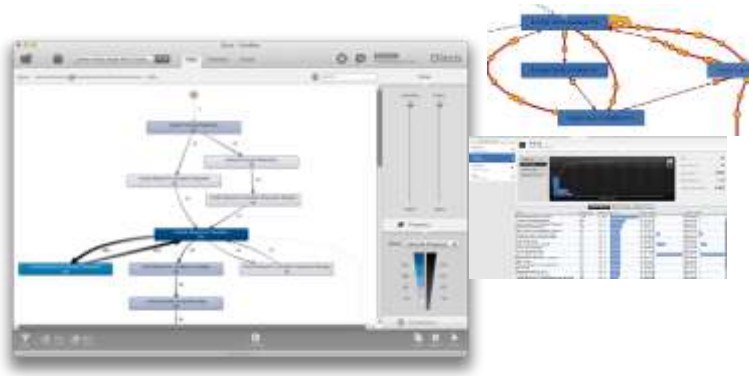
## New features in ProM 6.3

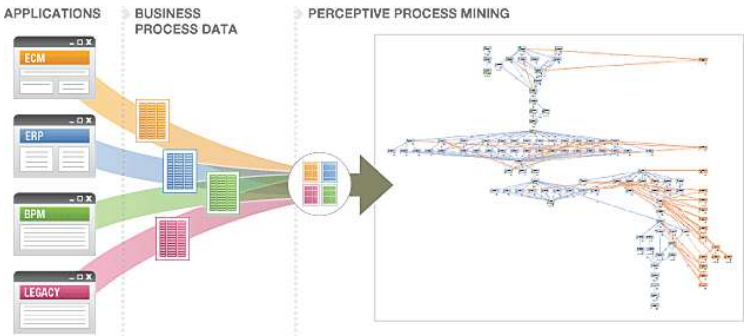
- Petri nets with data
- Stochastic Petri nets
- Support for configurable Petri nets
- Discovery and replay using divide-and-conquer techniques
- Discovery of services
- A new dotted chart
- A new genetic miner for structured processes
- A new inductive miner for structured processes
- Support for configuring configurable processes
- XES Cost extension
- Smart binding of classifier keys
- Caching of latest log info
- CSV export for key-value sets
- Import of BPMN 2.0 models from .bpmn files
- Established vs. runner-up packages

ProM 6.3 can be downloaded from www.promtools.org/prom6 or sourceforge.net/projects/prom.

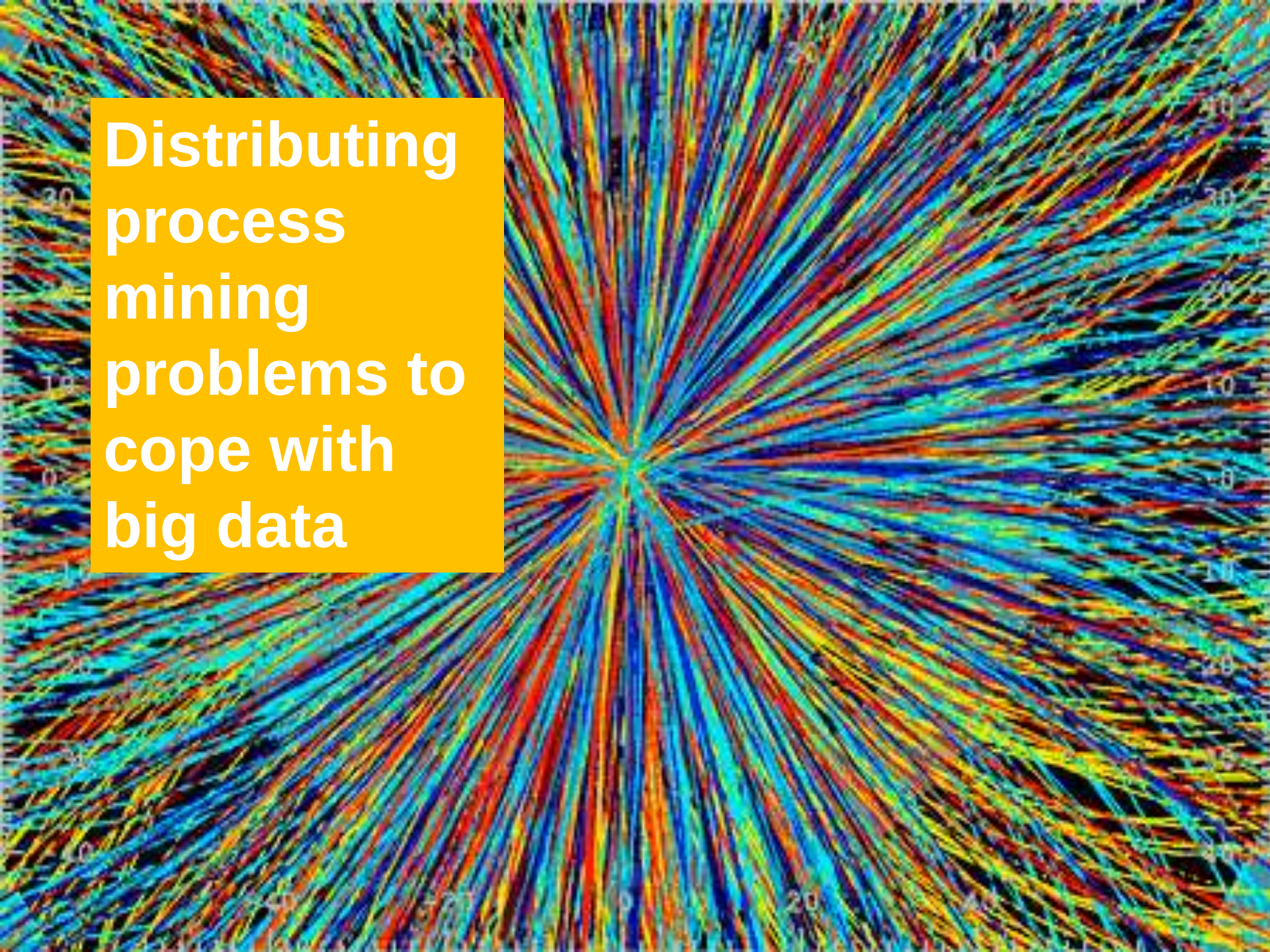
# Commercial Alternatives

- **Disco (Fluxicon)**
- **Perceptive Process Mining**
  **(before Futura Reflect and BPM|one)**
- **ARIS Process Performance Manager**
- **QPR ProcessAnalyzer**
- **Interstage Process Discovery (Fujitsu)**
- **Discovery Analyst (StereoLOGIC)**
- **XMAnalyzer (XMPro)**
- **…**






# Challenges


Distributing process mining problems to cope with big data


streaming
event data
(sensors, RFID, messages, etc.)

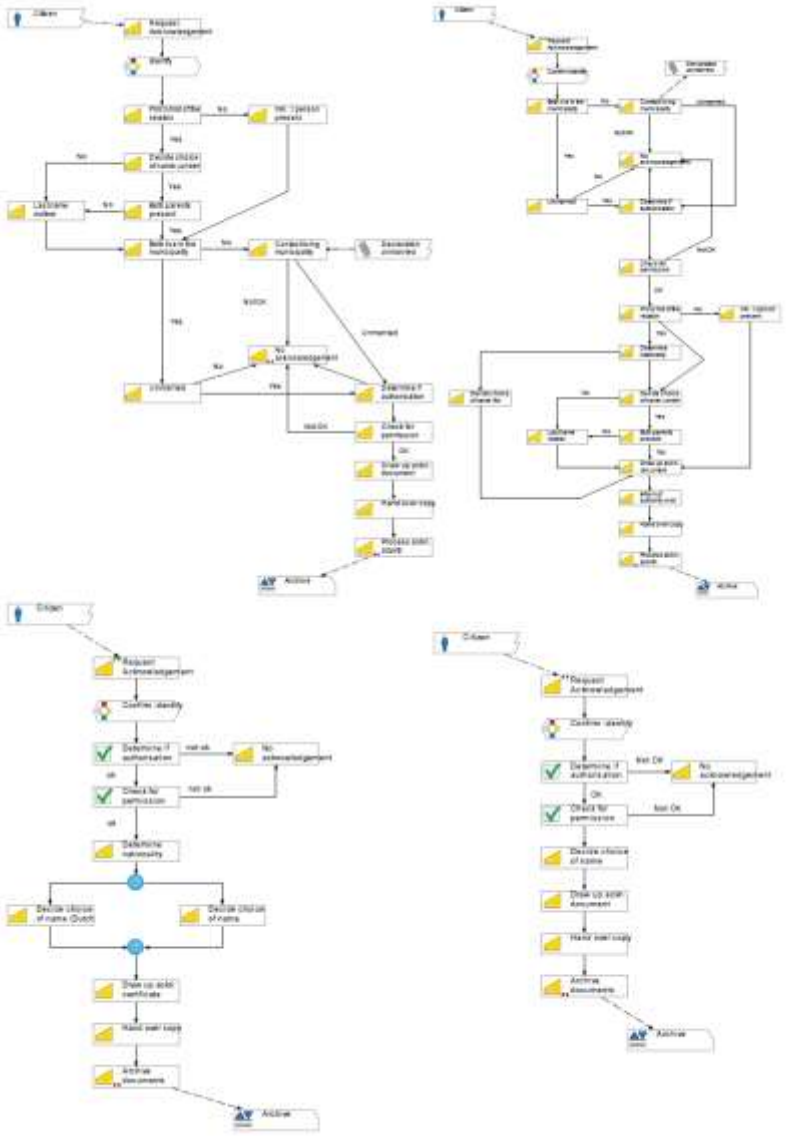
# On-the-fly process mining


## Operational support


Concept drift


# cross-organizational / comparative process mining


# Example: Dutch Municipalities
## (see Coselog project www.win.tue.nl/coselog/)

**+/- 430 Dutch municipalities**

# Another Example: Suncorp case

## End to end process has between 250-1000 process steps


- 25+ steps
- 50+ steps
- 75+ steps
- 100+ steps

**500** steps

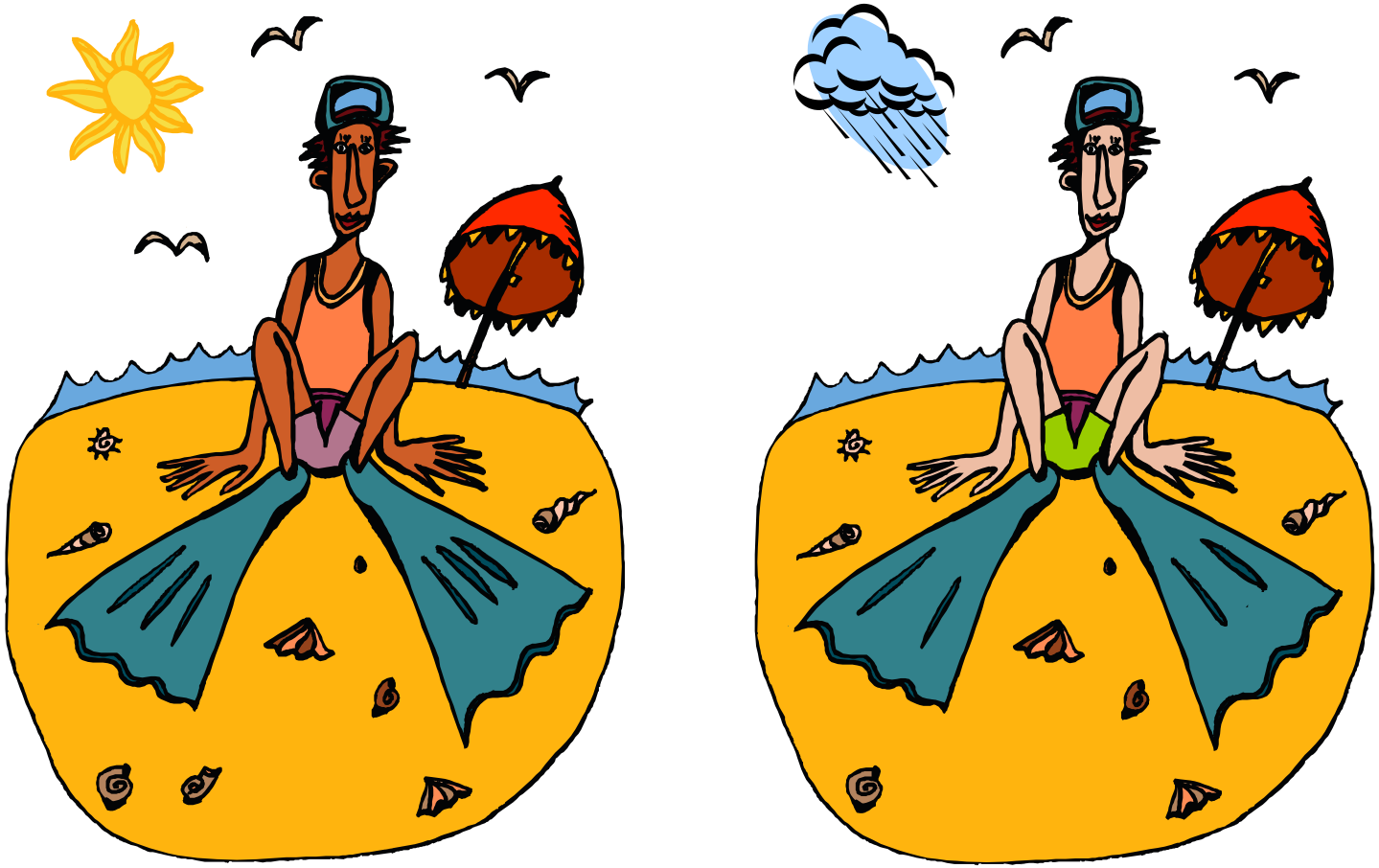
*Sources: Guidewire reference models, GIO CISSS Project, CI US&S P4PI Project*

| | AAMI | Apia | bingle | CIL | GIO | JUST·CAR | | SUNCORP | vero |
|---|---|---|---|---|---|---|---|---|---|
| Home | ☑ | ☑ | | | ☑ | | ☑ | ☑ | ☑ |
| Motor | ☑ | ☑ | ☑ | | ☑ | ☑ | ☑ | ☑ | ☑ |
| Commercial | ☑ | | | ☑ | ☑ | | | ☑ | ☑ |
| Liability | ☑ | ☑ | | | ☑ | | | ☑ | ☑ |
| CTP / WC | ☑ | ☑ | | | ☑ ☑ | | ☑ | ☑ | |

**30** variations

**Thanks to Marcello La Rosa (QUT)**

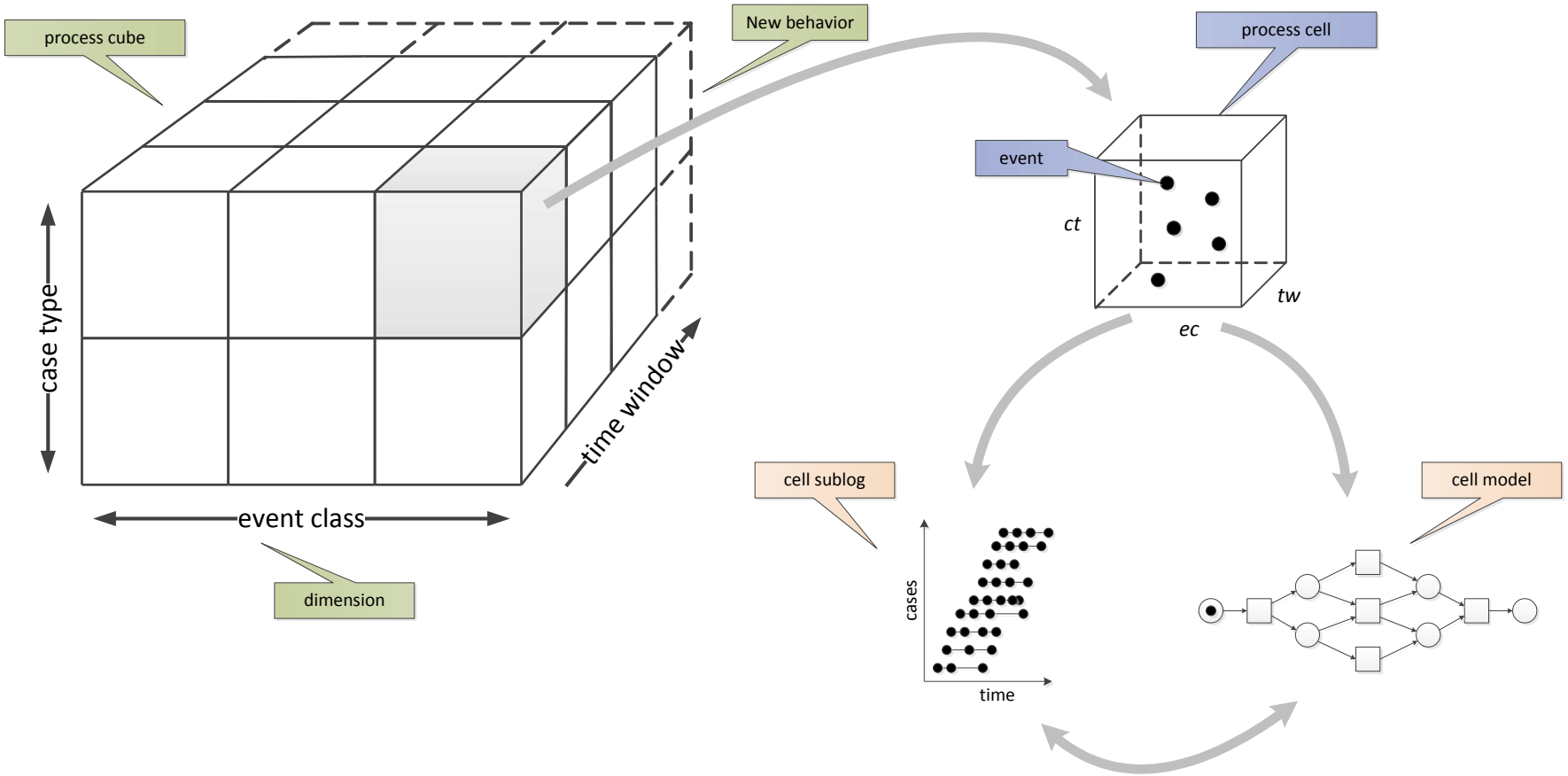
# Two variants of the same process …



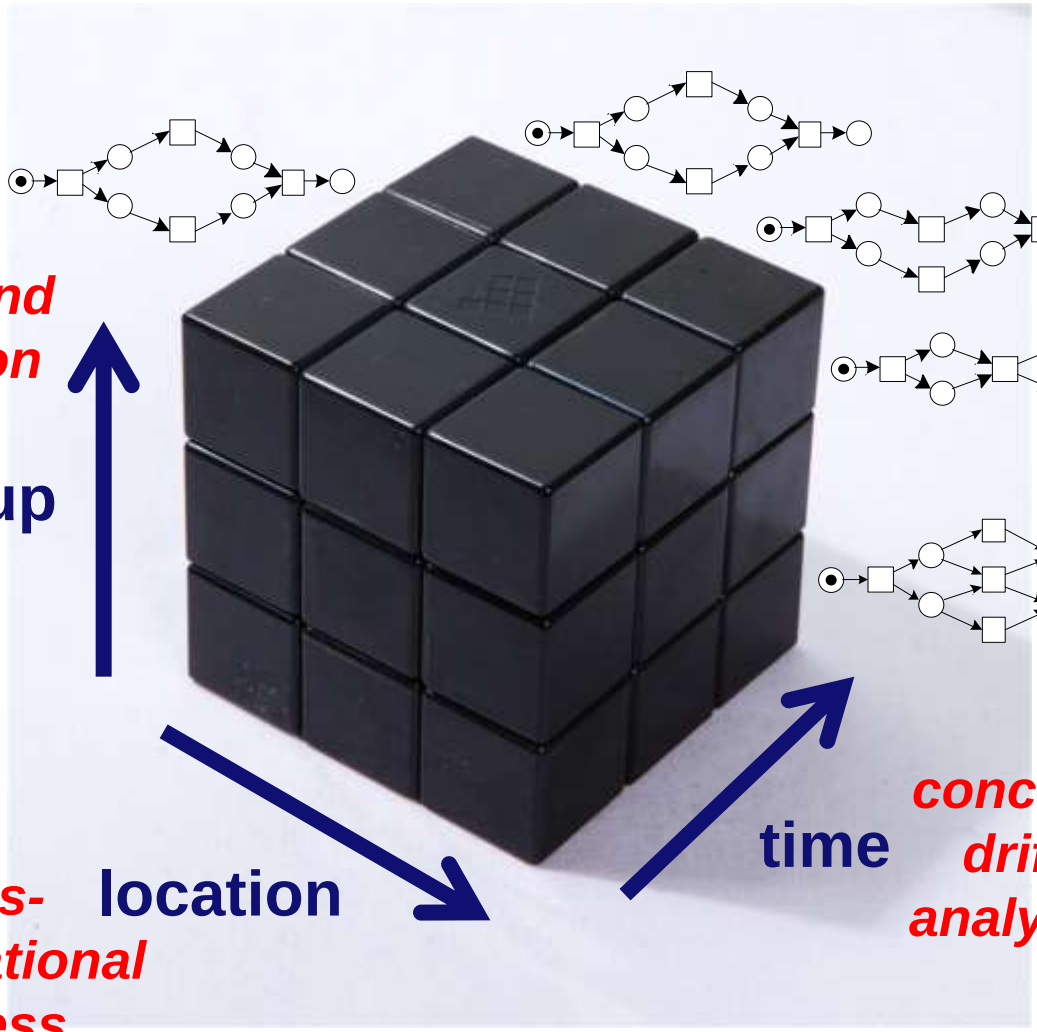

context aware
process mining

# Process Cube

# Process Cube: Dimensions and Cells



# Example Dimensions



*clustering and classification*

group

acbe
abce
ade
acbe
abbe
abce
ade
ade

*cross-organizational process mining*

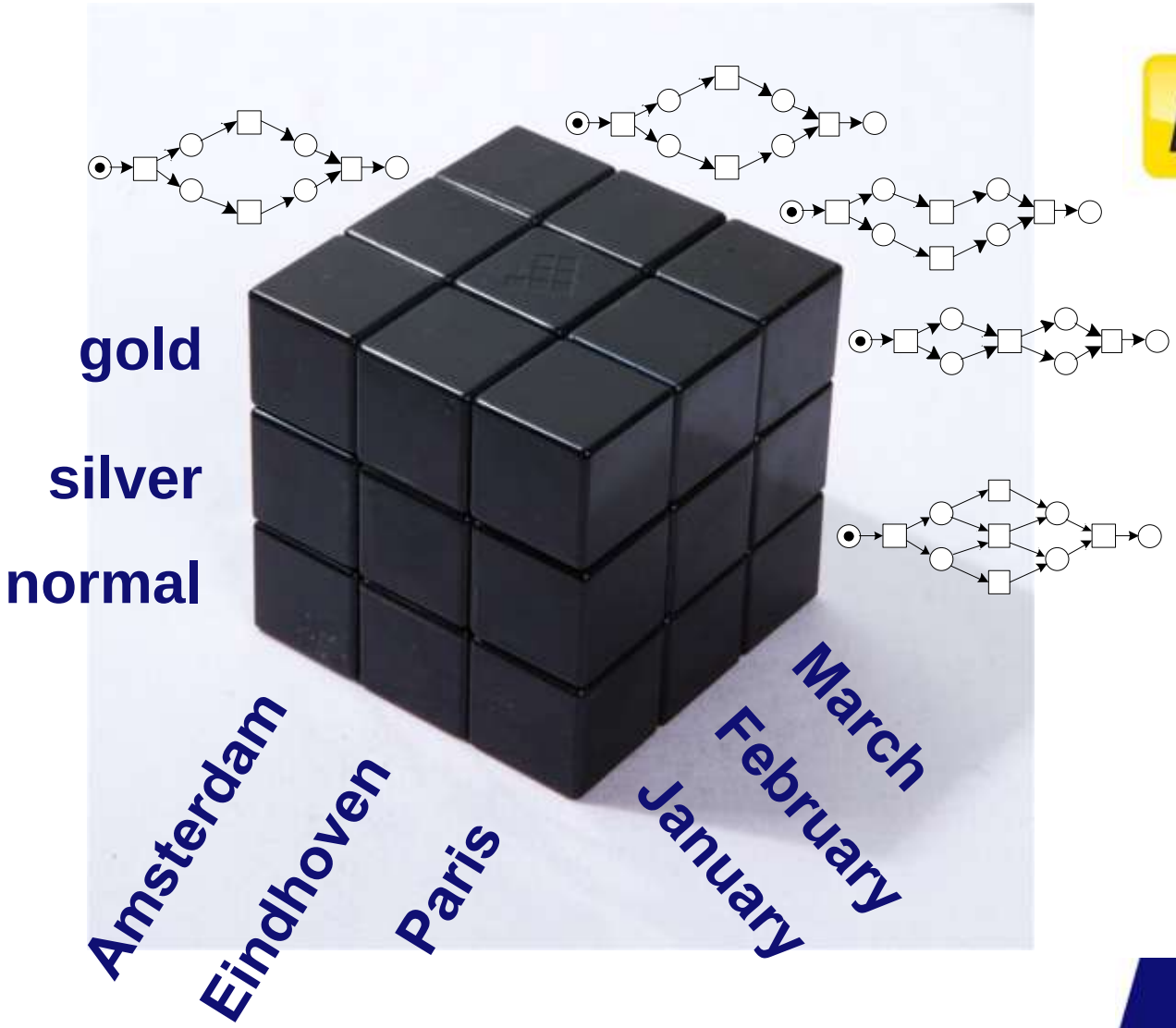
location

time

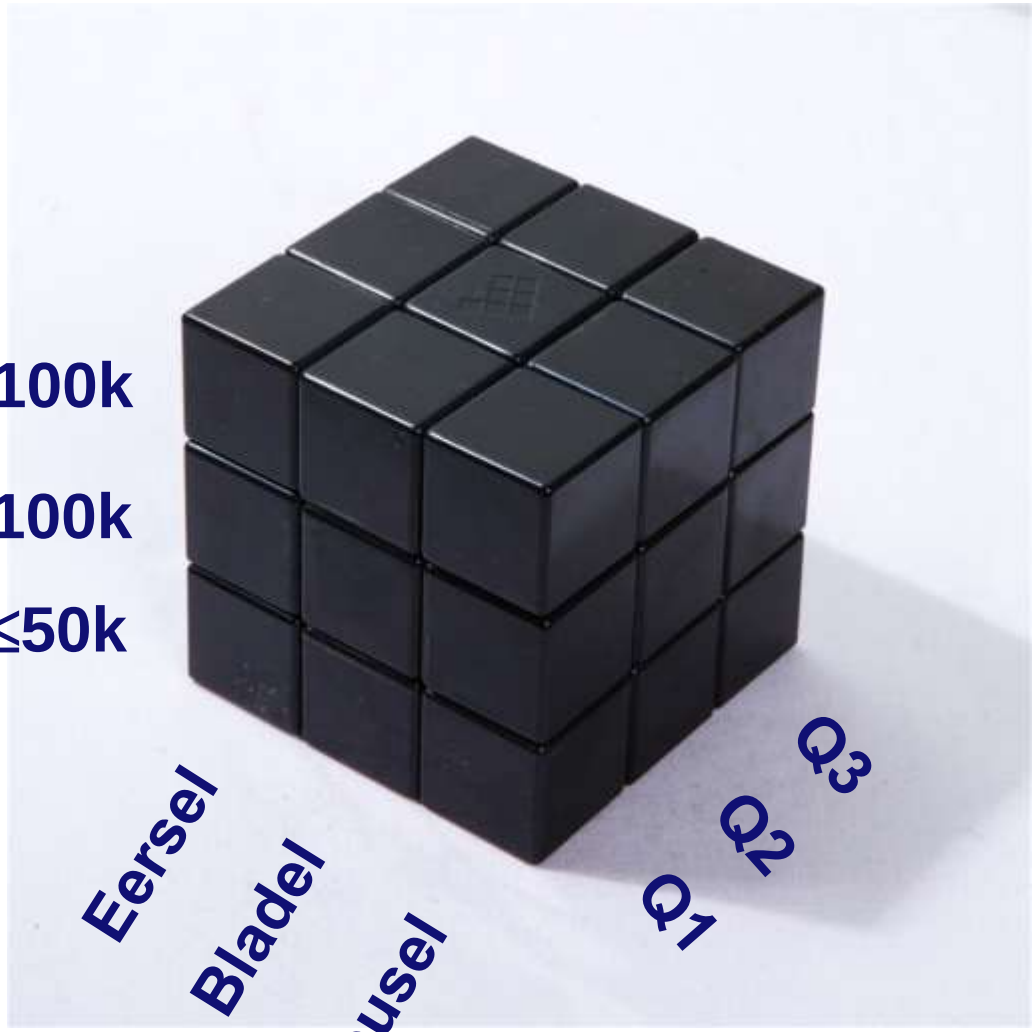
*concept drift analysis*

*cases may be distributed over multiple cells, the same event may belong to multiple cells*

# Example

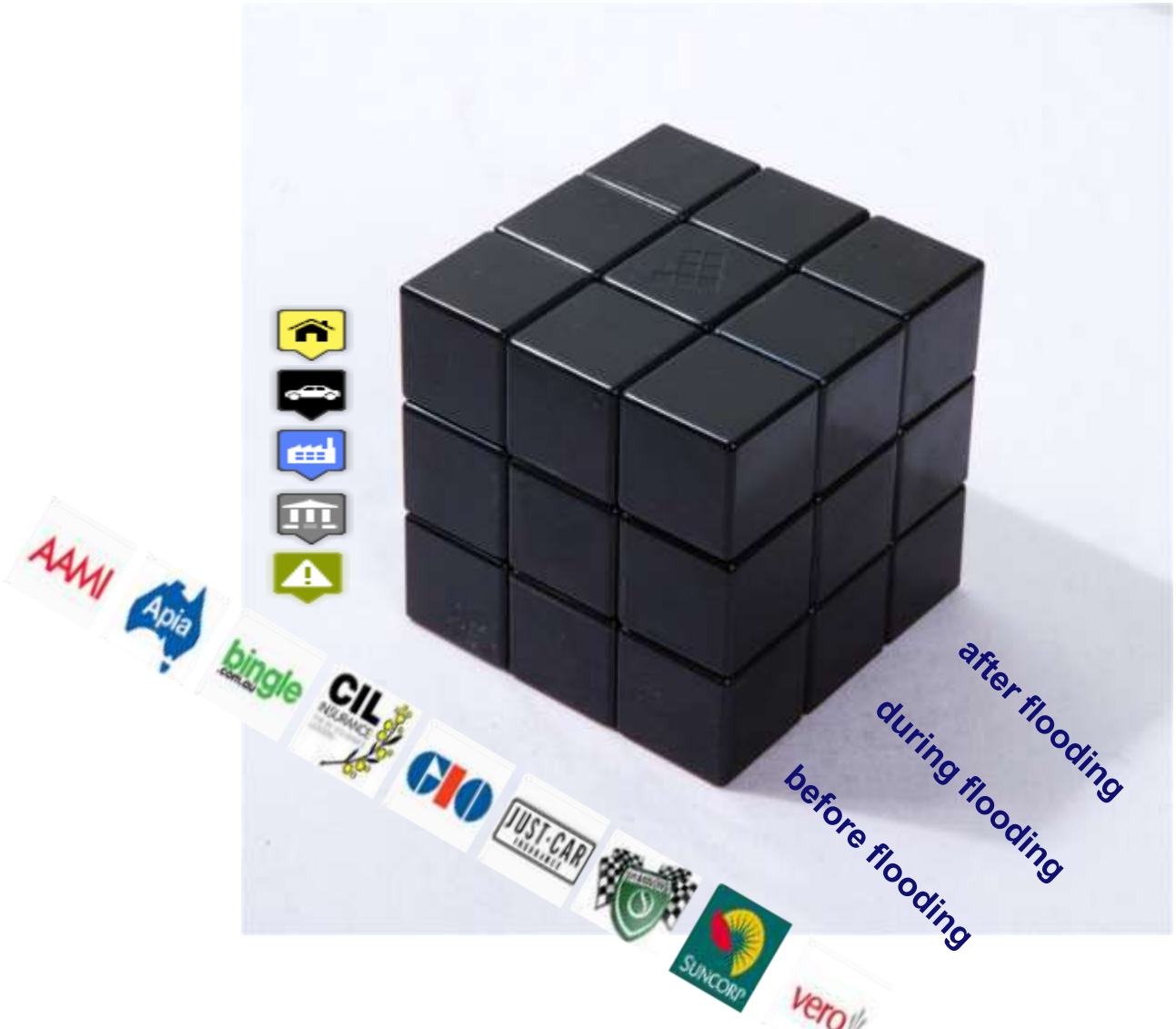

gold
silver
normal
Amsterdam
Eindhoven
Paris
January
February
March

# Another Example



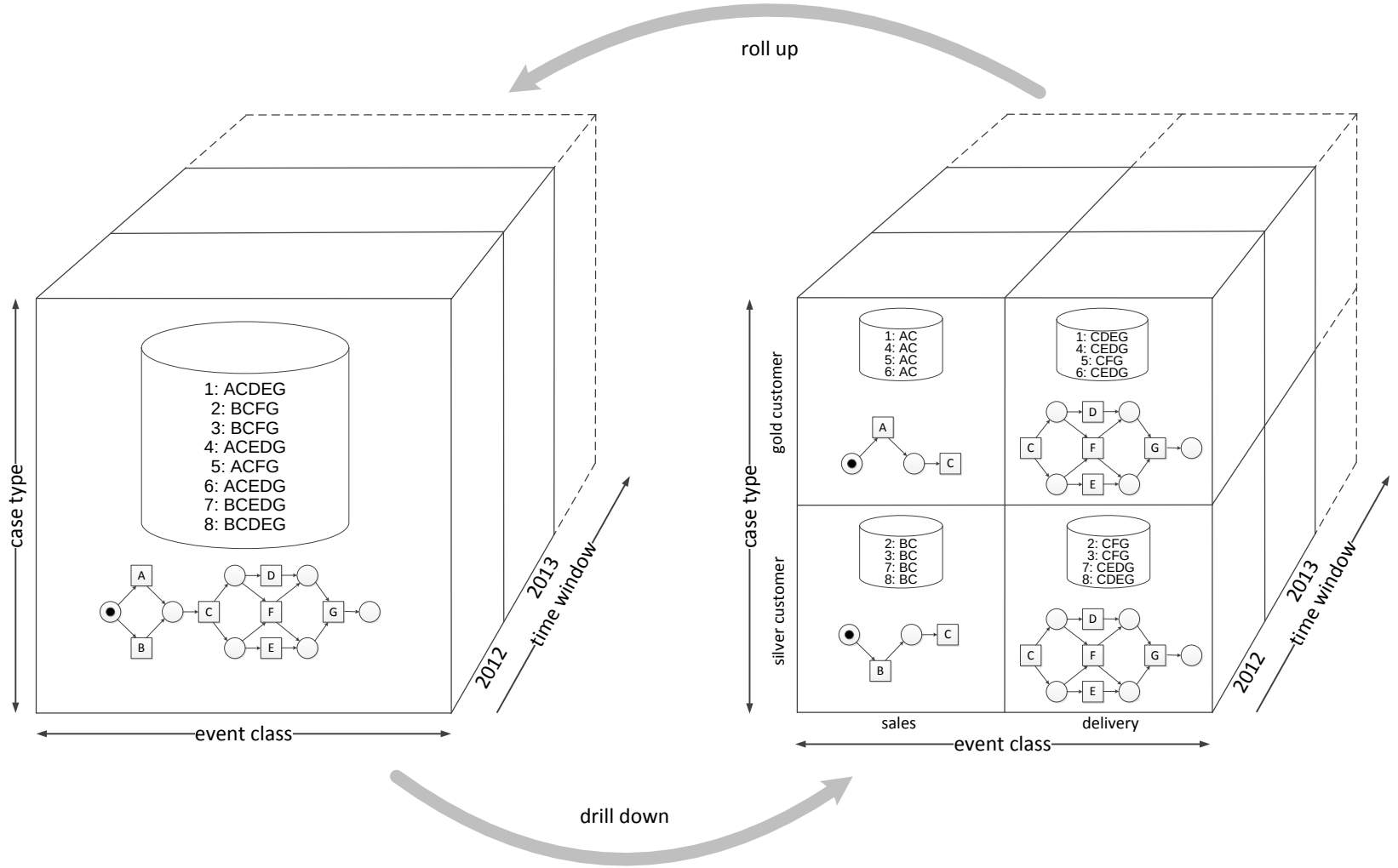

CoSeLoG Deelnemers

# Another Example

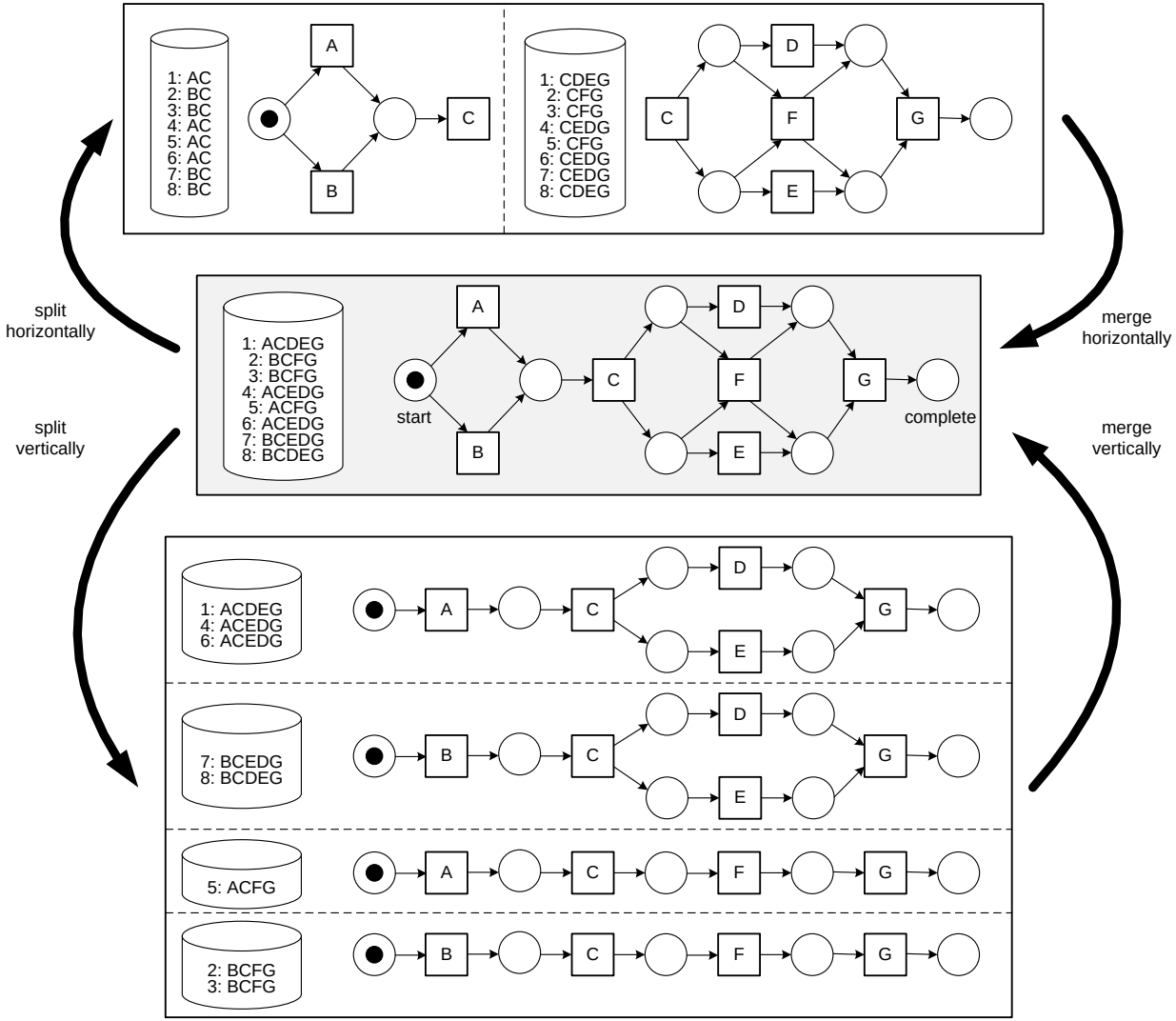



SUNCORP

# Roll up and drill down



# Two foundational ways of spitting event data: horizontal or vertical



# Ingredients

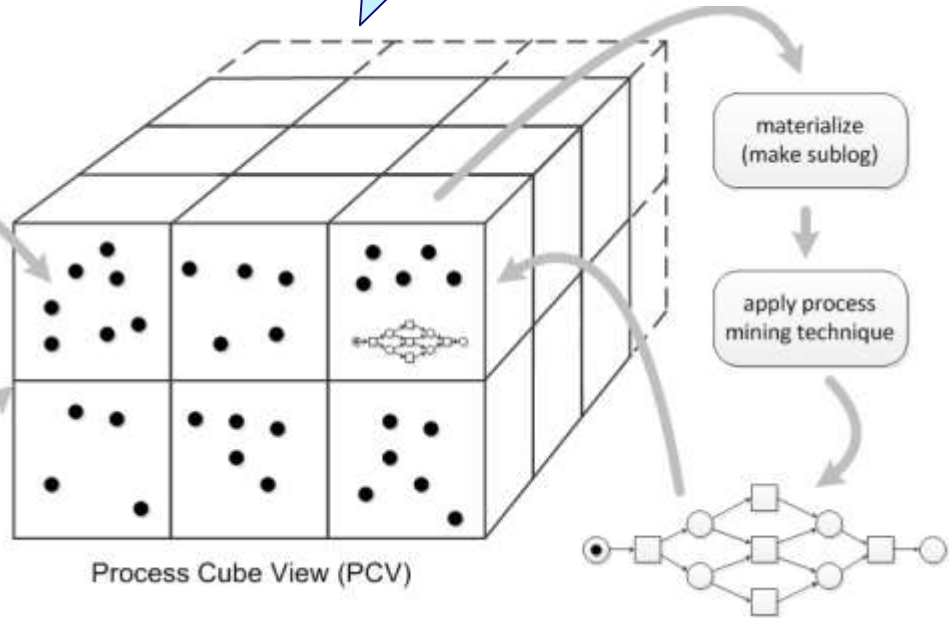
**Event Base (EB) with (raw) event data**

**Process Cube View (PCV) based on PCS that can be linked to real events**

| case id | properties | | event id | properties | | | | |
|---|---|---|---|---|---|---|---|---|
| | type | total | | timestamp | activity | resource | cost | ... |
| 1 | gold | 1600 | 35654423 | 30-12-2012:11.02 | A | John | 300 | ... |
| | | | 35654424 | 30-12-2012:11.06 | C | Ann | 400 | ... |
| | | | 35654425 | 30-12-2012:11.12 | D | Pete | 100 | ... |
| | | | 35654426 | 30-12-2012:11.18 | E | Pete | 400 | ... |
| | | | 35654427 | 30-12-2012:11.19 | G | Pete | 400 | ... |
| 2 | silver | 900 | 35655526 | 30-12-2012:16.10 | B | John | 200 | ... |
| | | | 35655527 | 30-12-2012:16.14 | C | Ann | 450 | ... |
| | | | 35655528 | 30-12-2012:16.26 | F | Sue | 150 | ... |
| | | | 35655529 | 30-12-2012:16.36 | G | Sue | 100 | ... |
| ... | ... | ... | ... | ... | ... | ... | ... | ... |

Event Base (EB)


Process Cube Structure (PCS)




Process Cube View (PCV)

**Process Cube Structure (PCS) defining dimensions independent of content**

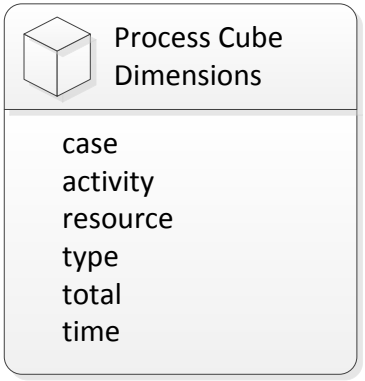
# Event Base $(E, P, \pi)$

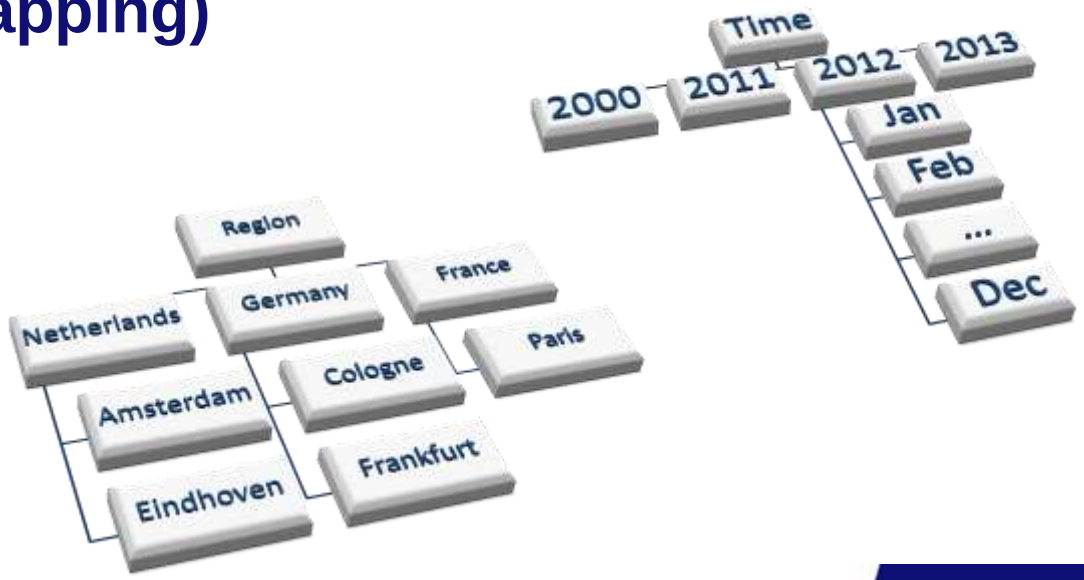
- **Set of events $E$**
- **Set of properties $P$**
- **Partial mapping $\pi$**
  **($\pi_p(e)=v$ means event $e$ has property $p$ with value $v$)**

| case id | properties | | event id | properties | | | |
|---|---|---|---|---|---|---|---|
| | type | total | | timestamp | activity | resource | cost |
| 1 | gold | 1600 | 35654423 | 30-12-2012:11.02 | A | John | 300 |
| | | | 35654424 | 30-12-2012:11.06 | C | Ann | 400 |
| | | | 35654425 | 30-12-2012:11.12 | D | Pete | 100 |
| | | | 35654426 | 30-12-2012:11.18 | E | Pete | 400 |
| | | | 35654427 | 30-12-2012:11.19 | G | Pete | 400 |
| 2 | silver | 900 | 35655526 | 30-12-2012:16.10 | B | John | 200 |
| | | | 35655527 | 30-12-2012:16.14 | C | Ann | 450 |
| | | | 35655528 | 30-12-2012:16.26 | F | Sue | 150 |
| | | | 35655529 | 30-12-2012:16.36 | G | Sue | 100 |
| ... | ... | ... | ... | ... | ... | ... | ... |

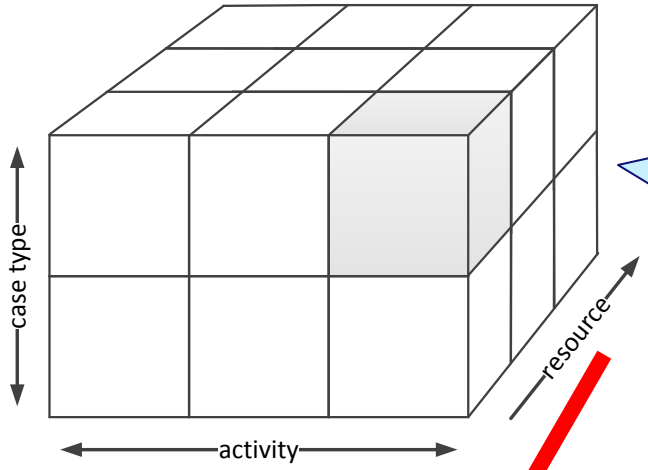
# Process Cube Structure *(D,type,hier)*

- ***D*** **is the set of dimensions**
- ***type*** **defines the type of each dimension (string, int, etc.)**
- ***hier*** **defines the hierarchy per dimension (does <u>not</u> need to be a tree, elements at the same level may be partially overlapping)**



# Compatible: $D \subseteq P$ and correct types



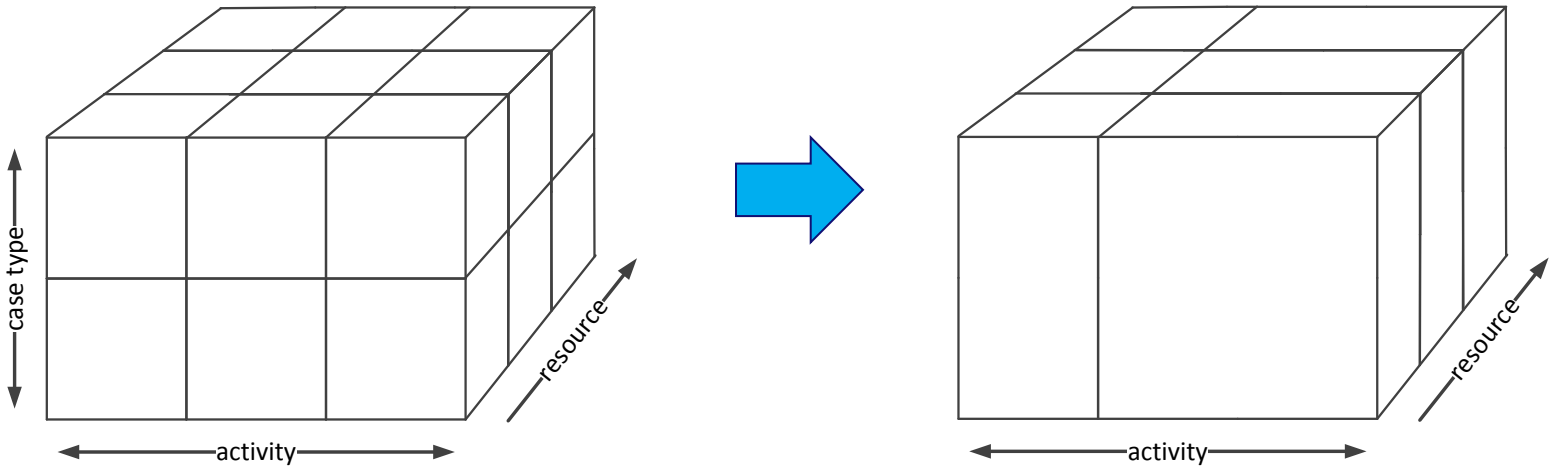

**Process Cube Structure (PCS) defining dimensions *D* independent of content**

| case id | properties | | event id | properties | | | |
|---|---|---|---|---|---|---|---|
| | type | total | | timestamp | activity | resource | cost |
| 1 | gold | 1600 | 35654423 | 30-12-2012:11.02 | A | John | 300 |
| | | | 35654424 | 30-12-2012:11.06 | C | Ann | 400 |
| | | | 35654425 | 30-12-2012:11.12 | D | Pete | 100 |
| | | | 35654426 | 30-12-2012:11.18 | E | Pete | 400 |
| | | | 35654427 | 30-12-2012:11.19 | G | Pete | 400 |
| 2 | silver | 900 | 35655526 | 30-12-2012:16.10 | B | John | 200 |
| | | | 35655527 | 30-12-2012:16.14 | C | Ann | 450 |
| | | | 35655528 | 30-12-2012:16.26 | F | Sue | 150 |
| | | | 35655529 | 30-12-2012:16.36 | G | Sue | 100 |
| ... | ... | ... | ... | ... | ... | ... | ... |

**Event Base (EB) with (raw) event data**

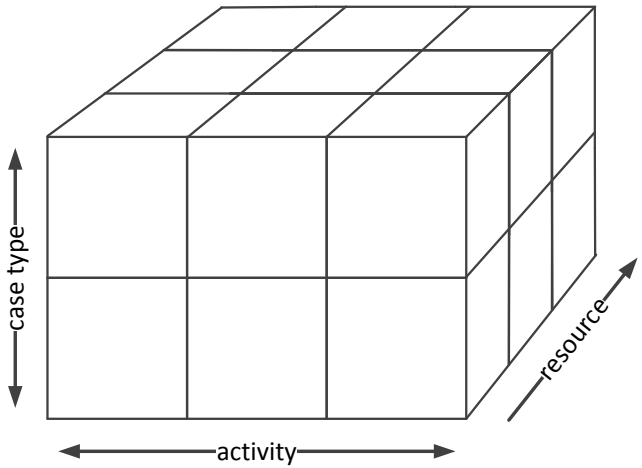
# Process Cube View $(D_{sel}, sel)$

- View on a process cube structure $(D,type,hier)$.
- $D_{sel} \subseteq D$ are the selected dimensions.
- $sel$ is a function determining the structure of each dimension in $D_{sel}$.

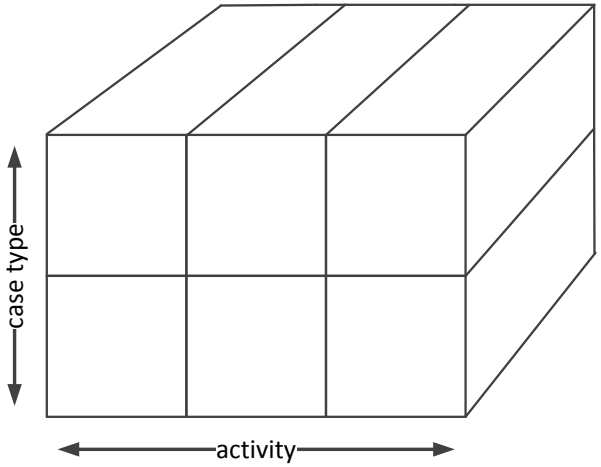


# Process Cube View Example (1/3): Dimension *resource* is not selected

$PCS = (D, type, hier)$

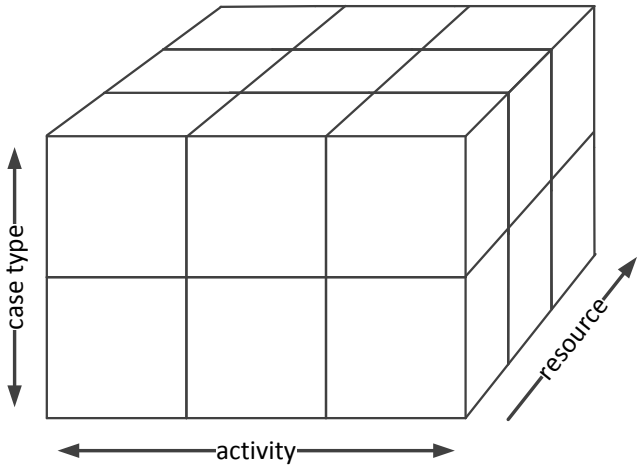



$PCV = (D_{sel}, sel)$

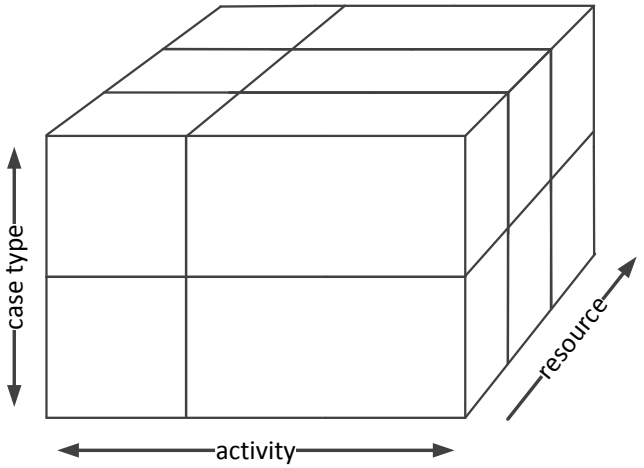


# Process Cube View Example (2/3): Dimension *activity* is coarsened

$PCS = (D, type, hier)$



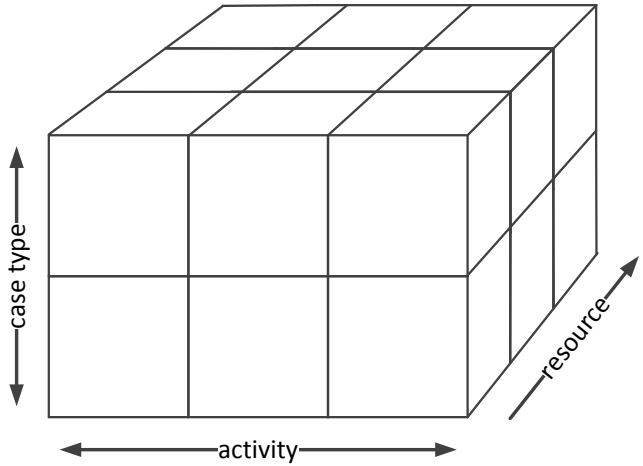

$PCV = (D_{sel}, sel)$



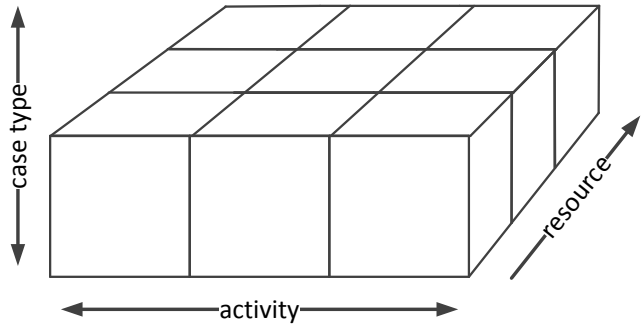

**e.g., per subprocess rather than per activity**

# Process Cube View Example (3/3): Dimension *case type* is filtered

$PCS = (D, type, hier)$

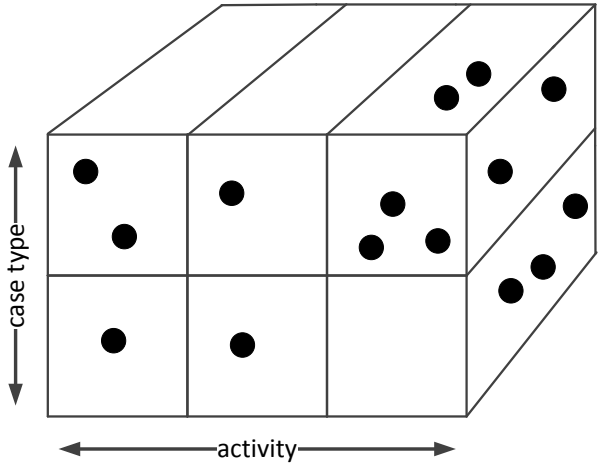



$PCV = (D_{sel}, sel)$

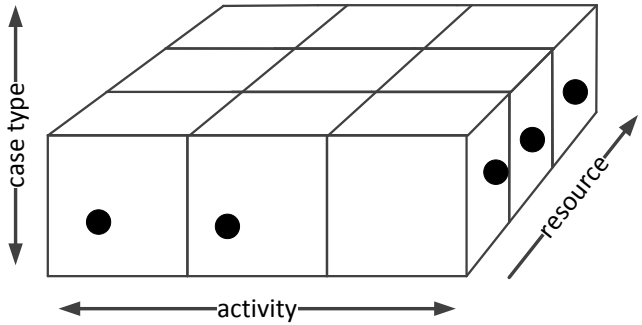



**part of the case type dimension is not selected**

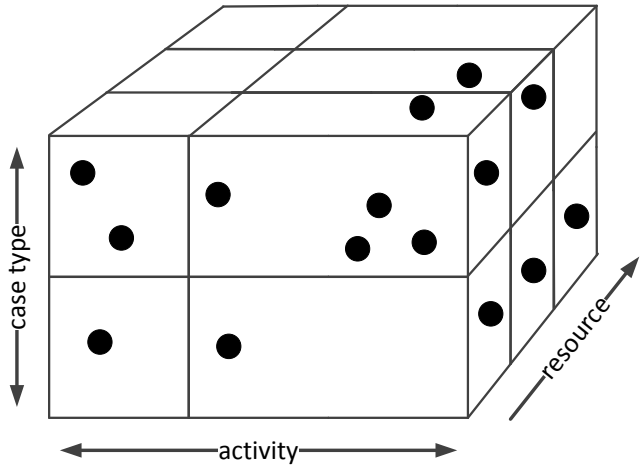
# Materialized Process Cube View




**3 x 2 x 1 = 6 event logs**

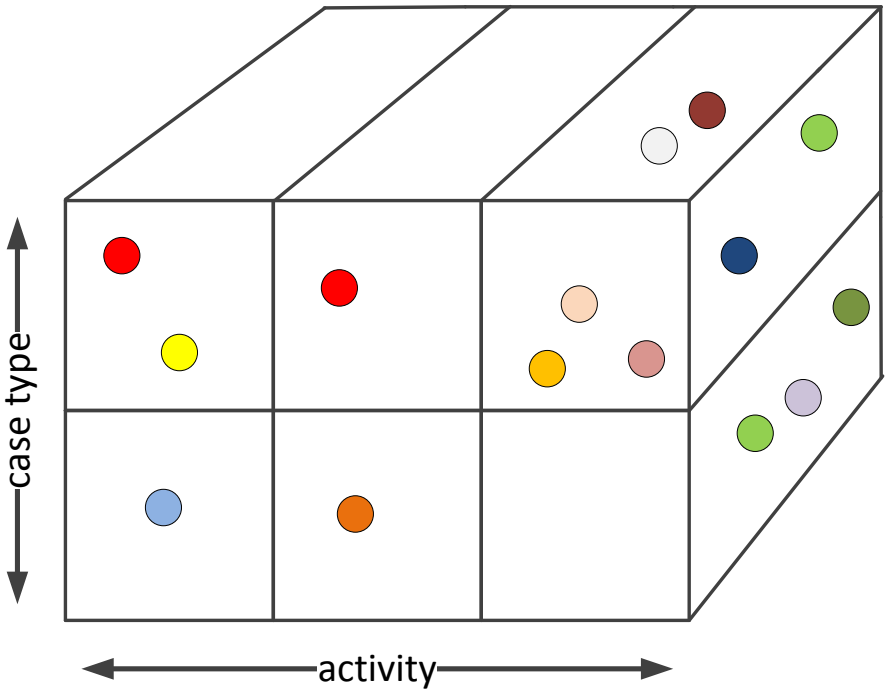



**3 x 1 x 3 = 9 event logs**




**2 x 2 x 3 = 12 event logs**

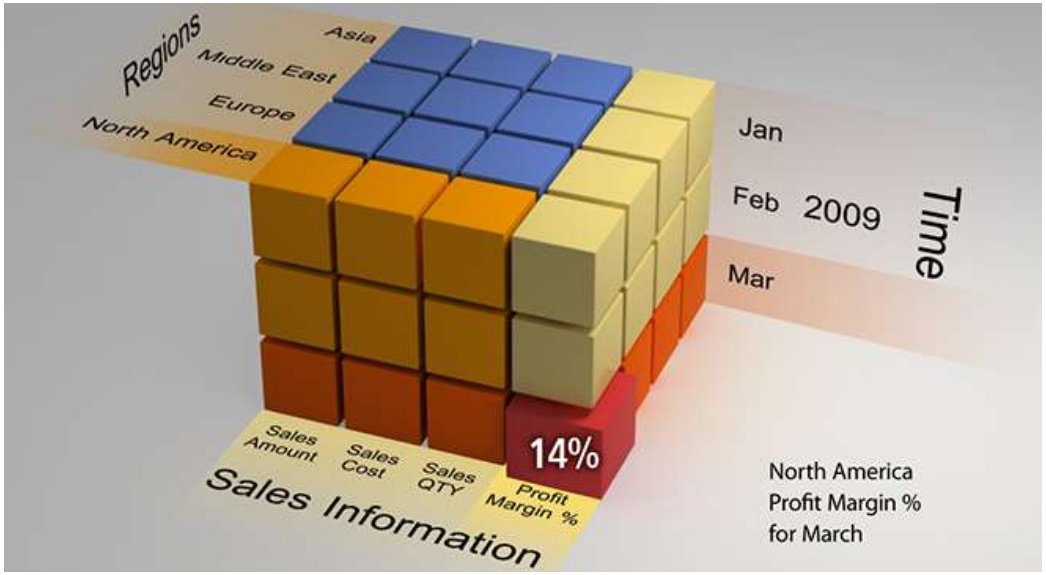
# An event may be part of multiple cells!!

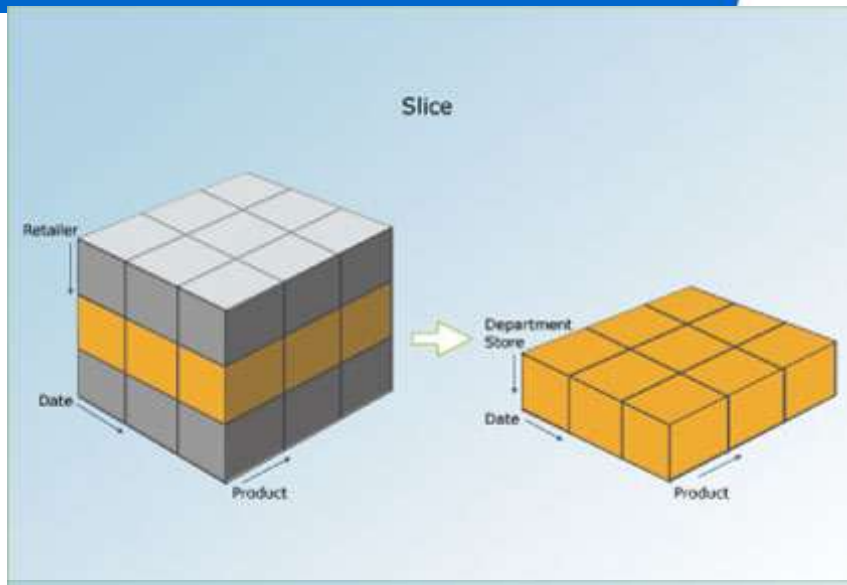


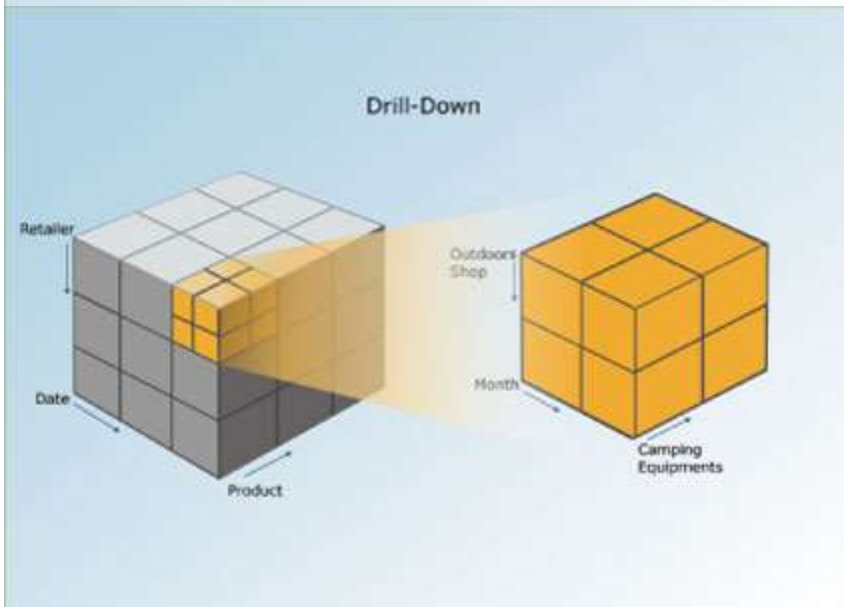

- departments may overlap
- subprocesses share interfaces
- contracts involving multiple parties
- ….

- events are unique,
- but may be part of multiple cells

# OLAP (Online Analytical Processing)



# OLAP operations can be formalized using the definitions in the paper

**Definition 7 (Slice).** *Let* $PCS = (D, type, hier)$ *be a process cube structure and* $PCV = (D_{sel}, sel)$ *a view of* $PCS$*. For any* $d \in D_{sel}$ *and* $V \in sel(d)$*:* $slice_{d,V}(PCV) = (D'_{sel}, sel')$ *with* $D'_{sel} = D_{sel} \setminus \{d\}$*,* $sel'(d) = \{V\}$*, and* $sel'(d') = sel(d')$ *for* $d' \in D \setminus \{d\}$*.*

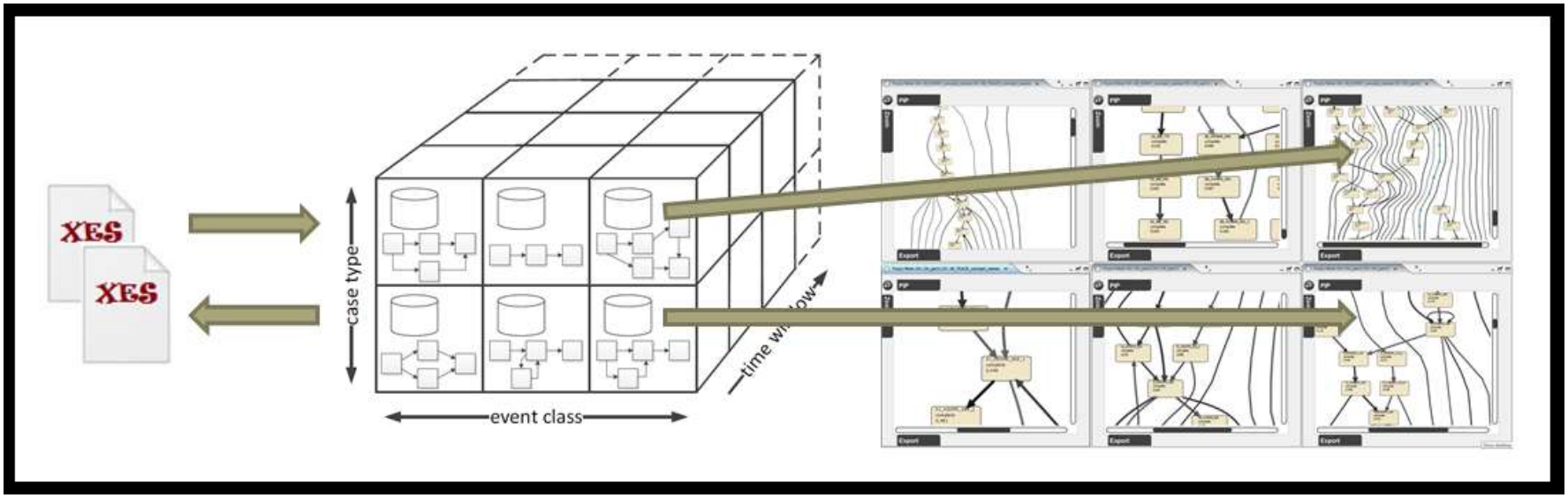
**Definition 8 (Dice).** *Let* $PCS = (D, type, hier)$ *be a process cube structure and* $PCV = (D_{sel}, sel)$ *a view of* $PCS$*. Let* $res \in D_{sel} \not\to \mathcal{U}_H$ *be a restriction such for any* $d \in dom(res)$*:* $res(d) \subseteq sel(d)$*.* $dice_{res}(PCV) = (D_{sel}, sel')$ *with* $sel'(d) = res(d)$ *for* $d \in dom(res)$ *and* $sel'(d) = sel(d)$ *for* $d \in D \setminus dom(res)$*.*

**Definition 9 (Change Granularity).** *Let* $PCS = (D, type, hier)$ *be a process cube structure and* $PCV = (D_{sel}, sel)$ *a view of* $PCS$*. Let* $d \in D_{sel}$ *and* $H \in \mathcal{U}_H$ *such that:*

- $H \subseteq hier(d)$,
- $\bigcup H = \bigcup sel(d)$*, and*
- *for any* $V_1, V_2 \in H$*:* $V_1 \subseteq V_2$ *implies* $V_1 = V_2$*.*

$chgr_{d,H}(PCV) = (D_{sel}, sel')$ *with* $sel'(d) = H$*, and* $sel'(d') = sel(d')$ *for* $d' \in D \setminus \{d\}$*.*

# Initial Implementation (work of Tatiana Mamaliga)

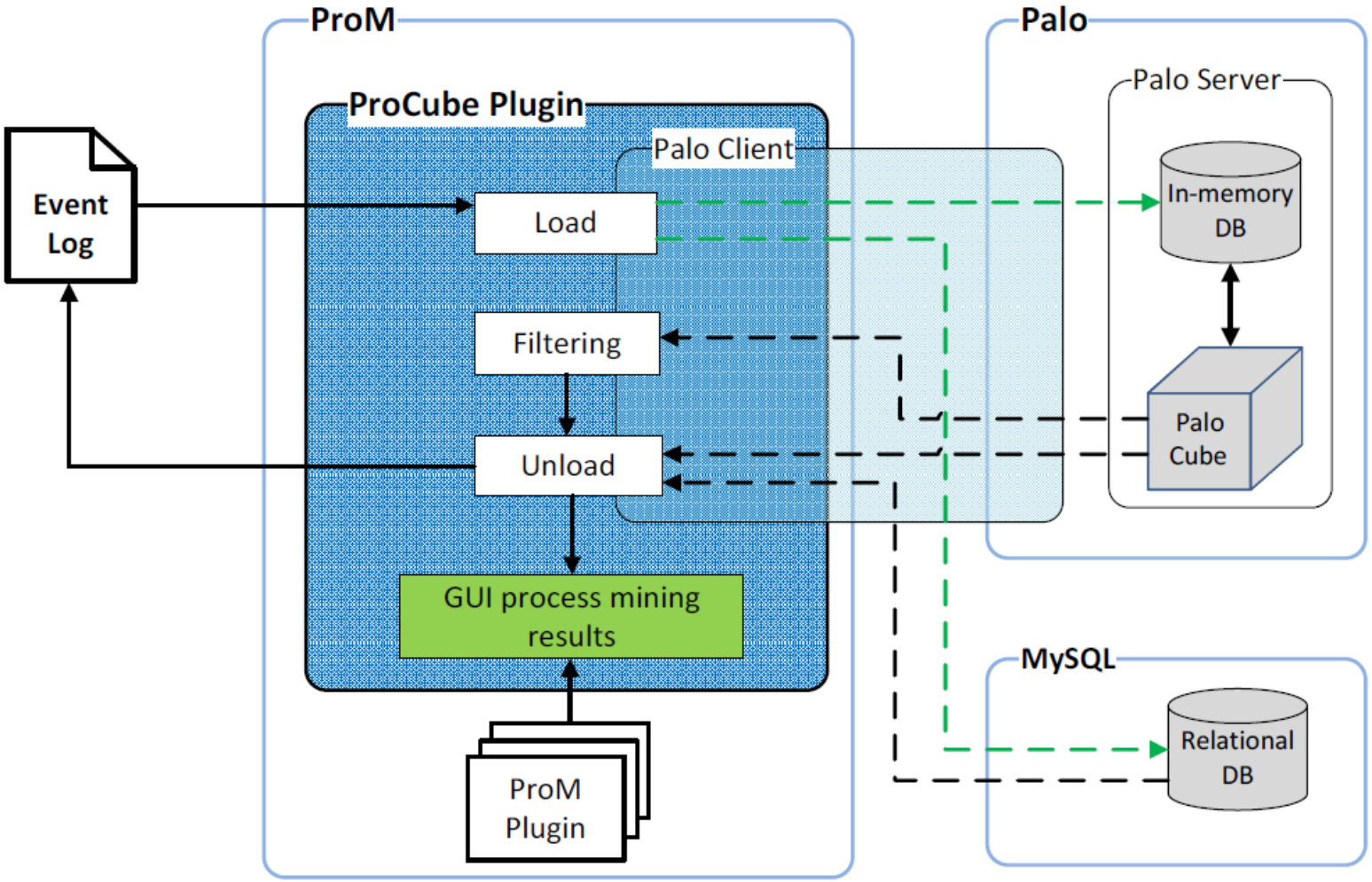



Palo by Jedox
plan analyse report

MySQL®

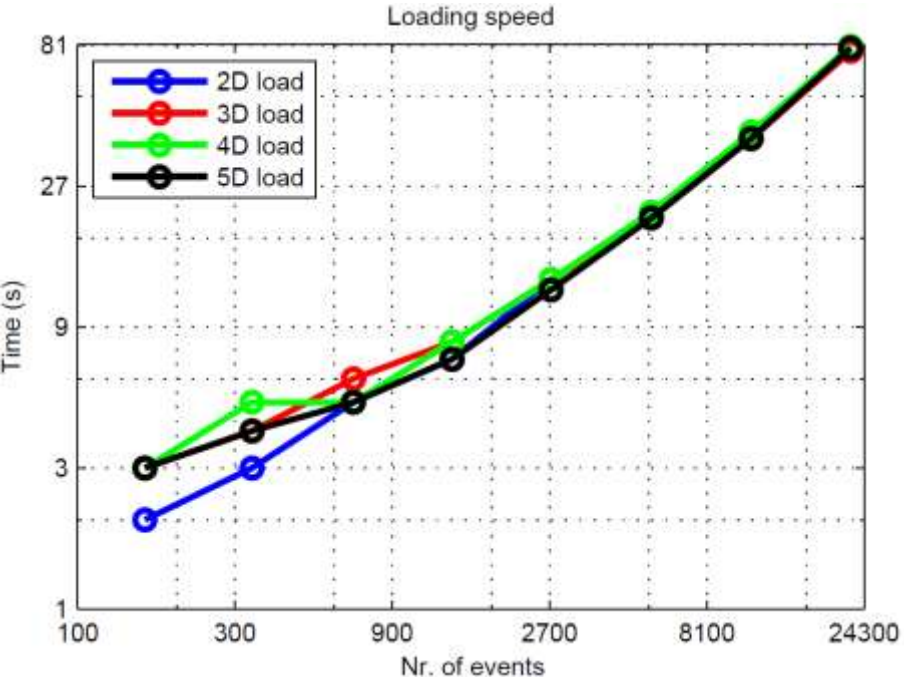
ProM
process mining workbench

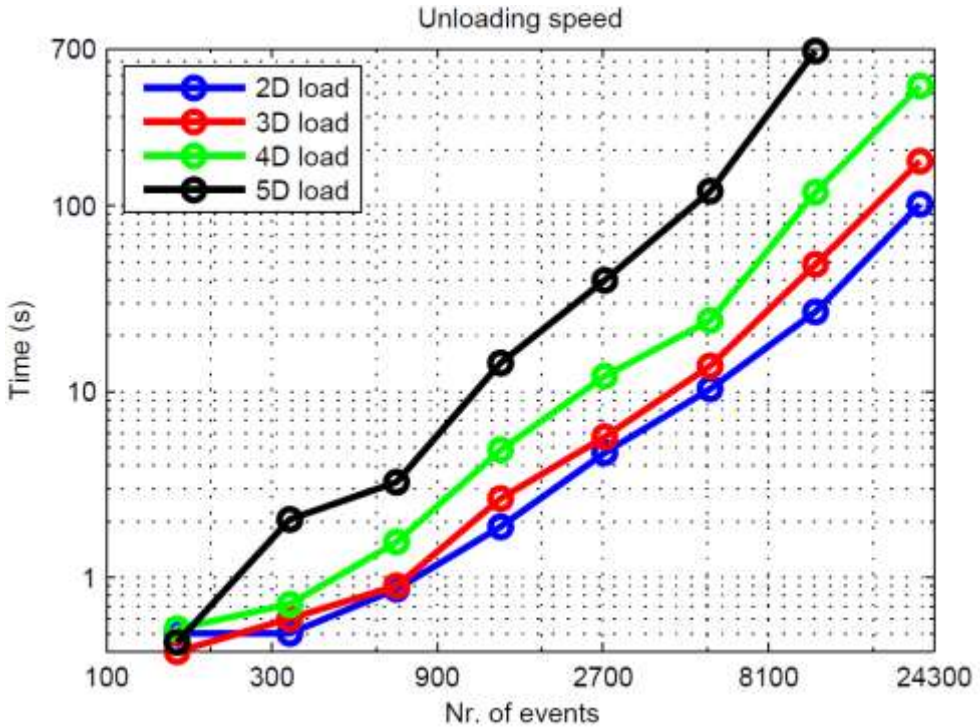
# Architecture




Log Dialog OX: All_EVENT_concept_names OY: All_TRACE_concept_names
Dashboard
Inspector
Key data
Processes 1
Cases 4
Events 33
Event classes 14
Event types 1
Events per case
Min 6 Mean 8 Max 10
Event classes per case
Min 6 Mean 8 Max 10
Log info
Start date
Tue Feb 21 10:52:13 CET 2012
End date
Tue Feb 21 10:52:13 CET 2012
Log Dialog OX: DiceOneEventLogs OY: All_TRACE_concept_names
Dashboard
Inspector
Key data
Processes 1
Cases 4
Events 14
Event classes 5
Event types 1
Events per case
Min 3 Mean 4 Max 4
Event classes per case
Min 3 Mean 4 Max 4
Log info
Start date
Tue Feb 21 10:52:13 CET 2012
End date
Tue Feb 21 10:52:13 CET 2012

# Sparsity is a problem when unloading cells to ProM

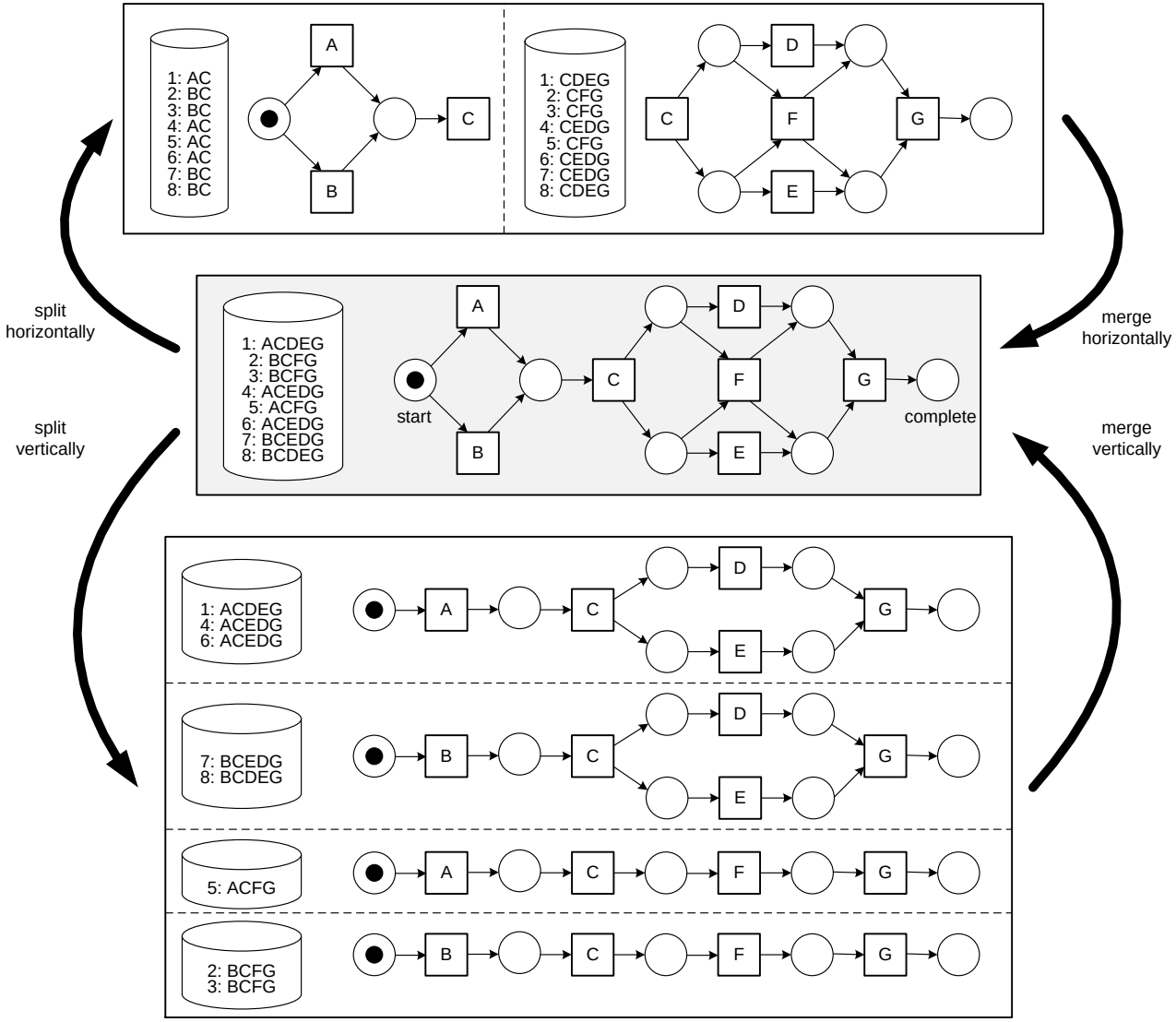


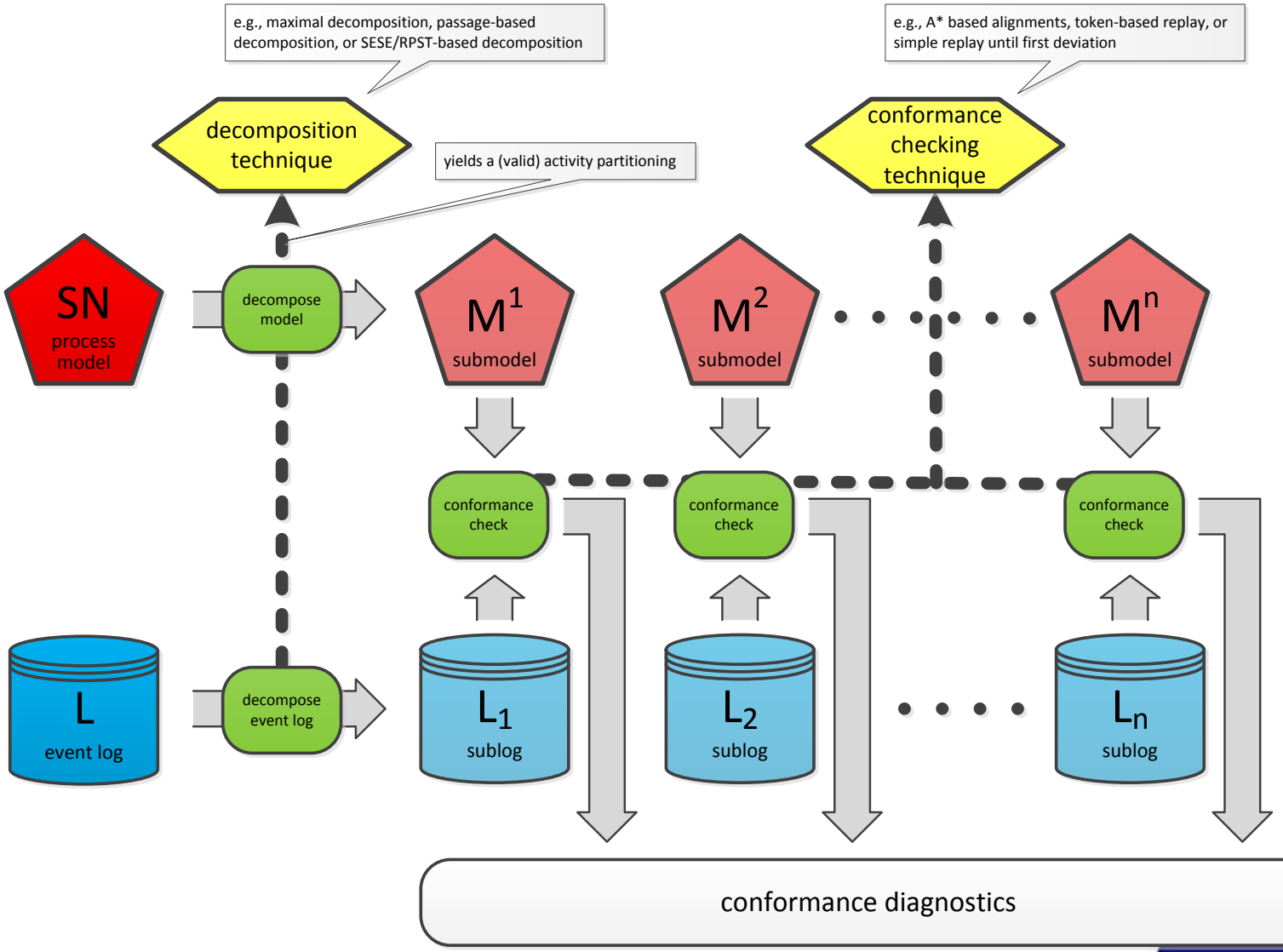

**Process cubes tend to be sparse. Palo does not handle this well.**

# Big Data: Opportunities and Challenges

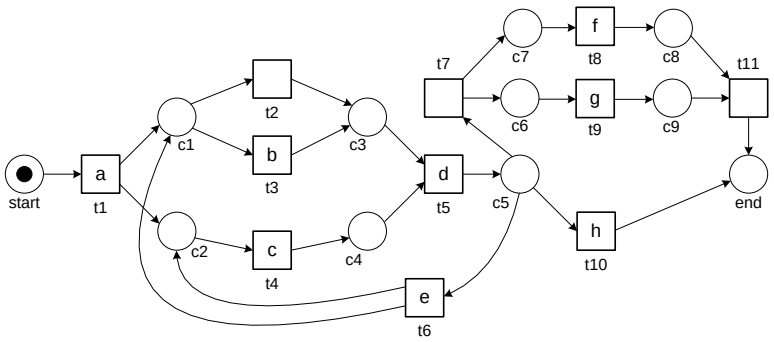
# Remember


1: AC
2: BC
3: BC
4: AC
5: AC
6: AC
7: BC
8: BC
A
B
C
1: CDEG
2: CFG
3: CFG
4: CEDG
5: CFG
6: CEDG
7: CEDG
8: CDEG
D
E
F
G
split horizontally
merge horizontally
1: ACDEG
2: BCFG
3: BCFG
4: ACEDG
5: ACFG
6: ACEDG
7: BCEDG
8: BCDEG
start
complete
split vertically
merge vertically
1: ACDEG
4: ACEDG
6: ACEDG
7: BCEDG
8: BCDEG
5: ACFG
2: BCFG
3: BCFG

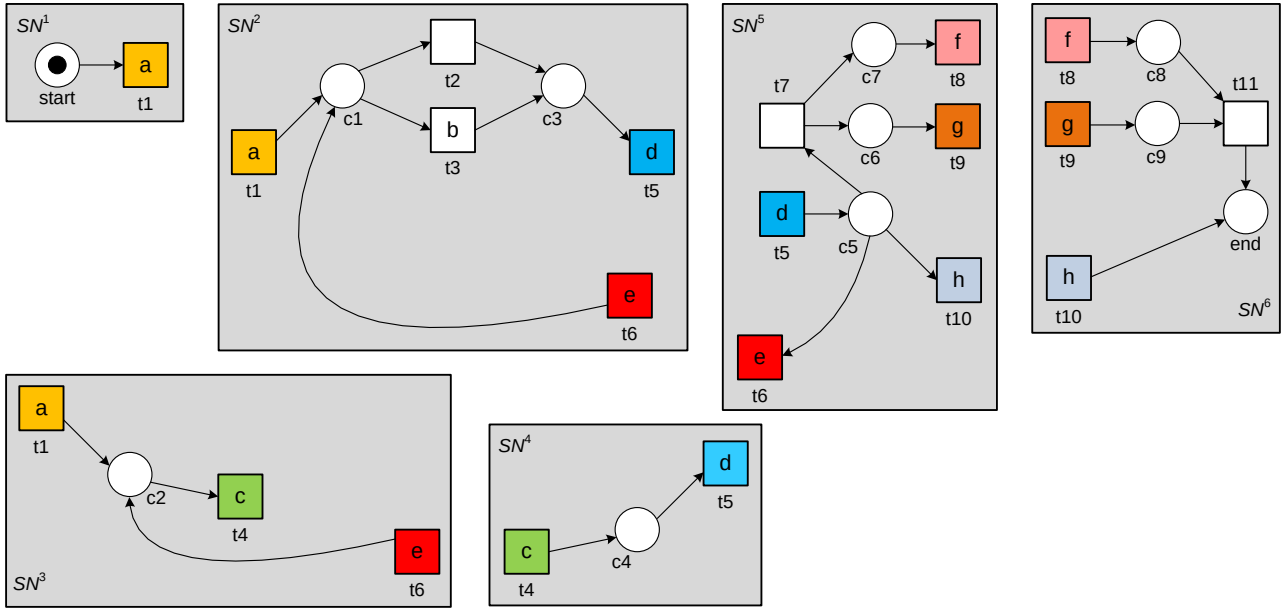
# Decomposing Conformance Checking



# Example of a valid decomposition

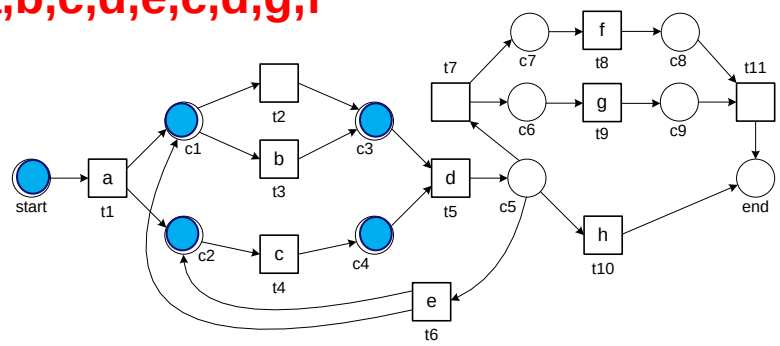


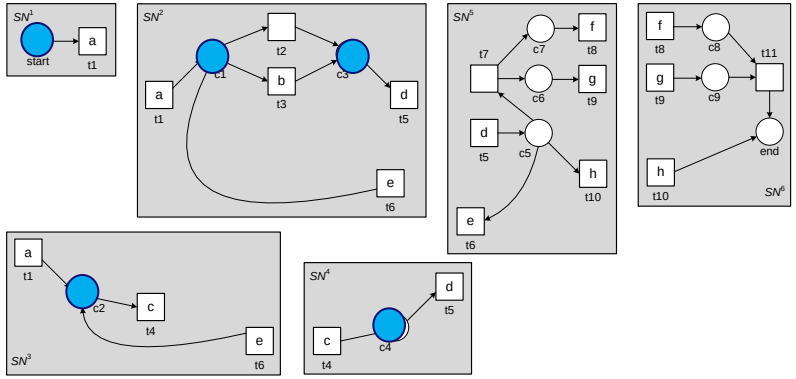

**Log can be split in the same way!**



# Example of alignment for observed trace a,b,c,d,e,c,d,g,f

**a,b,c,d,e,c,d,g,f**




$$\gamma_3 = \begin{array}{|c|c|c|c|c|c|c|c|c|c|c|c|} \hline 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 \\ \hline a & b & c & d & e & c & \gg & d & \gg & g & f & \gg \\ \hline a & b & c & d & e & c & \tau & d & \tau & g & f & \tau \\ t1 & t3 & t4 & t5 & t6 & t4 & t2 & t5 & t7 & t9 & t8 & t11 \\ \hline \end{array}$$

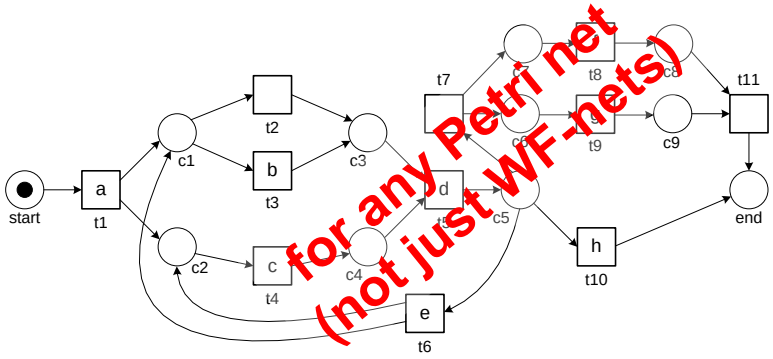
**Etc.**

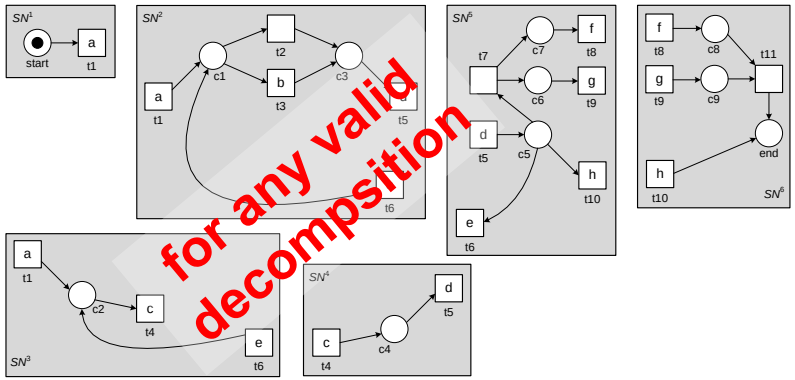
$$\gamma_3^1 = \begin{array}{|c|} \hline 1 \\ \hline a \\ \hline a \\ t1 \\ \hline \end{array} \qquad \gamma_3^2 = \begin{array}{|c|c|c|c|c|c|} \hline 1 & 2 & 4 & 5 & 7 & 8 \\ \hline a & b & d & e & \gg & d \\ \hline a & b & d & e & \tau & d \\ t1 & t3 & t5 & t6 & t2 & t5 \\ \hline \end{array} \qquad \gamma_3^3 = \begin{array}{|c|c|c|c|} \hline 1 & 3 & 5 & 6 \\ \hline a & c & e & c \\ \hline a & c & e & c \\ t1 & t4 & t6 & t4 \\ \hline \end{array}$$

$$\gamma_3^4 = \begin{array}{|c|c|c|c|} \hline 3 & 4 & 6 & 8 \\ \hline c & d & c & d \\ \hline c & d & c & d \\ t4 & t5 & t4 & t5 \\ \hline \end{array} \qquad \gamma_3^5 = \begin{array}{|c|c|c|c|c|c|} \hline 4 & 5 & 8 & 9 & 10 & 11 \\ \hline d & e & d & \gg & g & f \\ \hline d & e & d & \tau & g & f \\ t5 & t6 & t5 & t7 & t9 & t8 \\ \hline \end{array} \qquad \gamma_3^6 = \begin{array}{|c|c|c|} \hline 10 & 11 & 12 \\ \hline g & f & \gg \\ \hline g & f & \tau \\ t9 & t8 & t11 \\ \hline \end{array}$$

# Conformance checking can be decomposed !!!

- **General result for any valid decomposition: Any event log or trace is perfectly fitting the overall model if and only if it is also fitting all the individual fragments**

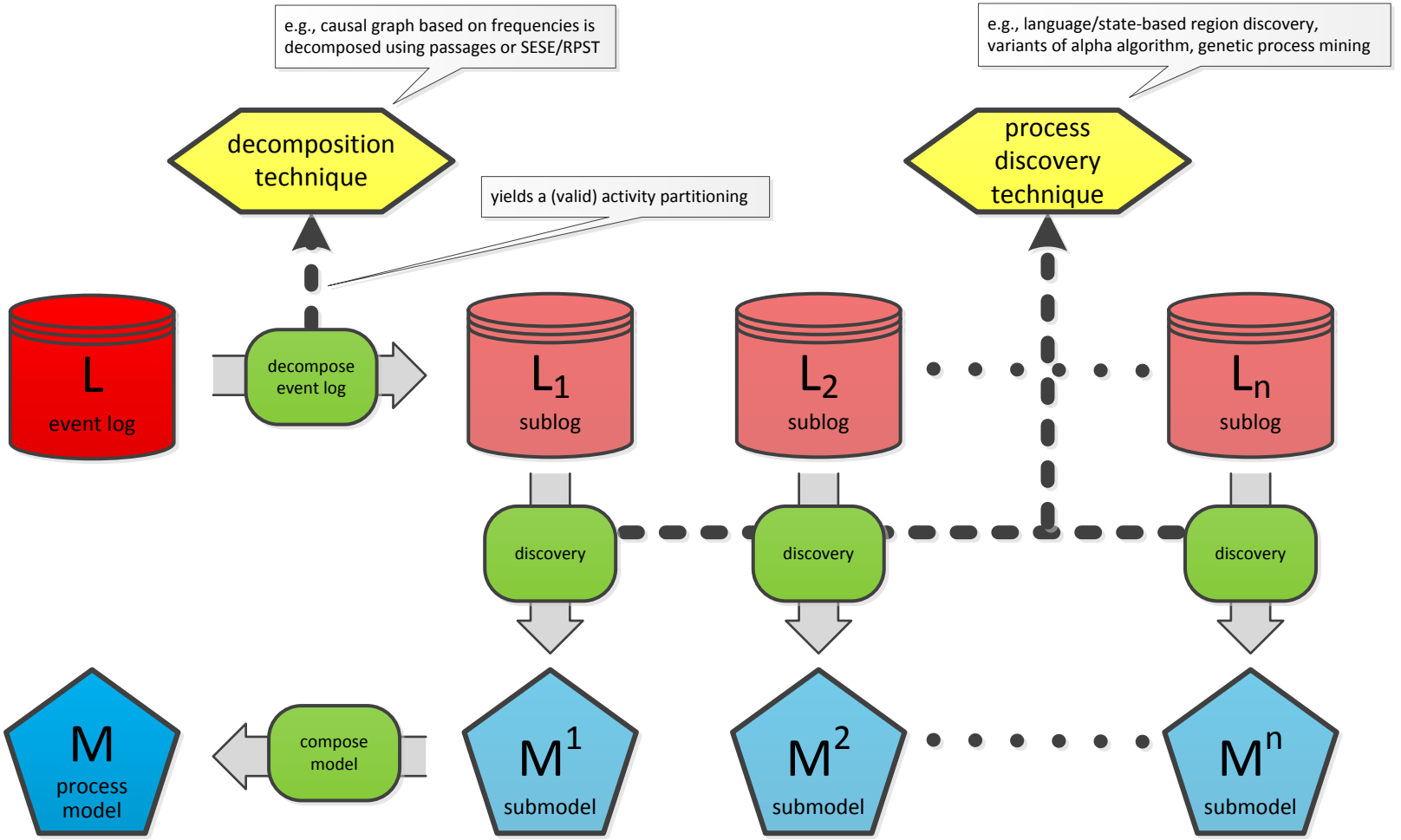




Wil van der Aalst, Decomposing Petri nets for process mining: A generic approach. Distributed and Parallel Databases, Volume 31, Issue 4, pp 471-507, 2013

# Decomposing Process Discovery



# Learn more about decomposing process mining problems?

- **W.M.P. van der Aalst. Decomposing Petri Nets for Process Mining: A Generic Approach. *Distributed and Parallel Databases,* 31(4):471-507, 2013.**


- W.M.P. van der Aalst. A General Divide and Conquer Approach for Process Mining. In M. Ganzha, L. Maciaszek, and M. Paprzycki, editors,*Federated Conference on Computer Science and Information Systems (FedCSIS 2013)*, pages 1-10. IEEE Computer Society, 2013.
- W.M.P. van der Aalst. Decomposing Process Mining Problems Using Passages. In S. Haddad and L. Pomello, editors, *Applications and Theory of Petri Nets 2012*, volume 7347 of *Lecture Notes in Computer Science*, pages 72-91. Springer-Verlag, Berlin, 2012.
- J. Munoz-Gama, J. Carmona, and W.M.P. van der Aalst. Hierarchical Conformance Checking of Process Models Based on Event Logs. In J.M. Colom and J. Desel, editors, *Applications and Theory of Petri Nets 2013*, volume 7927 of *Lecture Notes in Computer Science*, pages 291-310. Springer-Verlag, Berlin, 2013.
- J. Munoz-Gama, J. Carmona, and W.M.P. van der Aalst. Conformance Checking in the Large: Partitioning and Topology. In F. Daniel, J. Wang, and B. Weber, editors, *International Conference on Business Process Management (BPM 2013)*, volume 8094 of *Lecture Notes in Computer Science*, pages 130-145. Springer-Verlag, Berlin, 2013.
- E. Verbeek and W.M.P. van der Aalst. Decomposing Replay Problems: A Case Study. In D. Moldt and H. Roelke, editors, *Proceedings of the International Workshop on Petri Nets in Software Engineering (PNSE 2013)*, volume 989 of *CEUR Workshop Proceedings*, pages 213-232. CEUR-WS.org, 2013.

# Conclusion

- **Challenges in process mining: concept drift, comparing processes, cases, and organizations, need for contextual analysis, limited scalability, performance problems.**
- **Unifying view: Process Cubes.**
- **Similar to OLAP but also many differences.**
- **Related to decomposing process mining problems.**
- **Initial implementations, but just the starting point.**

Wil M. P. van der Aalst

# Process Mining

## Discovery, Conformance and Enhancement of Business Processes

More and more information about business processes is recorded by information systems in the form of so-called "event logs". Despite the omnipresence of such data, most organizations diagnose problems based on fiction rather than facts. Process mining is an emerging discipline based on process model-driven approaches and data mining. It not only allows organizations to fully benefit from the information stored in their systems, but it can also be used to check the conformance of processes, detect bottlenecks, and predict execution problems.

Wil van der Aalst delivers the first book on process mining. It aims to be self-contained while covering the entire process mining spectrum from process discovery to operational support. In Part I, the author provides the basics of business process modeling and data mining necessary to understand the remainder of the book. Part II focuses on process discovery as the most important process mining task. Part III moves beyond discovering the control flow of processes and highlights conformance checking, and organizational and time perspectives. Part IV guides the reader in successfully applying process mining in practice, including an introduction to the widely used open-source tool ProM. Finally, Part V takes a step back, reflecting on the material presented and the key open challenges.

Overall, this book provides a comprehensive overview of the state of the art in process mining. It is intended for business process analysts, business consultants, process managers, graduate students, and BPM researchers.

Features and Benefits:

- First book on process mining, bridging the gap between business process modeling and business intelligence.
- Written by one of the most influential and most-cited computer scientists and the best-known BPM researcher.
- Self-contained and comprehensive overview for a broad audience in academia and industry.
- The reader can put process mining into practice immediately due to the applicability of the techniques and the availability of the open-source process mining software ProM.

Computer Science

ISBN 978-3-642-19344-6

springer.com

van der Aalst Process Mining

ProM
process mining workbench

Wil M. P. van der Aalst

Process Mining

Discovery, Conformance and Enhancement of Business Processes

**www.processmining.org**

**www.win.tue.nl/ieeetfpm/**

Springer