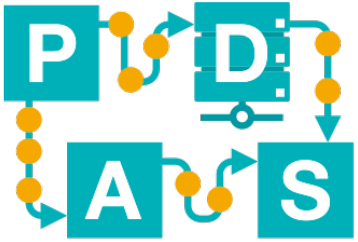
Conformance Checking (1/2)

Lecture 12

prof.dr.ir. Wil van der Aalst
www.vdaalst.com @wvdaalst
www.pads.rwth-aachen.de

BPI-L12

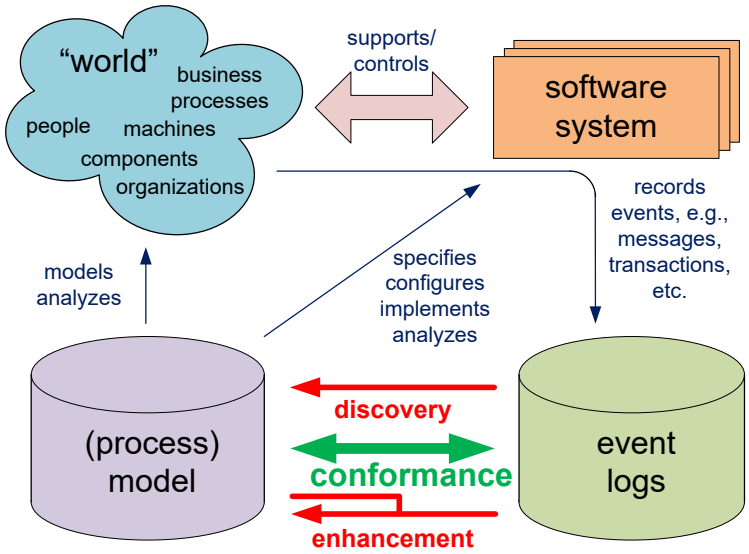


Chair of Process and Data Science

RWTH AACHEN UNIVERSITY

Conformance checking approaches

1. Conformance checking using causal footprints.
2. Conformance checking based on token-based replay.
3. Alignment-based conformance checking.

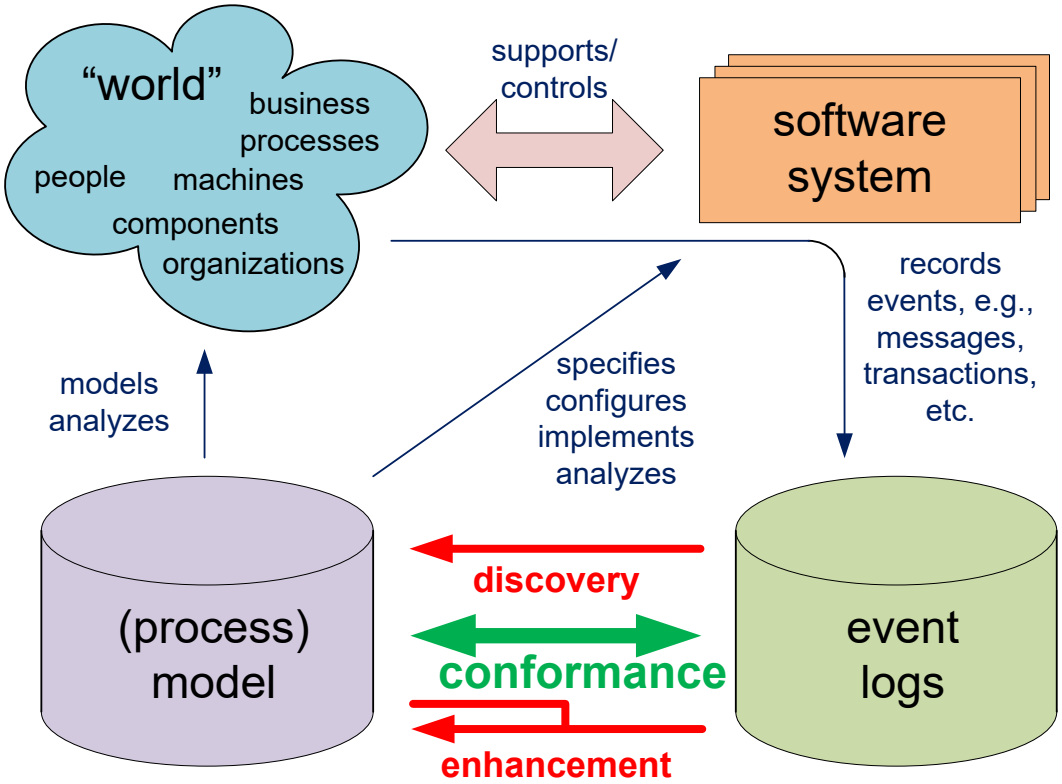


Chair of Process and Data Science

RWTH AACHEN UNIVERSITY

Conformance Checking

Conformance checking



Three main use cases

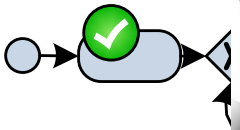
- **Compliance checking (for auditing, fraud detection, etc.)**
  - Audits are performed to ascertain the validity and reliability of information about organizations and their associated processes.
  - This is done to check whether business processes are executed within certain boundaries set by managers, governments, and other stakeholders.

- **Evaluating process discovery results / algorithms**
  - Comparing discovered process models with the data used to learn the model or with unseen test data.
  - Evaluating a model or an algorithm (see k-fold cross validation).
- **Conformance to specification (software, services, etc.).**

Positive or negative deviants?

“Breaking the glass” may save lives!

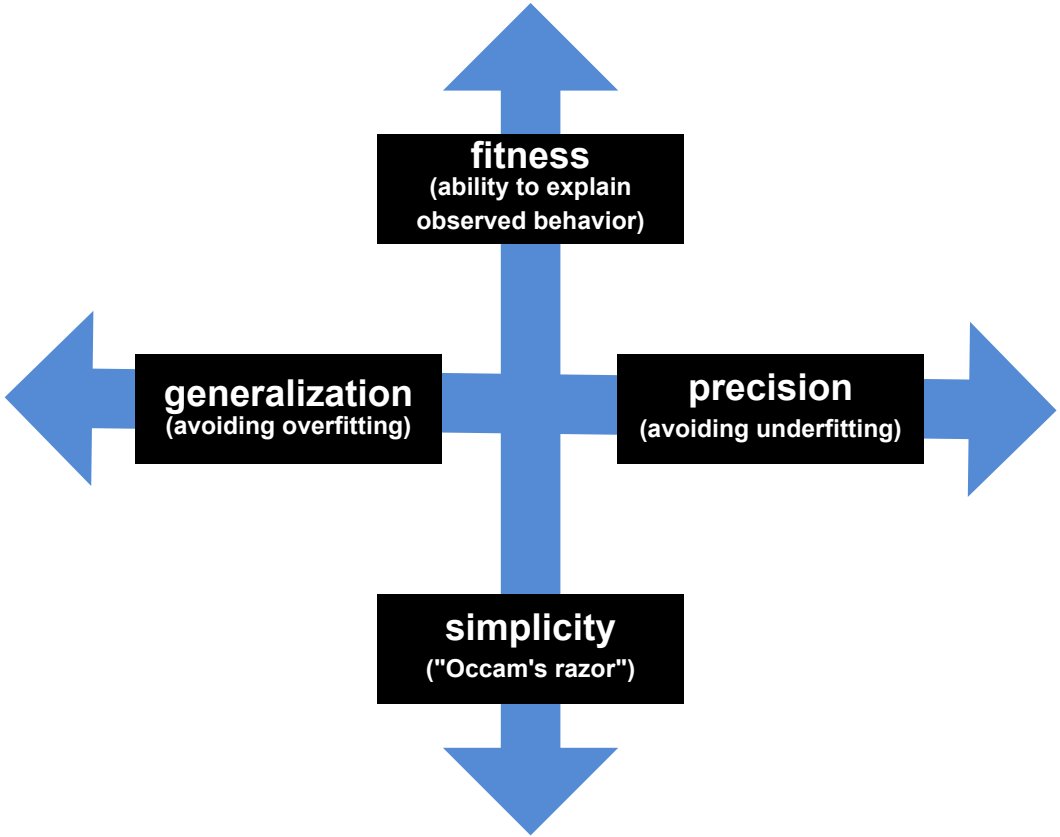
event log



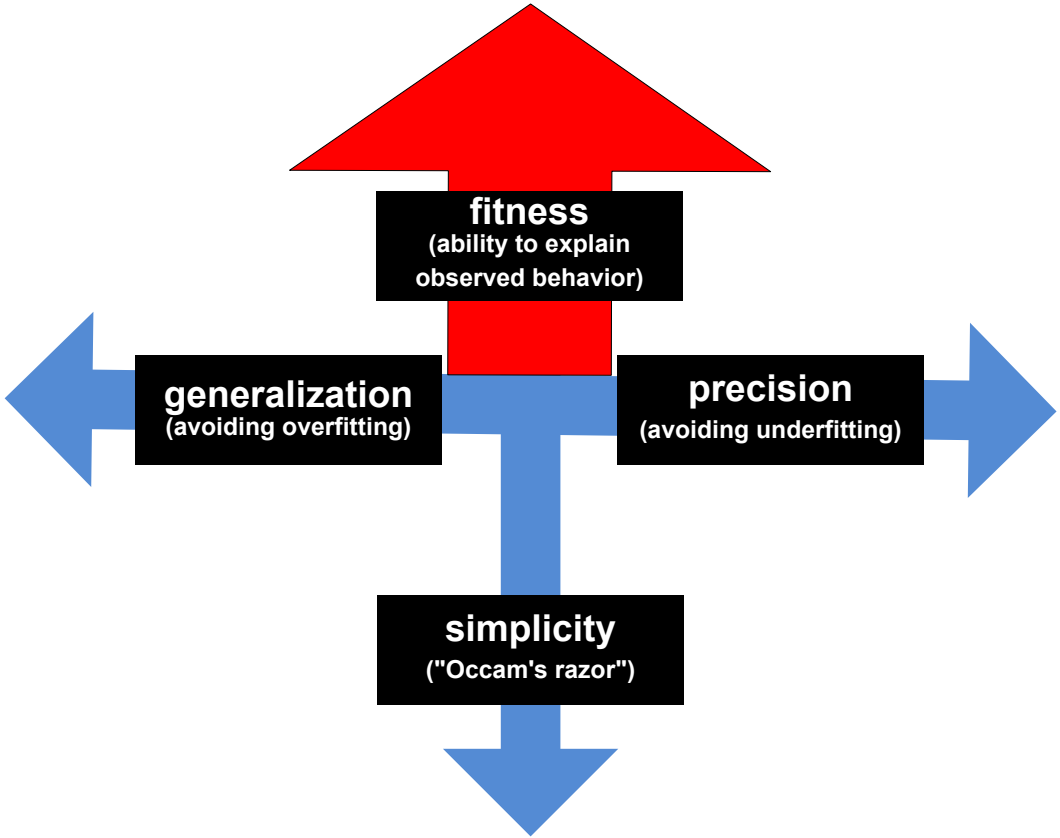
Is the model wrong or is t

learning from positive deviants

Four dimensions to compare log and model



Replay fitness is dominant

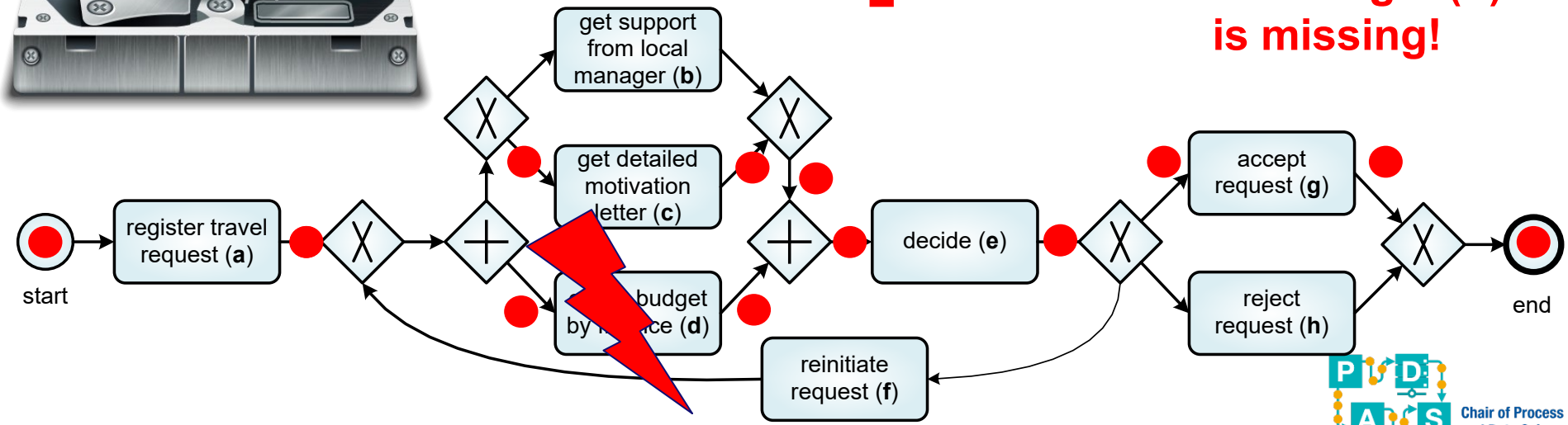


Replay example used before

a c e g

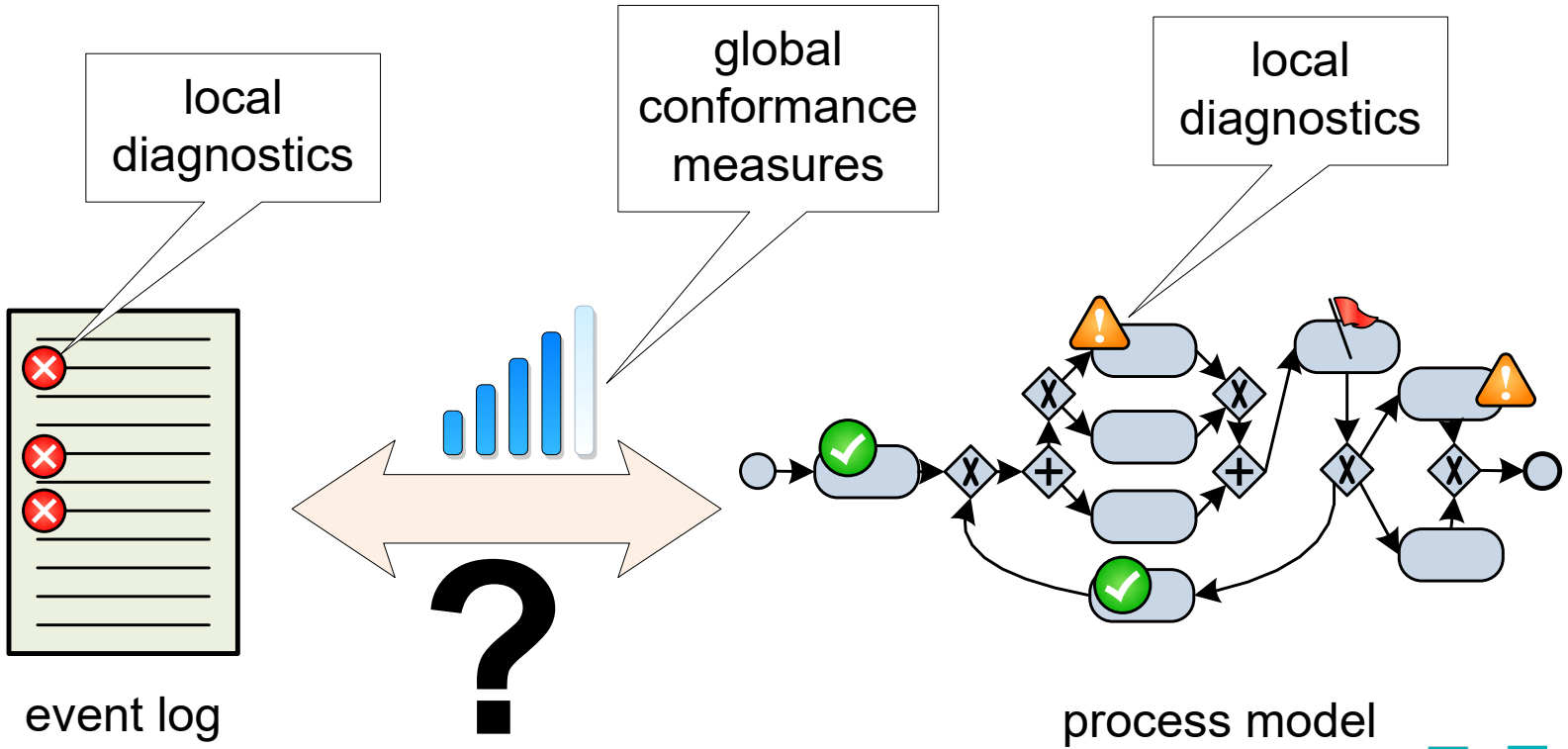
?

check budget (d) is missing!



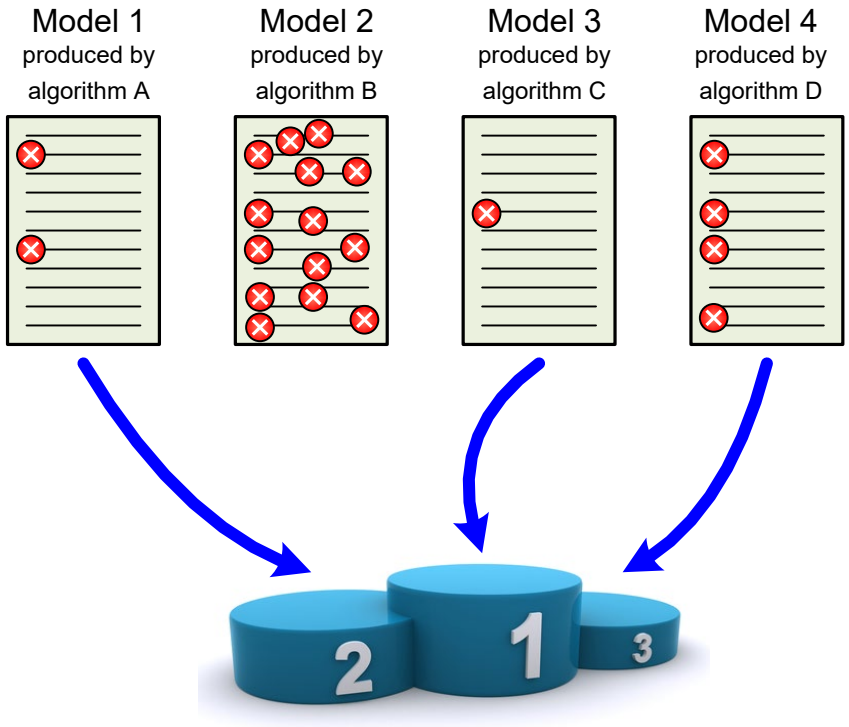
Chair of Process and Data Science

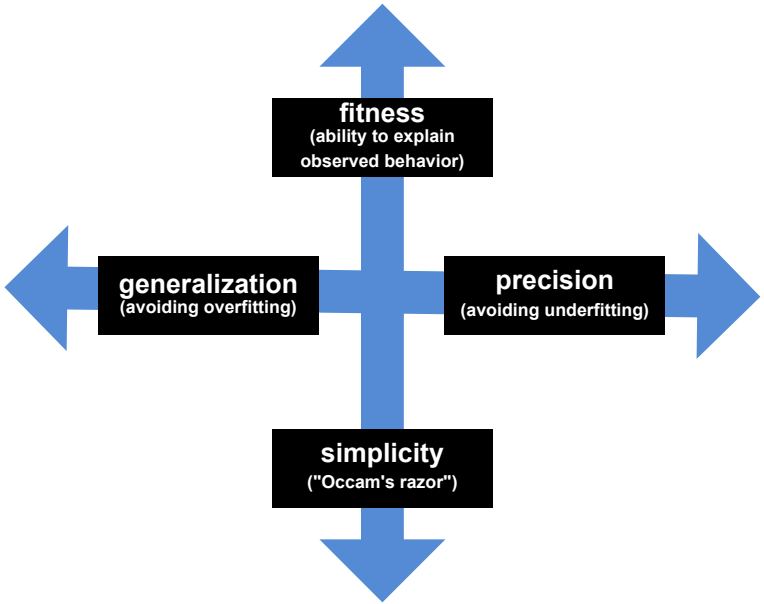
Conformance diagnostics and measures



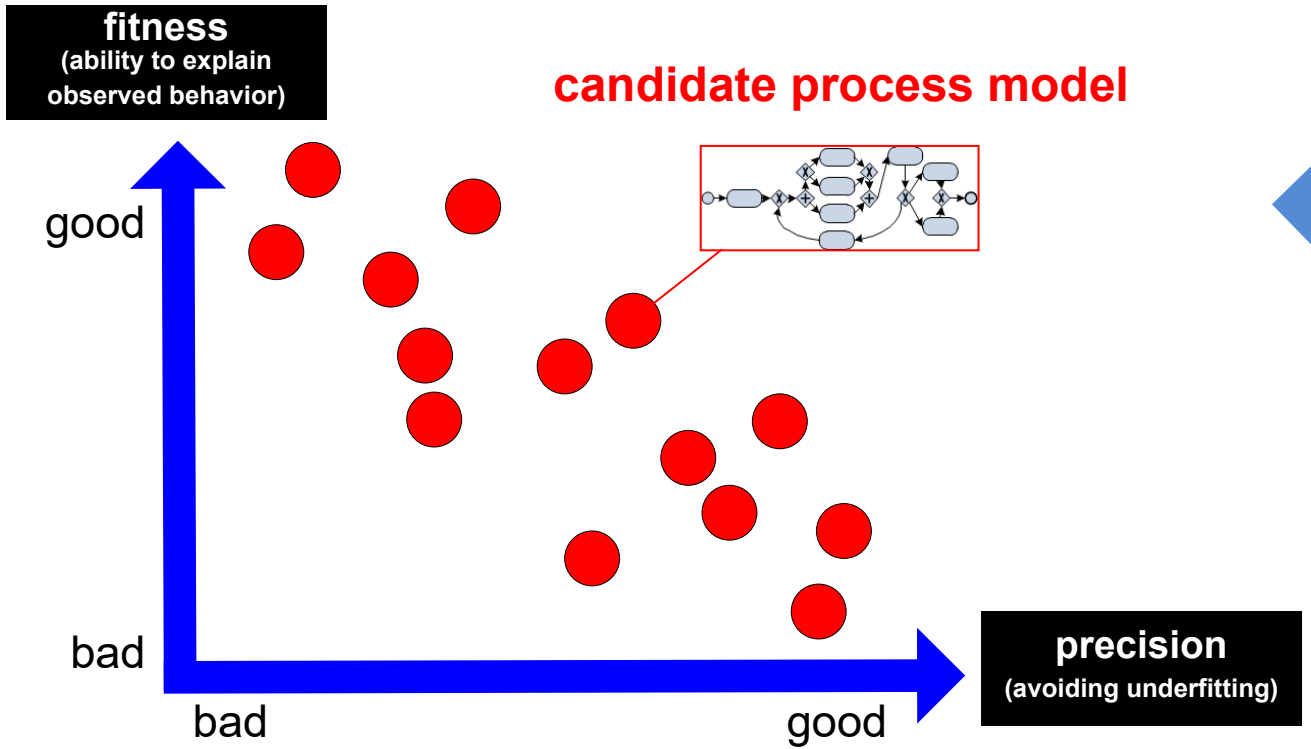
Chair of Process and Data Science

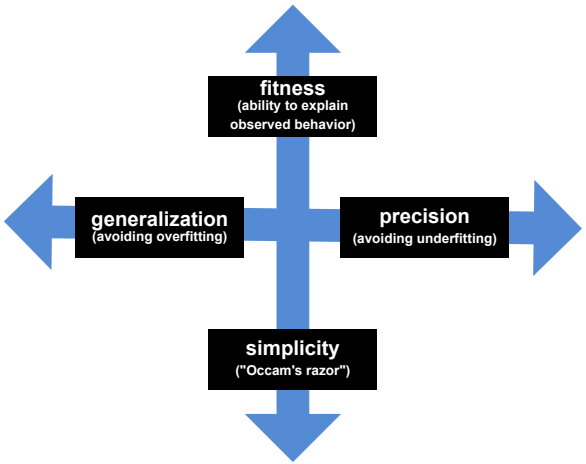
Evaluating process discovery algorithms





Pareto front considering two dimensions

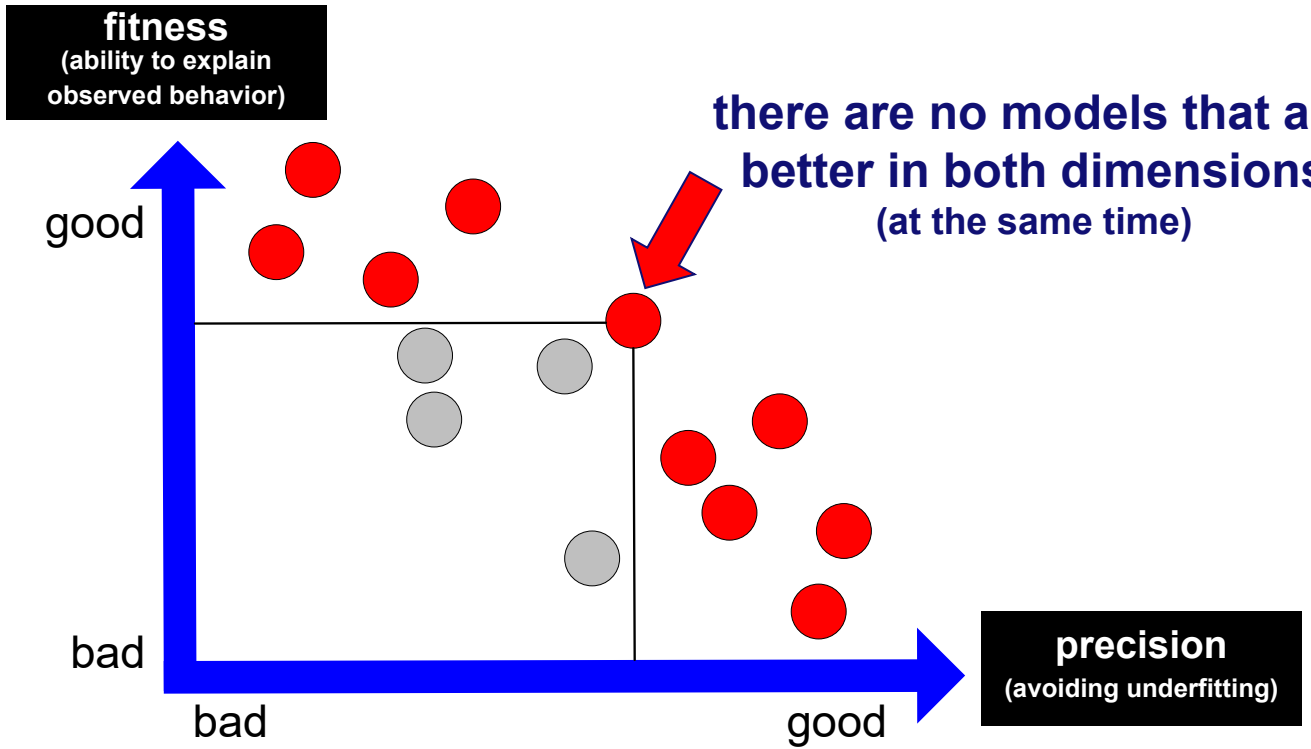




Chair of Process and Data Science

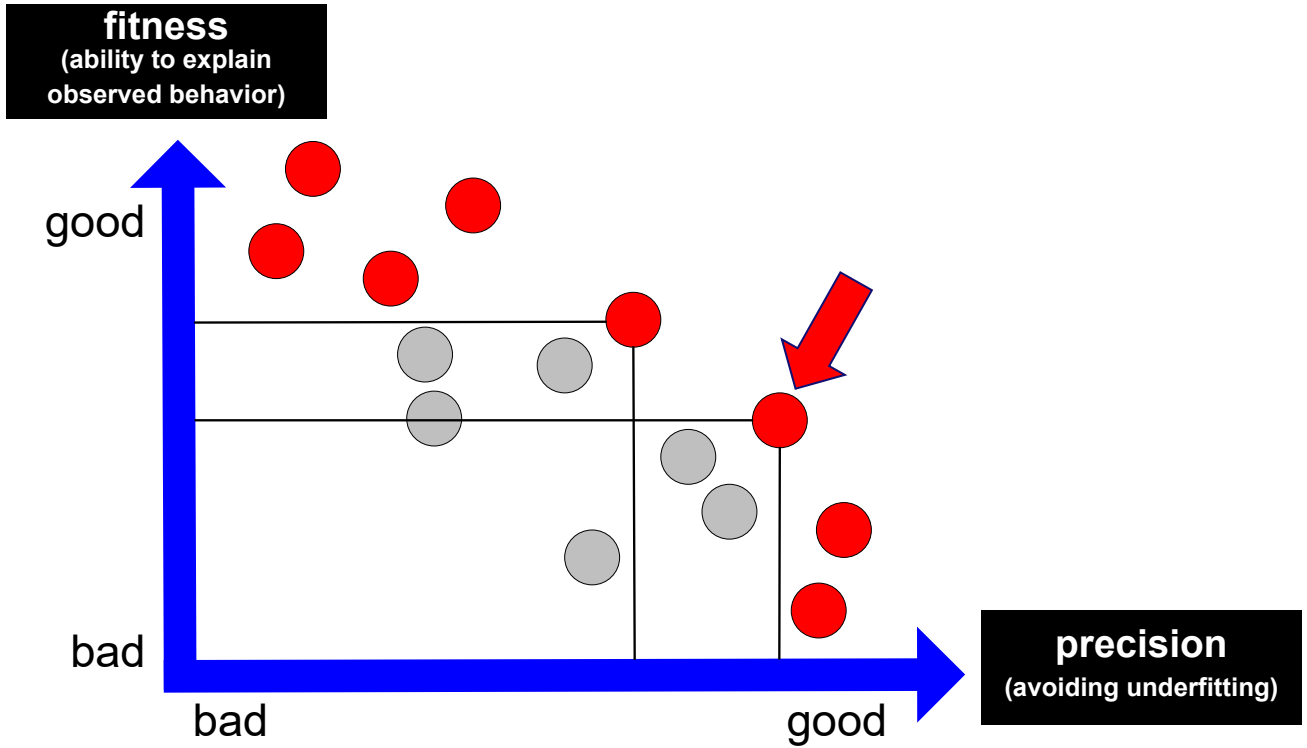
Pareto front considering two dimensions

Pareto efficiency/
Pareto optimality



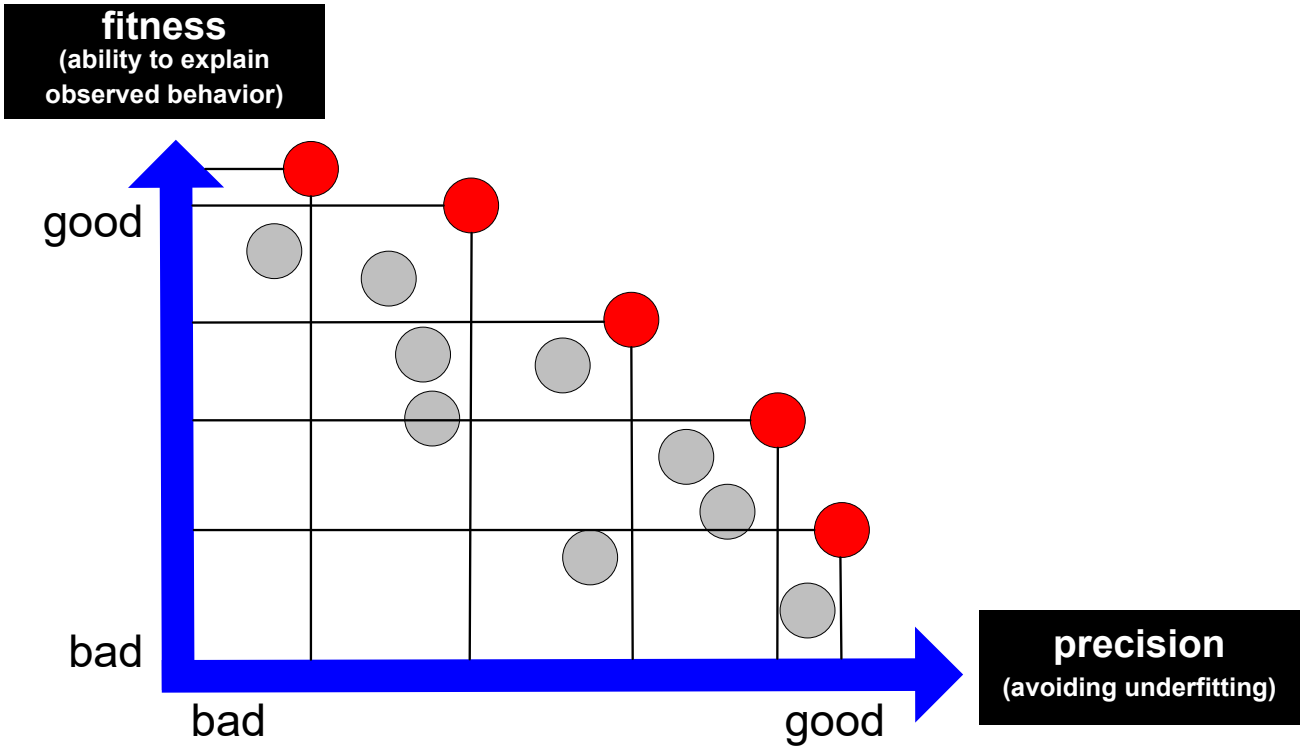
Chair of Process and Data Science

Pareto front considering two dimensions



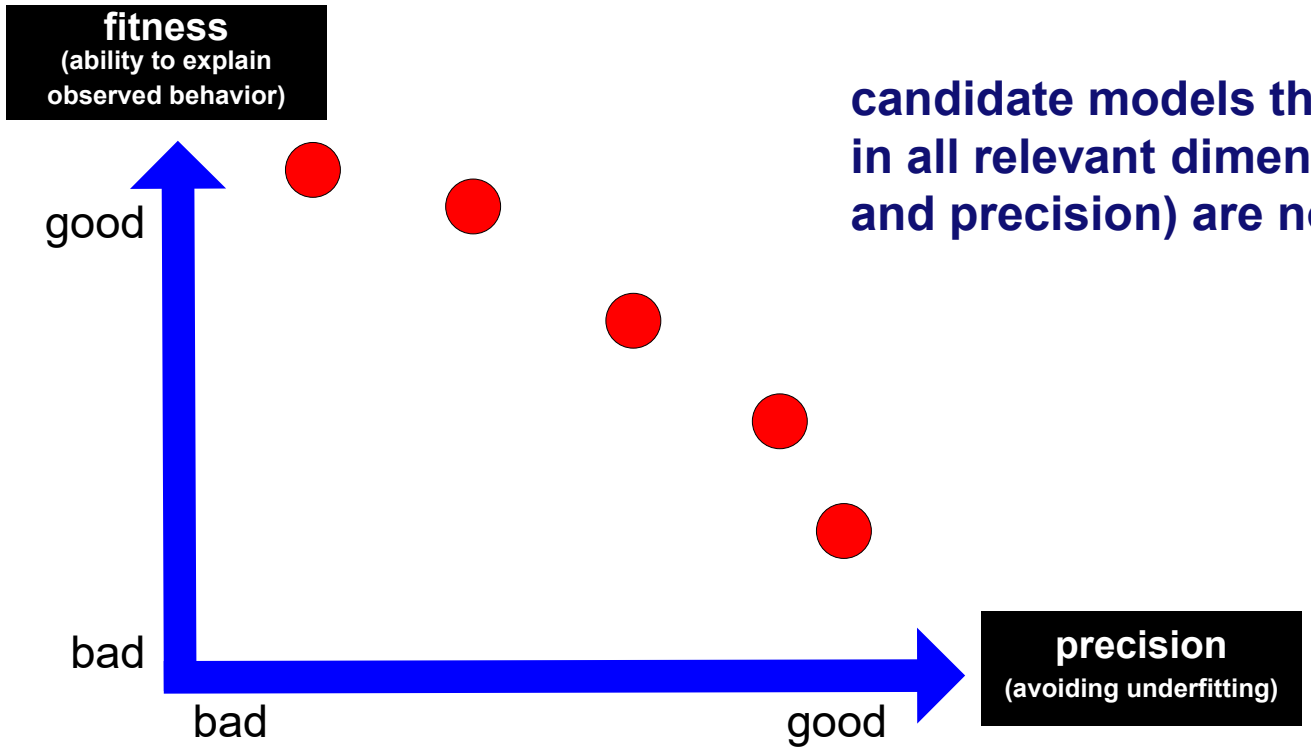
Chair of Process and Data Science

Pareto front considering two dimensions



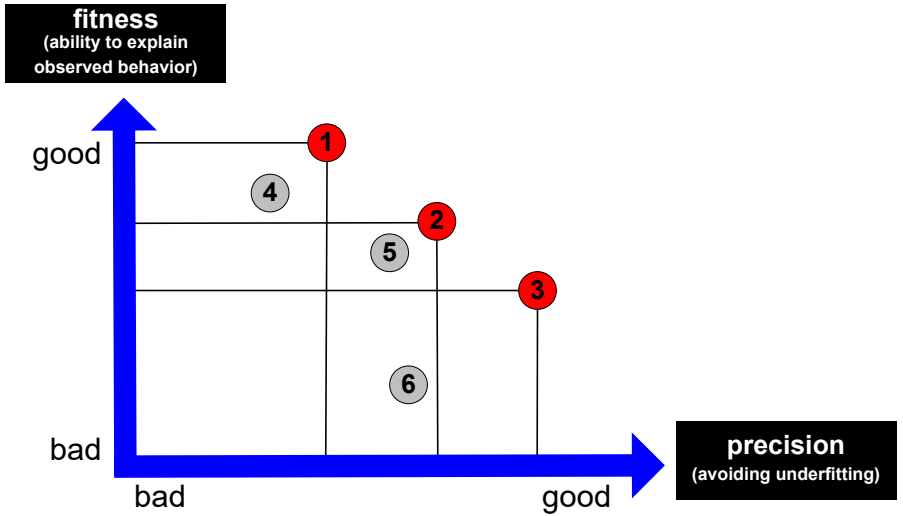
Chair of Process and Data Science

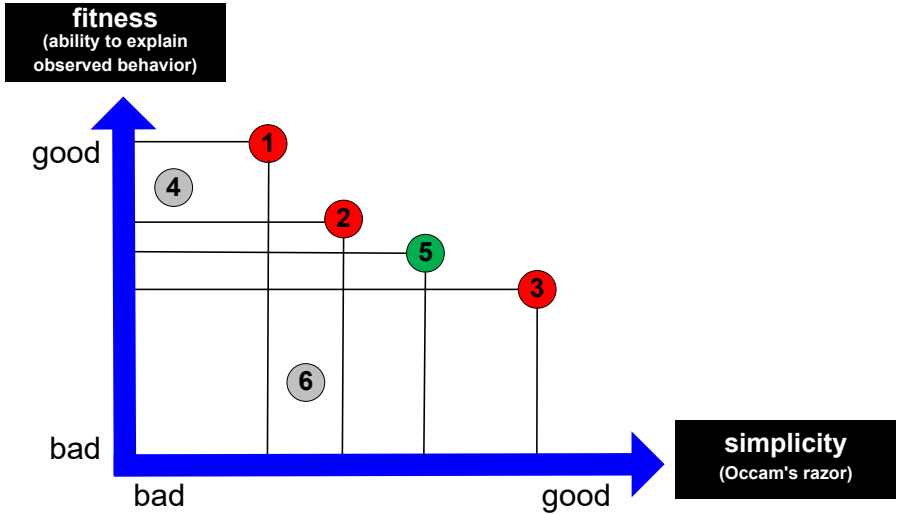
Pareto front considering two dimensions



candidate models that are dominated in all relevant dimensions (here fitness and precision) are not interesting

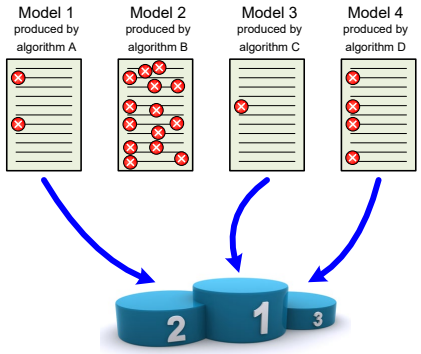
Adding another dimension: Simplicity

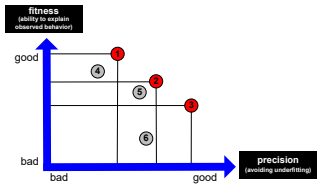


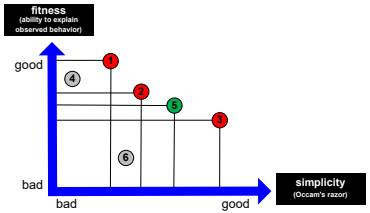


There is no model that (at the same time) has a better fitness, precision, and simplicity than model 5, i.e., it is not dominated.

Comparing discovered models is not easy



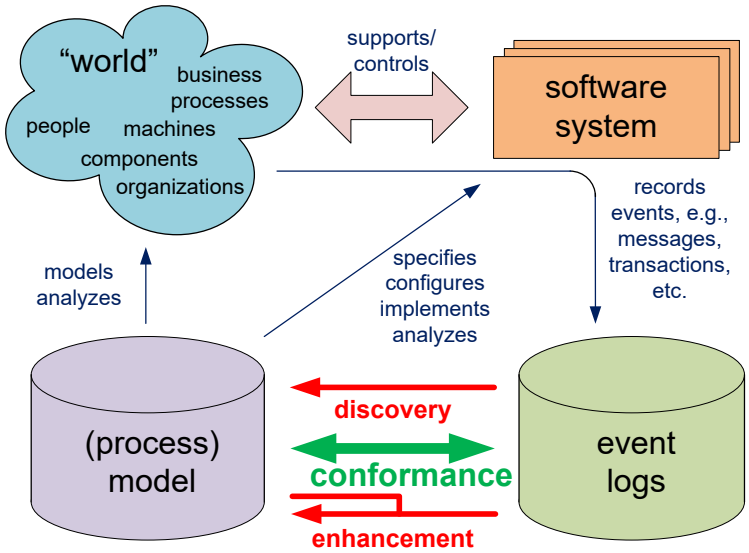




- There is no such thing as "the best model“.
- Moreover, models may serve different purposes (see maps).
- Initially, we focus on quantifying (replay) fitness (a.k.a. recall).

Conformance checking approaches

1. Conformance checking using causal footprints.
2. Conformance checking based on token-based replay.
3. Alignment-based conformance checking.



Chair of Process and Data Science

Focusing just on control-flow

- **An event log is a finite multiset of traces:**
  $$L = [\langle a, b, c, d \rangle^6, \langle a, b, c, d \rangle^4]$$
- **A model describes a (possibly infinite) set of traces:**
  $$M = \{\langle a, b, c, d \rangle, \langle a, b, c, d \rangle\}$$
- **As indicated before: Simple precision and recall measures do not work:**
  - **Loops lead to a recall of 0**
  - **Traces may be almost fitting**
  - **Log is only a sample and typically very incomplete**
- **Idea: Create a common finite abstraction, e.g., a DFG or causal footprint (next), dealing with incompleteness and loops.**

Causal footprints

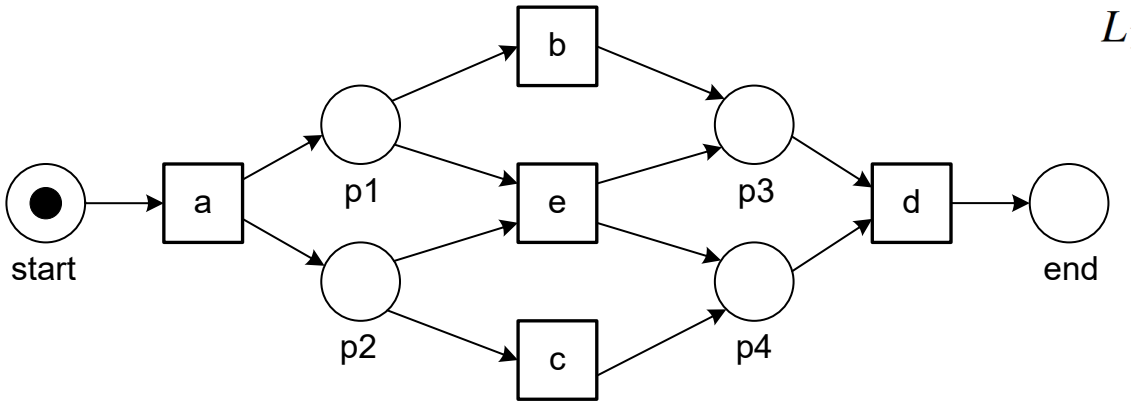
Footprint of L_1

$$L_1 = [\langle a,b,c,d\rangle^3, \langle a,c,b,d\rangle^2, \langle a,e,d\rangle]$$

	a	b	c	d	e
a	$\#_{L_1}$	$\rightarrow_{L_1}$	$\rightarrow_{L_1}$	$\#_{L_1}$	$\rightarrow_{L_1}$
b	$\leftarrow_{L_1}$	$\#_{L_1}$	$\|\|_{L_1}$	$\rightarrow_{L_1}$	$\#_{L_1}$
c	$\leftarrow_{L_1}$	$\|\|_{L_1}$	$\#_{L_1}$	$\rightarrow_{L_1}$	$\#_{L_1}$
		$\leftarrow_{L_1}$	$\leftarrow_{L_1}$	$\#_{L_1}$	$\leftarrow_{L_1}$
		$\#_{L_1}$	$\#_{L_1}$	$\rightarrow_{L_1}$	$\#_{L_1}$

- 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.**

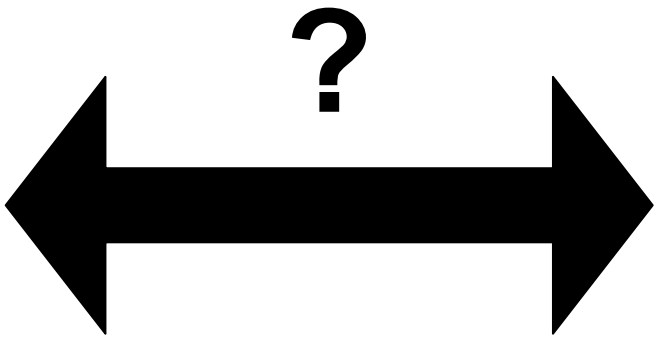
Discovered model has the same footprint

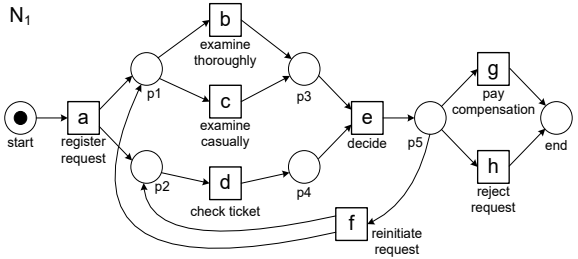


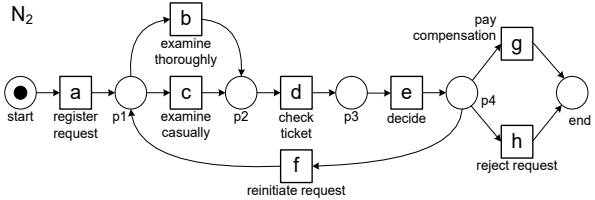
$$L_1 = [\langle a,b,c,d\rangle^3, \langle a,c,b,d\rangle^2, \langle a,e,d\rangle]$$

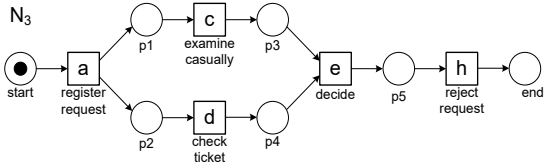
	a	b	c	d	e
a	$\#_{L_1}$	$\rightarrow_{L_1}$	$\rightarrow_{L_1}$	$\#_{L_1}$	$\rightarrow_{L_1}$
b	$\leftarrow_{L_1}$	$\#_{L_1}$	$\|\|_{L_1}$	$\rightarrow_{L_1}$	$\#_{L_1}$
c	$\leftarrow_{L_1}$	$\|\|_{L_1}$	$\#_{L_1}$	$\rightarrow_{L_1}$	$\#_{L_1}$
d	$\#_{L_1}$	$\leftarrow_{L_1}$	$\leftarrow_{L_1}$	$\#_{L_1}$	$\leftarrow_{L_1}$
e	$\leftarrow_{L_1}$	$\#_{L_1}$	$\#_{L_1}$	$\rightarrow_{L_1}$	$\#_{L_1}$

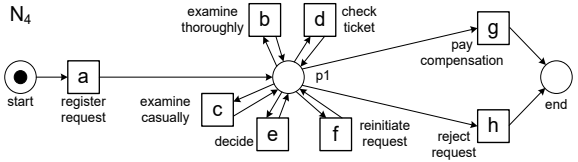
#	trace
455	acdeh
191	abdeg
177	adceh
144	abdeh
111	acdeg
82	adceg
56	adbeh
47	acdefdbeh
38	adbeg
33	acdefbdeh
14	acdefbdeg
11	acdefdbeg
9	adcefcdeh
8	adcefdbeh
5	adcefbdeg
3	acdefbdefdbeg
2	adcefdbeg
2	adcefbdefbdeg
1	adcefdbefbdeh
1	adbefbdefdbeg
1	adcefdbefcdefdbeg
1391	





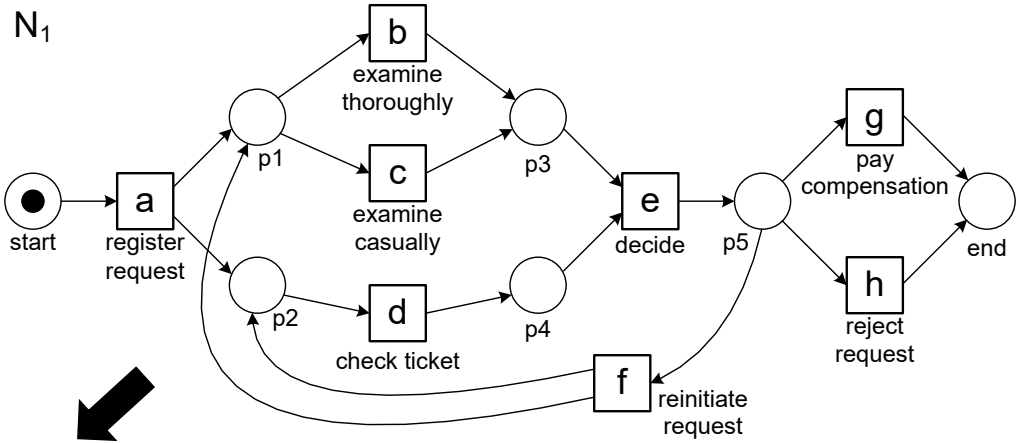






#	trace
455	acdeh
191	abdeg
177	adceh
144	abdeh
111	acdeg
82	adceg
56	adbeh
47	acdefdbeh
38	adbeg
33	acdefbdeh
14	acdefbdeg
11	acdefdbeg
9	adcefcdeh
8	adcefdbeh
5	adcefbdeg
3	acdefbdefdbeg
2	adcefdbeg
2	adcefbdefbdeg
1	adcefdbefbdeh
1	adbefbdefdbeg
1	adcefdbefcdefdbeg
1391	

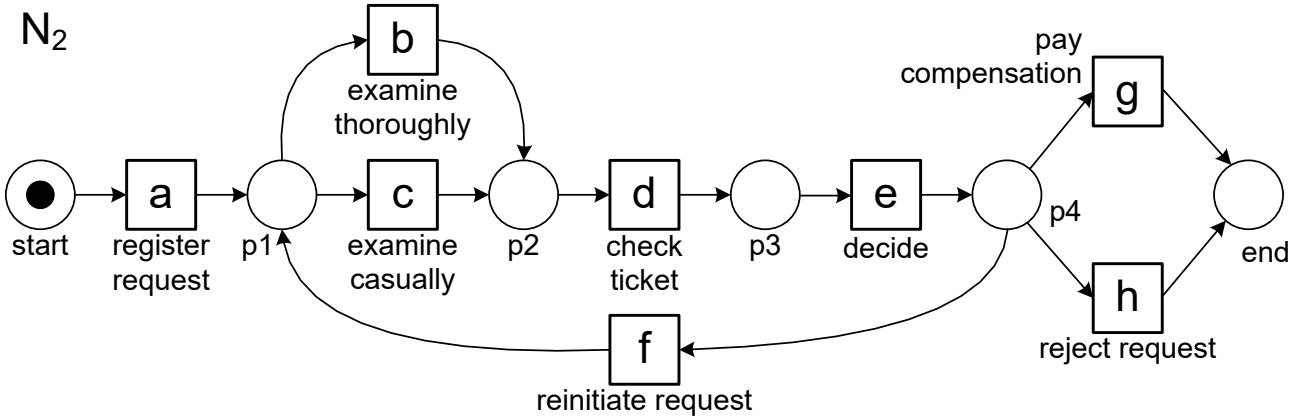
L_{full} and N_1



	a	b	c	d	e	f	g	h
a	#	→						#
b	←	#						#
c	←	#						#
d	←	‖	‖	#	→	←	#	#
e	#	←	←	←	#	→	→	→
f	#	→	→	→	←	#	#	#
g	#	→	→	→	←	#	#	#
h	#	#	#	#	←	#	#	#

footprint-based conformance = 1
(1 = perfect match)
(0 = worst match possible)

footprints of log and model coincide



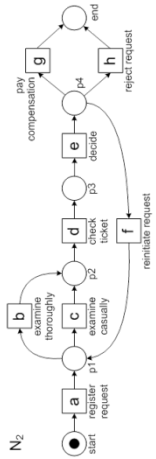
	a	b	c	d	e	f	g	h
a	#	$\rightarrow$	$\rightarrow$	#	#	#	#	#
b	$\leftarrow$	#	#	$\rightarrow$	#	$\leftarrow$	#	#
c	$\leftarrow$	#	#	$\rightarrow$	#	$\leftarrow$	#	#
d	#	$\leftarrow$	$\leftarrow$	#	$\rightarrow$	#	#	#
e	#	#	#	$\leftarrow$	#	$\rightarrow$	$\rightarrow$	$\rightarrow$
f	#	$\rightarrow$	$\rightarrow$	#	$\leftarrow$	#	#	#
g	#	#	#	#	$\leftarrow$	#	#	#
h	#	#	#	#	$\leftarrow$	#	#	#

L_{full}

#	trace
455	acdeh
191	abdeg
177	adceh
144	abdeh
111	acdeg
82	adceg
56	adbeh
47	acdefdbeh
38	adbeg
33	acdefbdeh
14	acdefbdeg
11	acdefdbeg
9	adcefcdeh
8	adcefdbeh
5	adcefbdeg
3	acdefbdefdbeg
2	adcefdbeg
2	adcefbdefbdeg
1	adcefdbefbdeh
1	adbefbdefdbeg
1	adcefdbefcdefdbeg
1391	

	a	b	c	d	e	f	g	h
a	#	→	→	→	#	#	#	#
b	←	#	#	‖	→	←	#	#
c	←	#	#	‖	→	←	#	#
d	←	‖	‖	#	→	←	#	#
e	#	←	←	←	#	→	→	→
f	#	→	→	→	←	#	#	#
g	#	#	#	#	←	#	#	#
h	#	#	#	#	←	#	#	#

N_2



	a	b	c	d	e	f	g	h
a	#	→	→	#	#	#	#	#
b	←	#	#	→	#	←	#	#
c	←	#	#	→	#	←	#	#
d	#	←	←	#	→	#	#	#
e	#	#	#	←	#	→	→	→
f	#	→	→	#	←	#	#	#
g	#	#	#	#	←	#	#	#
h	#	#	#	#	←	#	#	#

Quantifying the differences

	a	b	c	d	e	f	g	h
a				$\rightarrow: \#$				
b				$\parallel: \rightarrow$	$\rightarrow: \#$			
c				$\parallel: \rightarrow$	$\rightarrow: \#$			
d	$\leftarrow: \#$	$\parallel: \leftarrow$	$\parallel: \leftarrow$			$\leftarrow: \#$		
e		$\leftarrow: \#$	$\leftarrow: \#$					
f				$\rightarrow: \#$				
g								
h								

$$1 - \frac{12}{64} = 0.8125$$

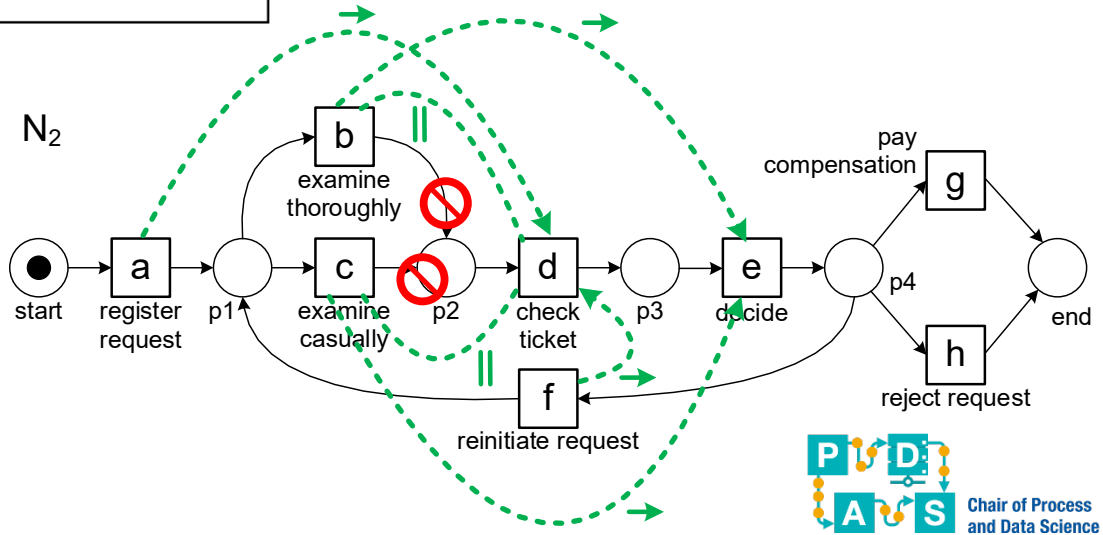
(x:y where x is in log and y in N_2)

footprint-based conformance

Diagnostics

	a	b	c	d	e	f	g	h
a				$\rightarrow$: #				
b				$\parallel$:$\rightarrow$	$\rightarrow$: #			
c				$\parallel$:$\rightarrow$	$\rightarrow$: #			
d	$\leftarrow$: #	$\parallel$:$\leftarrow$	$\parallel$:$\leftarrow$			$\leftarrow$: #		
e		$\leftarrow$: #	$\leftarrow$: #					
f				$\rightarrow$: #				
g								
h								

(x:y where x is in log and y in N_2)



Chair of Process and Data Science

Question

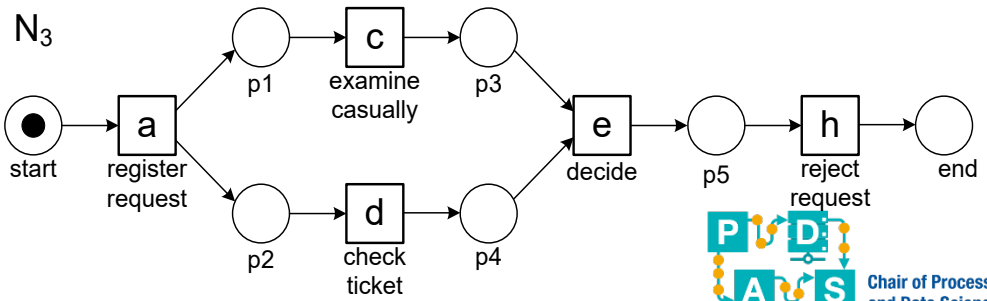
Estimate footprint-based conformance

#	trace
455	acdeh
191	abdeg
177	adceh
144	abdeh
111	acdeg
82	adceg
56	adbeh
47	acdefdbeh
38	adbeg
33	acdefbdeh
14	acdefbdeg
11	acdefdbeg
9	adcefcdeh
8	adcefdbeh
5	adcefbdeg
3	acdefbdefdbeg
2	adcefdbeg
2	adcefbdefbdeg
1	adcefdbefbdeh
1	adbefbdefdbeg
1	adcefdbefcdefdbeg
1391	

	a	*b*	*c*	*d*	*e*	*f*	*g*	*h*
a	#	→	→	→	#	#	#	#
b	←	#	#	‖	→	←	#	#
c	←	#	#	‖	→	←	#	#
d	←	‖	‖	#	→	←	#	#
e	#	←	←	←	#	→	→	→
f	#	→	→	→	←	#	#	#
g	#	#	#	#	←	#	#	#
h	#	#	#	#	←	#	#	#

?

Estimate the fraction of matching cells in footprint matrices



Chair of Process and Data Science

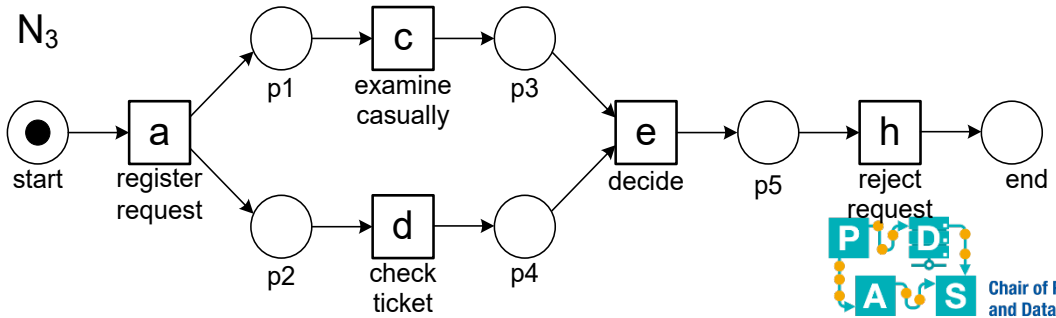
Answer

	a	b	c	d	e	f	g	h
a	#	#	→	→	#	#	#	#
b	#	#	#	#	#	#	#	#
c	←	#	#	‖	→	#	#	#
d	←	#	‖	#	→	#	#	#
e	#	#	←	←	#	#	#	→
f	#	#	#	#	#	#	#	#
g	#	#	#	#	#	#	#	#
h	#	#	#	#	←	#	#	#

Color
- Log
- Model

$$1 - \frac{16}{64} = 0.75$$

footprint-based conformance



Chair of Process and Data Science

Question

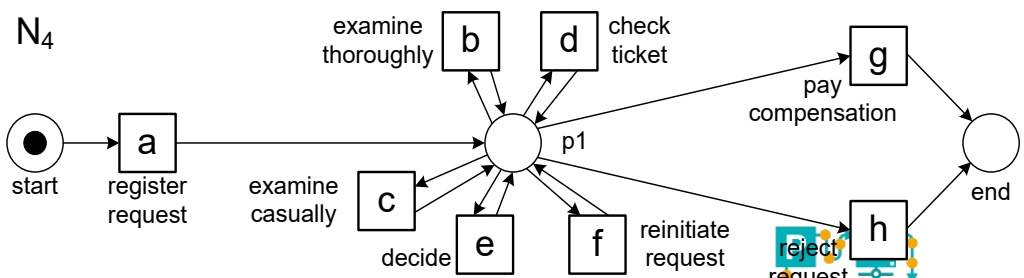
Estimate footprint-based conformance

#	trace
455	acdeh
191	abdeg
177	adceh
144	abdeh
111	acdeg
82	adceg
56	adbeh
47	acdefdbeh
38	adbeg
33	acdefbdeh
14	acdefbdeg
11	acdefdbeg
9	adcefcdeh
8	adcefdbeh
5	adcefbdeg
3	acdefbdefdbeg
2	adcefdbeg
2	adcefbdefbdeg
1	adcefdbefbdeh
1	adbefbdefdbeg
1	adcefdbefcdefdbeg
1391	

	a	b	c	d	e	f	g	h
a	#	→	→	→	#	#	#	#
b	←	#	#	‖	→	←	#	#
c	←	#	#	‖	→	←	#	#
d	←	‖	‖	#	→	←	#	#
e	#	←	←	←	#	→	→	→
f	#	→	→	→	←	#	#	#
g	#	#	#	#	←	#	#	#
h	#	#	#	#	←	#	#	#

Estimate the fraction of matching cells in footprint matrices

?



Chair of Process and Data Science

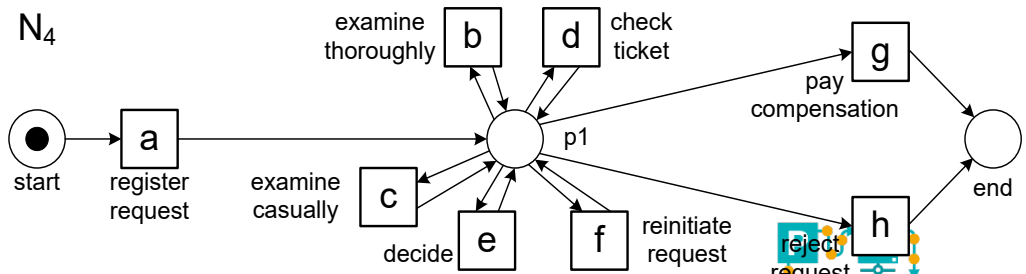
Answer

	a	b	c	d	e	f	g	h
a	#	→	→	→	#	#	#	#
b	←	#	#	∥	#	#	#	#
c	←	#	#	∥	#	#	#	#
d	←	∥	∥	#	#	#	#	#
e	#	#	#	#	#	#	→	→
f	#	#	#	#	#	#	#	#
g	#	#	#	#	←	#	#	#
h	#	#	#	#	←	#	#	#

Color
- Log
- Model

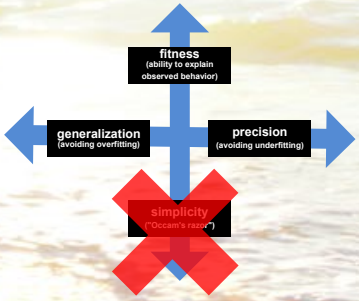
$$1 - \frac{45}{64} = 0.296875$$

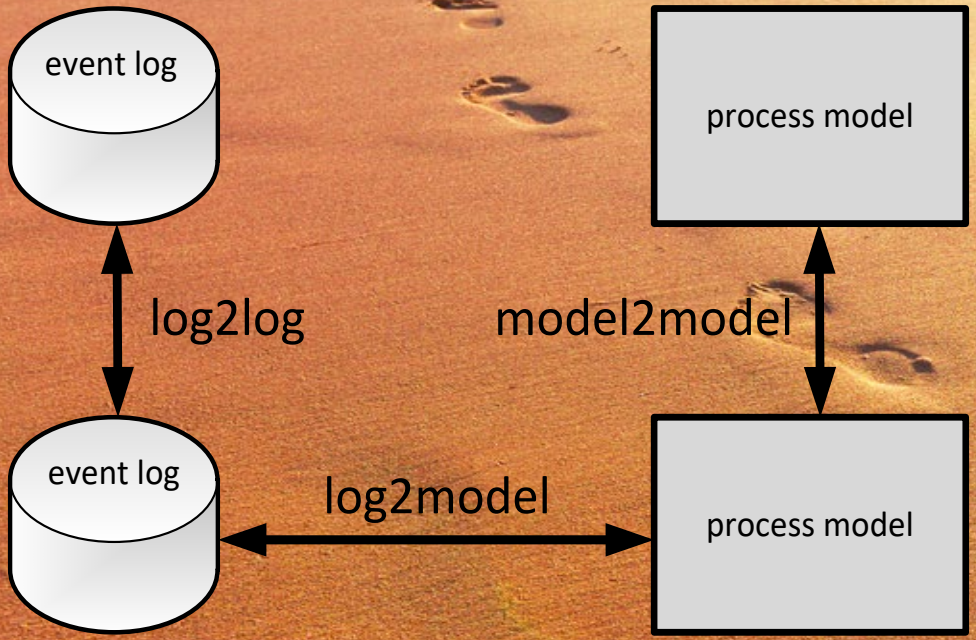
footprint-based conformance

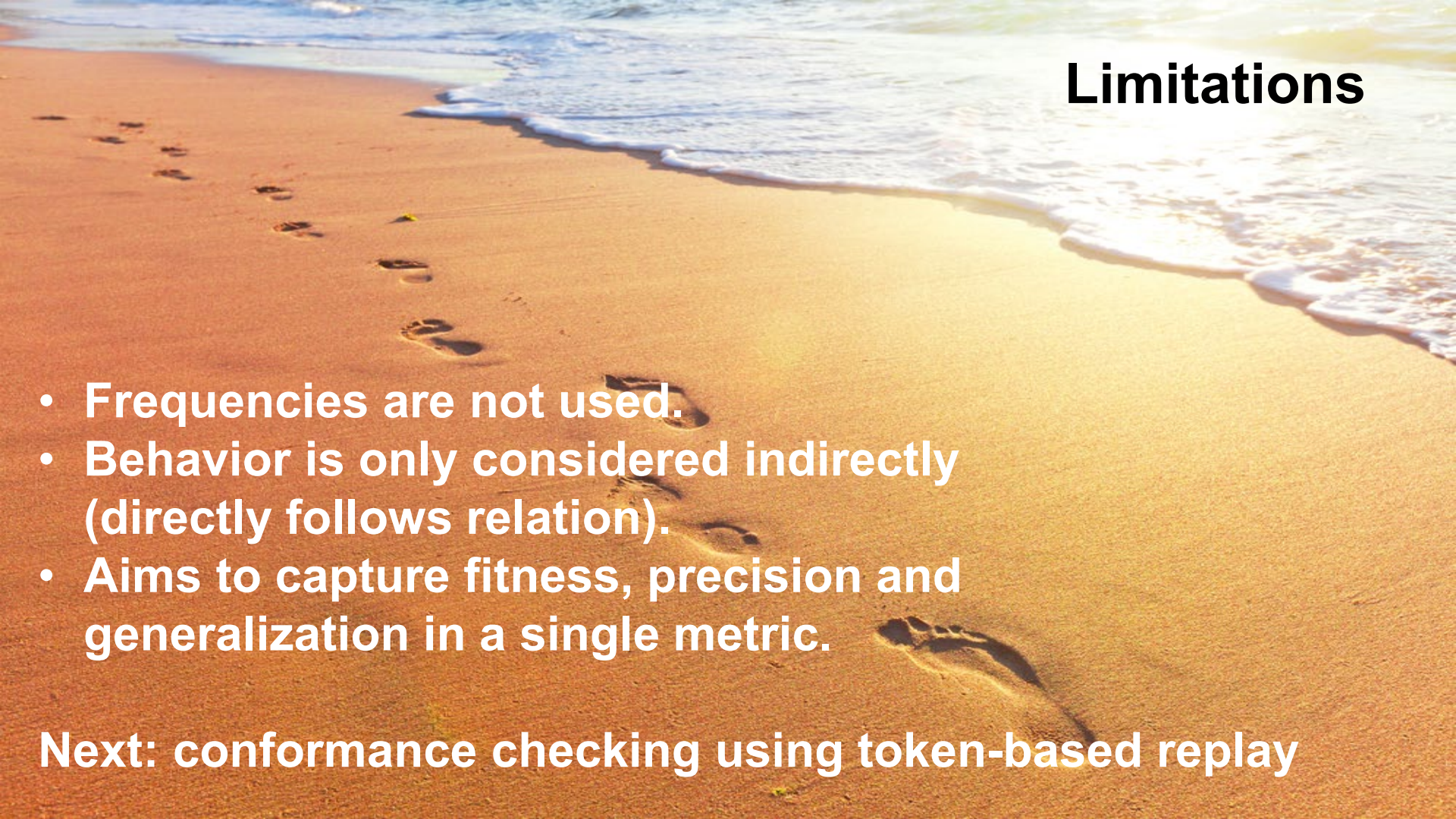


Chair of Process and Data Science

flexible


fitness
(ability to explain observed behavior)
generalization
(avoiding overfitting)
precision
(avoiding underfitting)
simplicity
("Occam's razor")


event log
process model
log2log
model2model
event log
log2model
process model

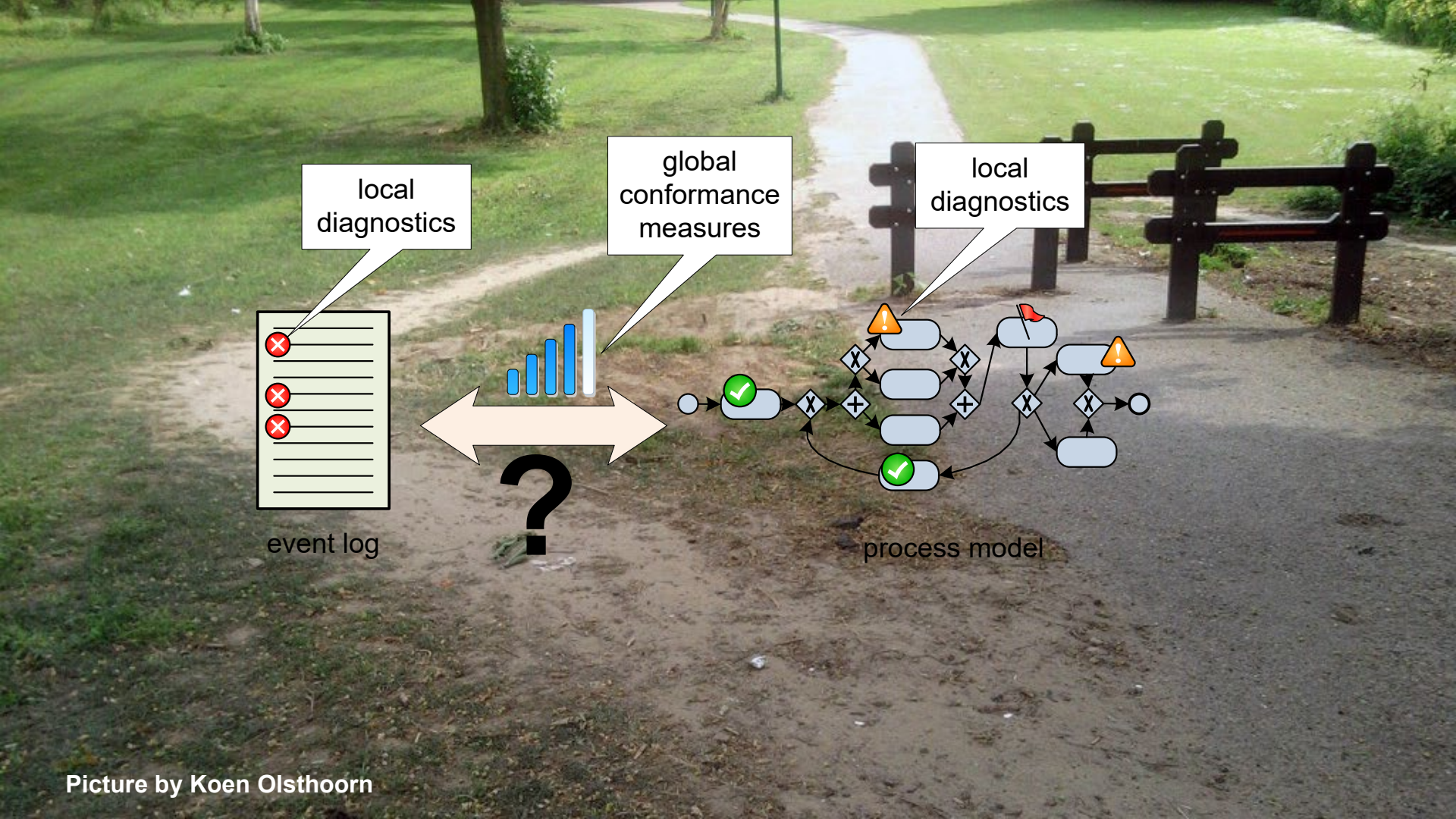


Limitations

- Frequencies are not used.
- Behavior is only considered indirectly (directly follows relation).
- Aims to capture fitness, precision and generalization in a single metric.

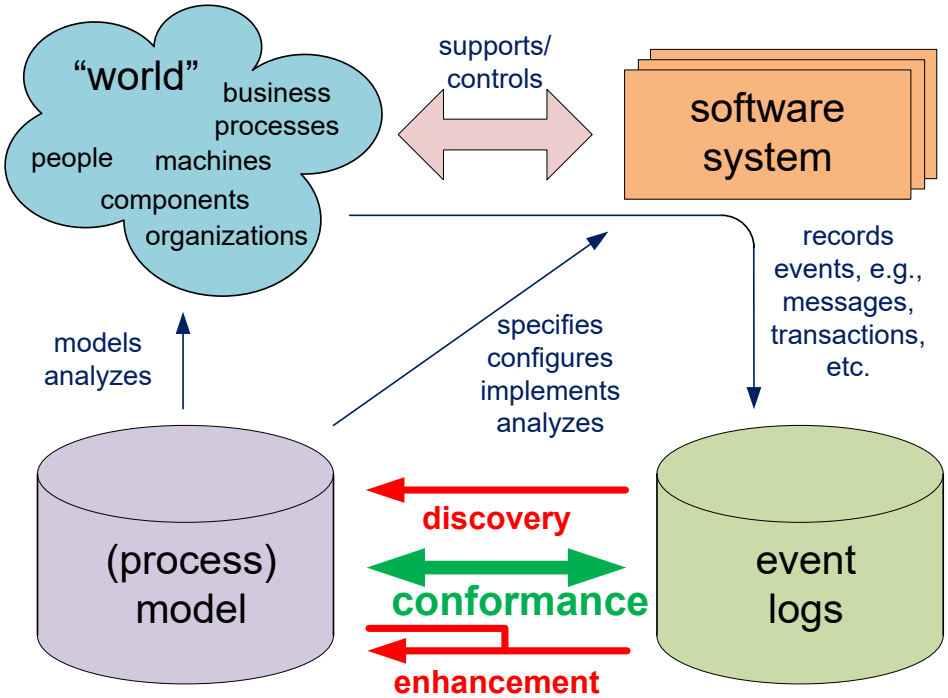
Next: conformance checking using token-based replay

Conformance checking using token-based replay



Picture by Koen Olsthoorn

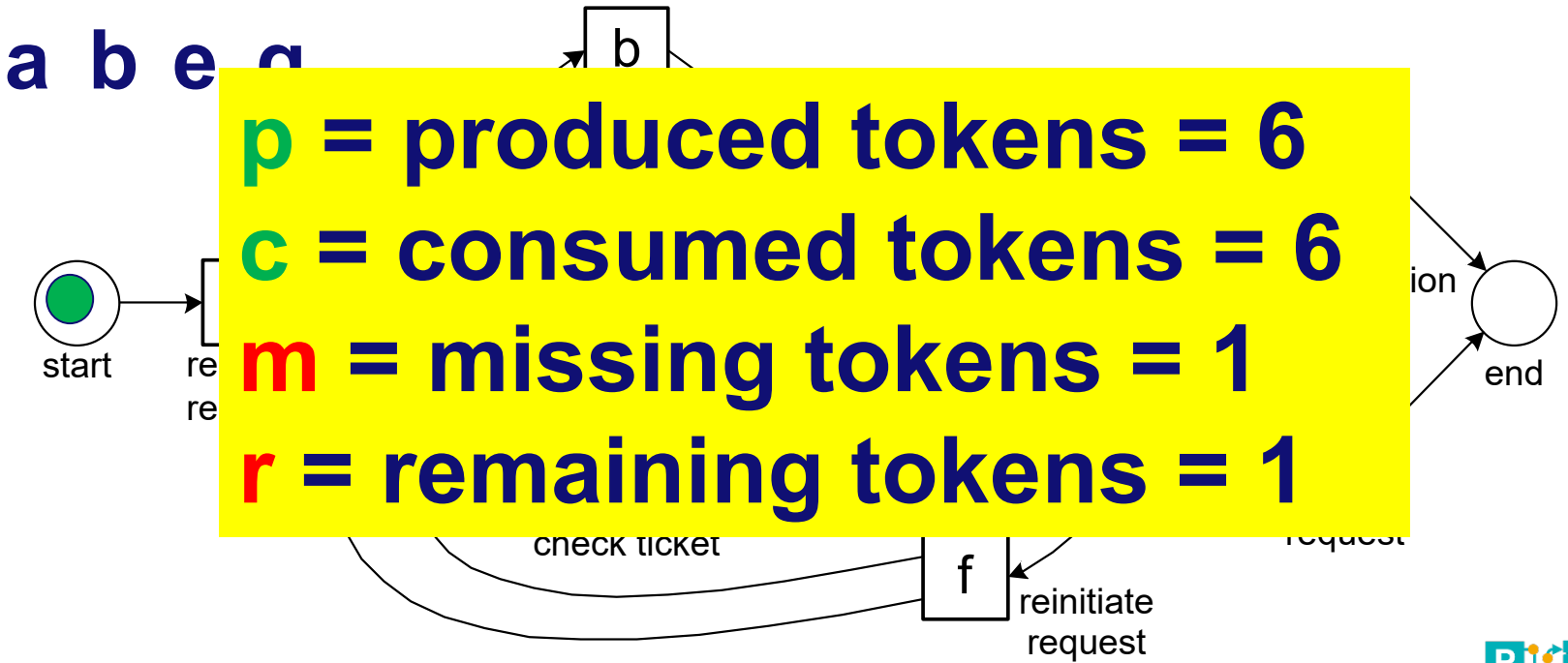
Conformance checking



1. Conformance checking using causal footprints.
2. **Conformance checking based on token-based replay.**
3. Alignment-based conformance checking.

Chair of Process and Data Science

Counting tokens while replaying



Quantifying fitness at the trace level

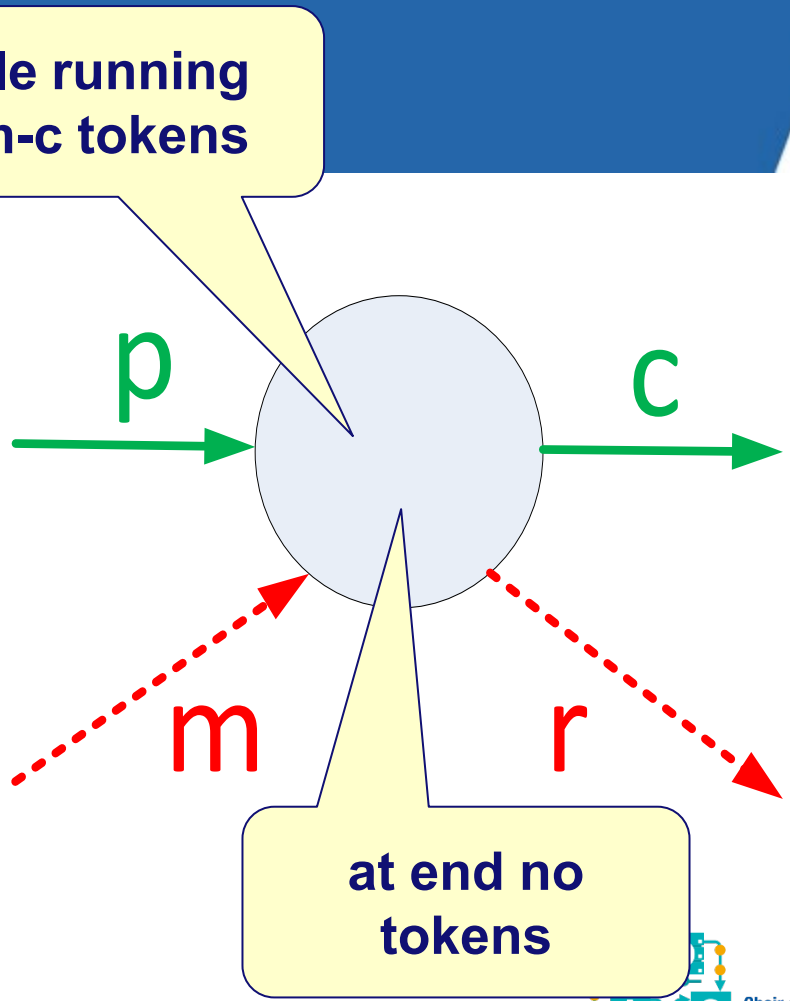
$$fitness(\sigma, N) = \frac{1}{2}\left(1 - \frac{1}{6}\right) + \frac{1}{2}\left(1 - \frac{1}{6}\right) = 0.83333$$

p = produced tokens = 6
c = consumed tokens = 6
m = missing tokens = 1
r = remaining tokens = 1

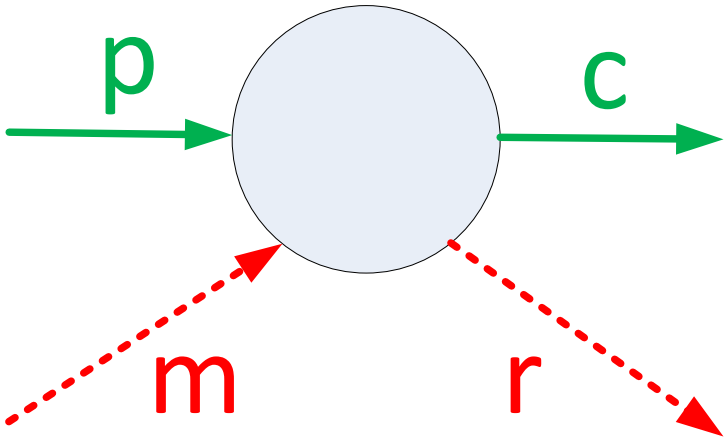
Approach (1/3)

Use four counters:

- p = produced tokens
- c = consumed tokens
- m = missing tokens
  (consumed while not there)
- r = remaining tokens
  (produced but not consumed)

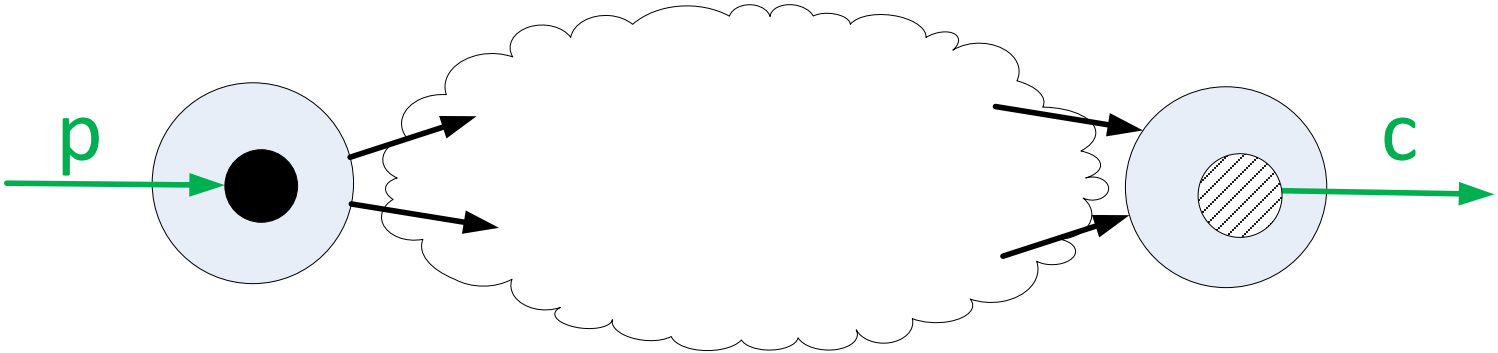


Approach (2/3)



- **Invariants**
  - **At any time: p+m ≥ c ≥ m (also per place)**
  - **At the end: r = p + m - c (also per place)**

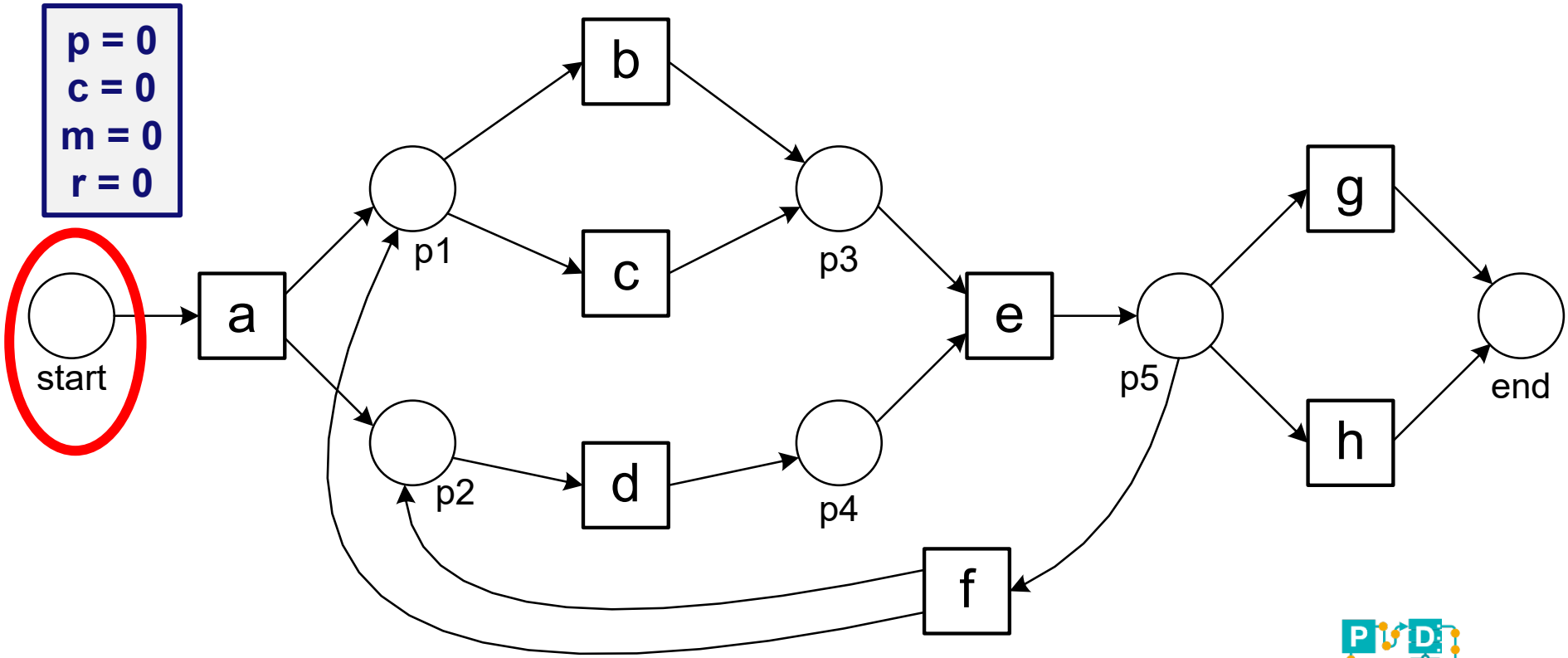
Approach (3/3)



Initialization and finalization:

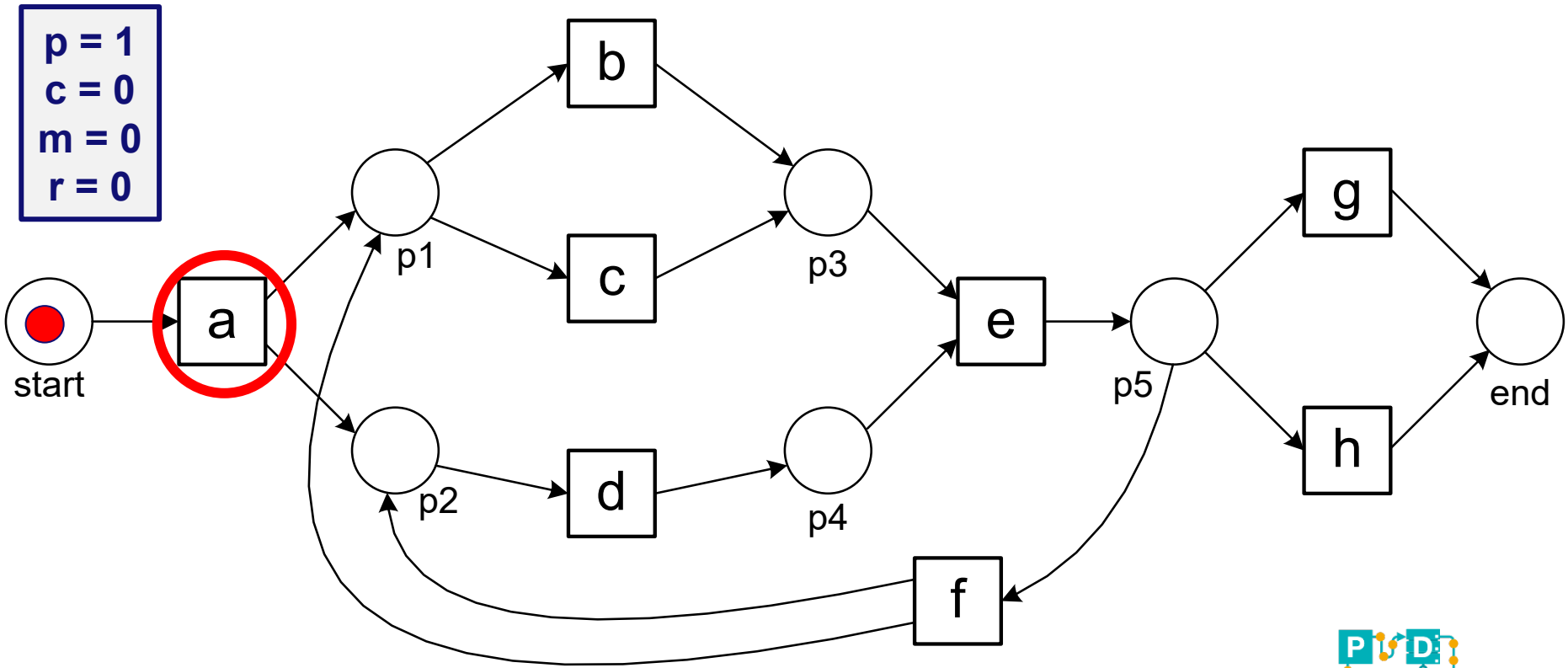
- **In the beginning a token is produced for the source place: p = 1.**
- **At the end a token is consumed from the sink place (also if not there): c' = c + 1.**

Replaying $\sigma_1 = \langle a, c, d, e, h \rangle$



Chair of Process and Data Science

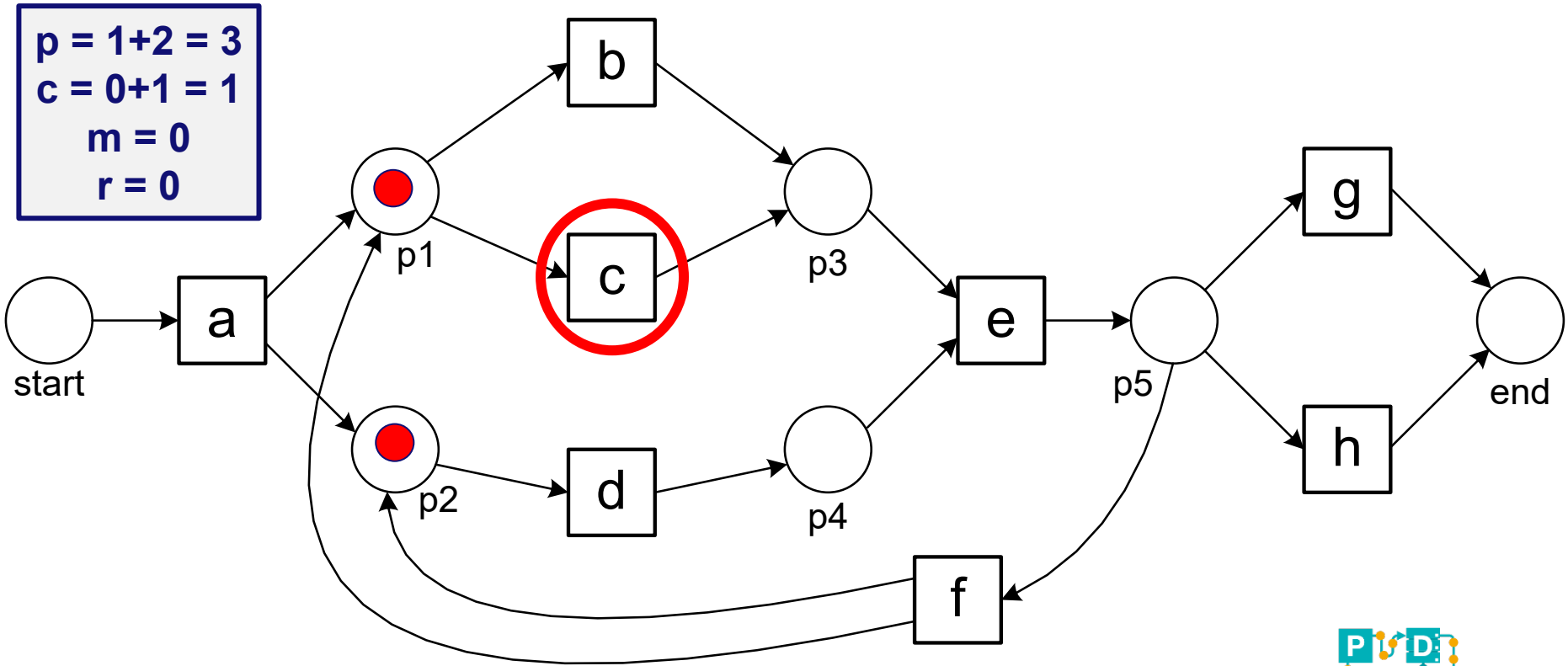
Replaying $\sigma_1 = \langle a, c, d, e, h \rangle$



Chair of Process and Data Science

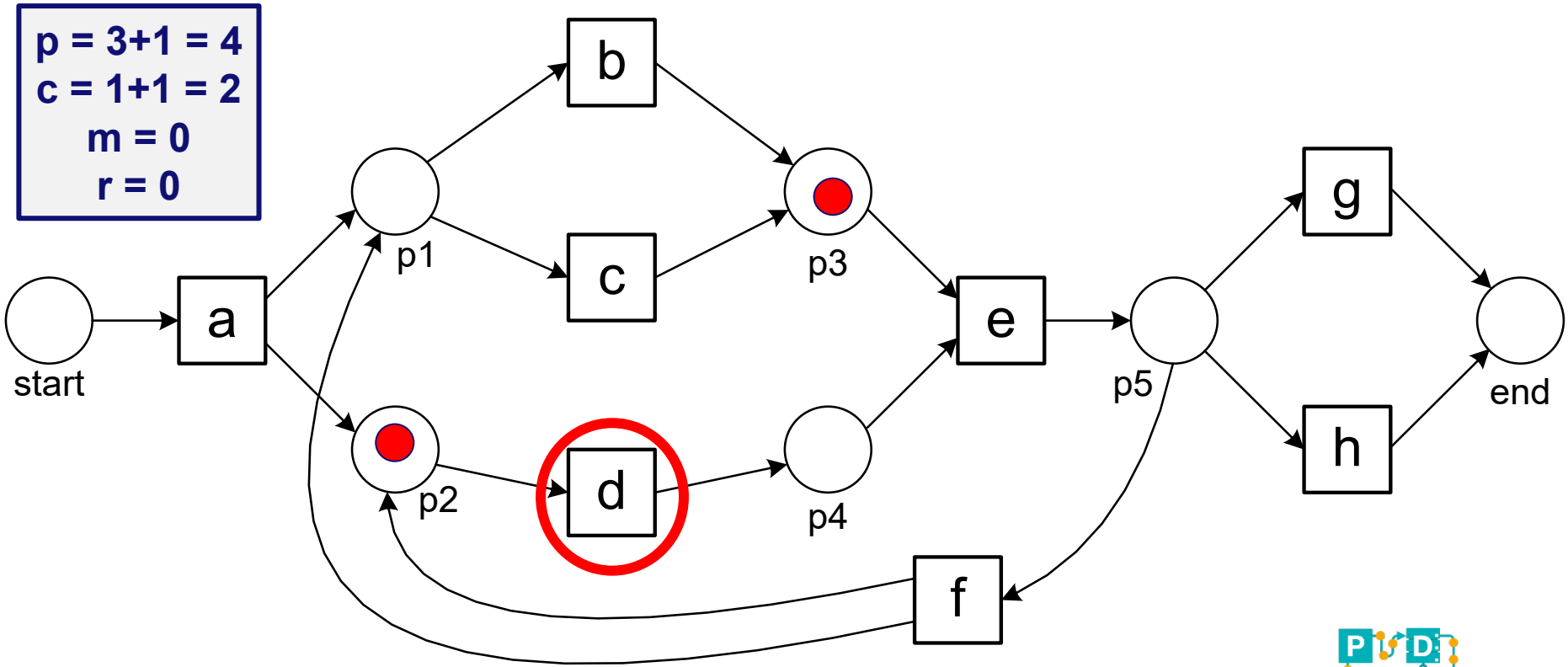
Replaying

$$\sigma_1 = \langle a, c, d, e, h \rangle$$



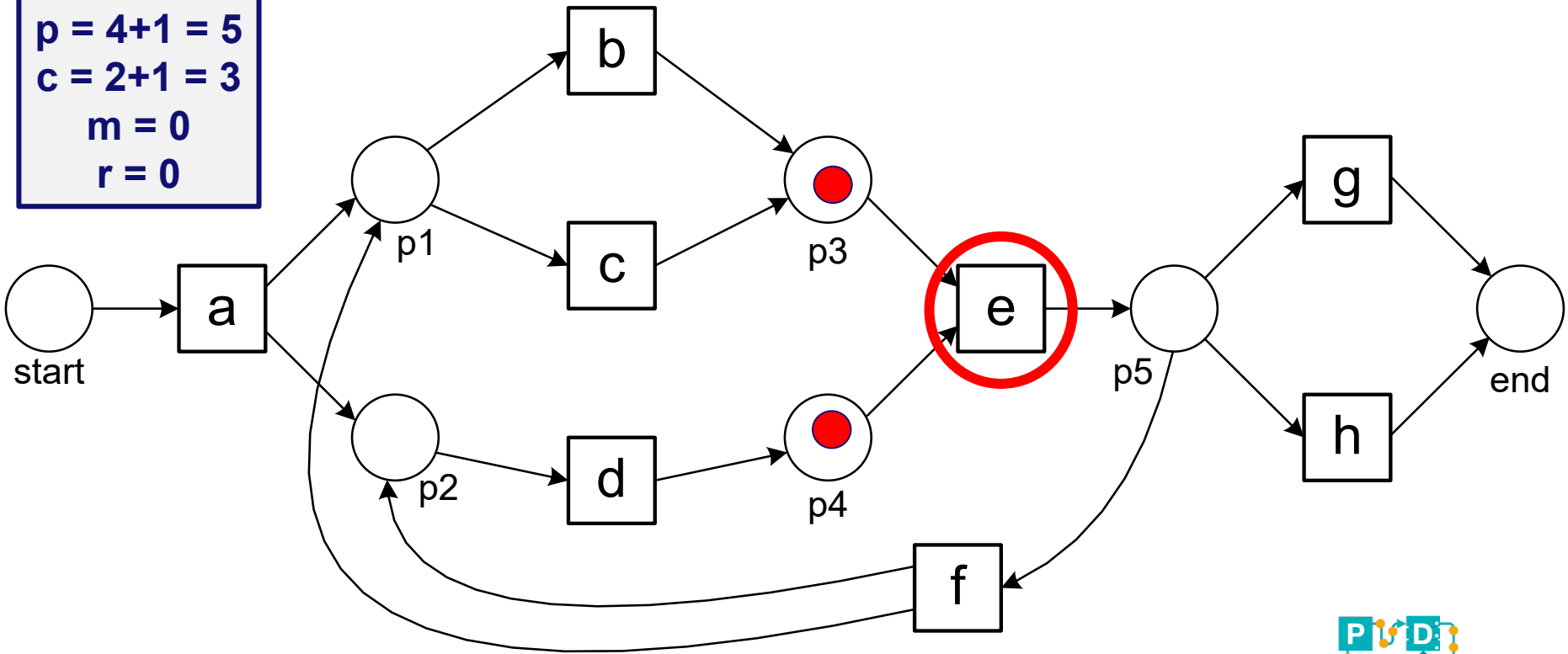
Replaying

$$\sigma_1 = \langle a, c, d, e, h \rangle$$



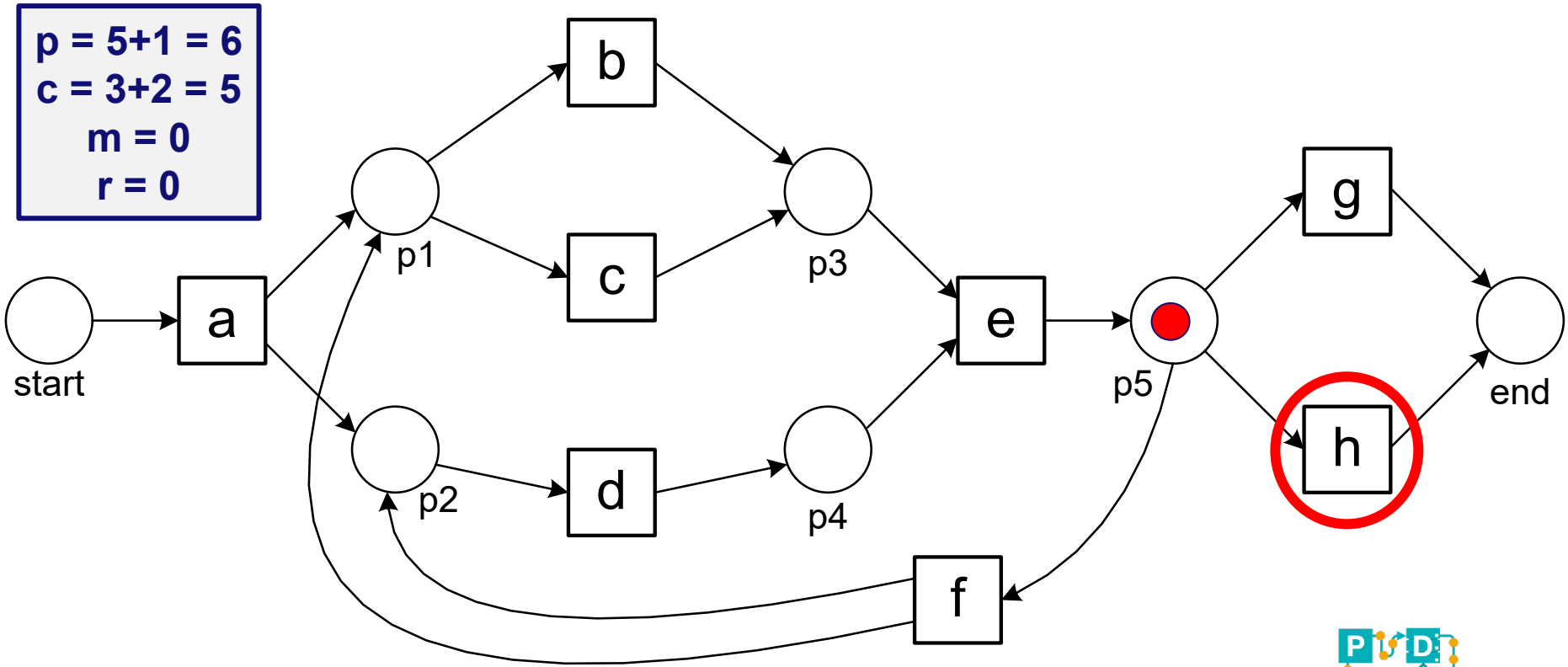
Replaying $\sigma_1 = \langle a, c, d, e, h \rangle$

p = 4+1 = 5
c = 2+1 = 3
m = 0
r = 0

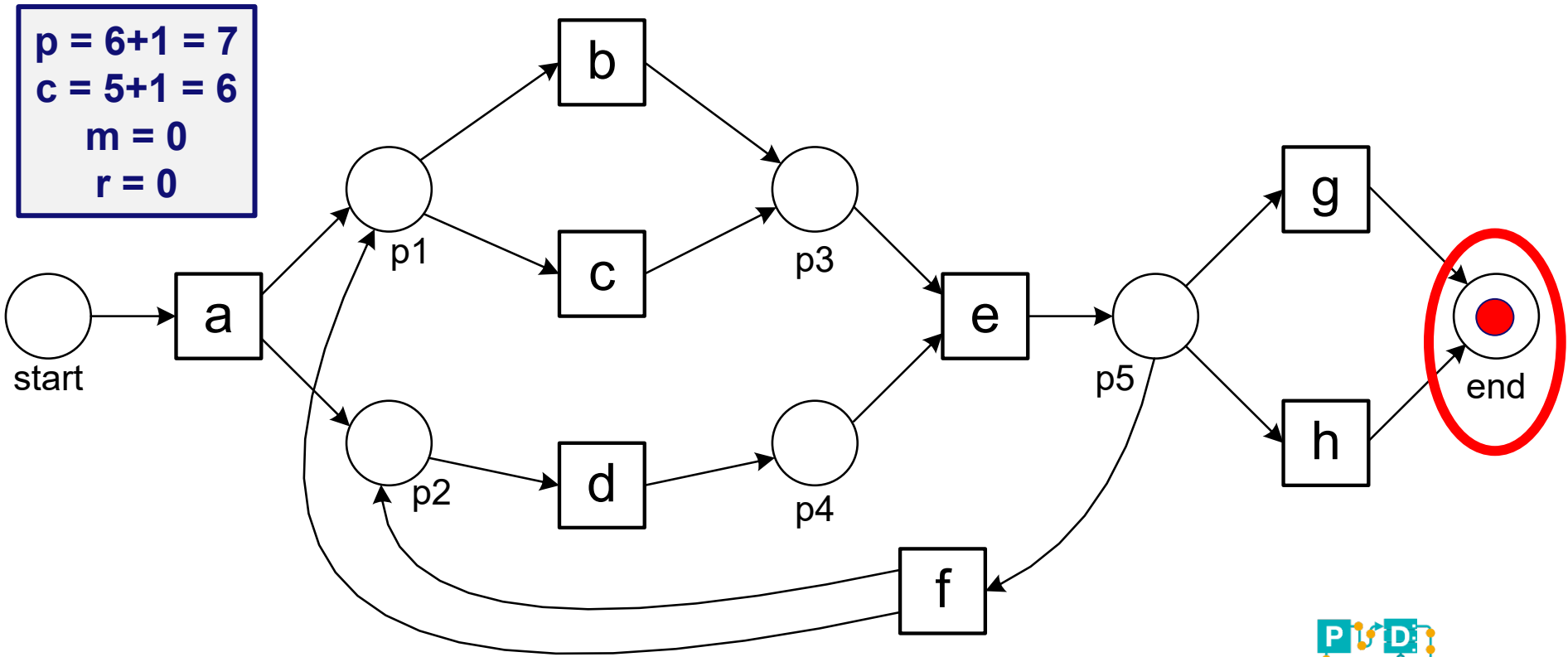


Chair of Process and Data Science

Replaying $\sigma_1 = \langle a, c, d, e, h \rangle$

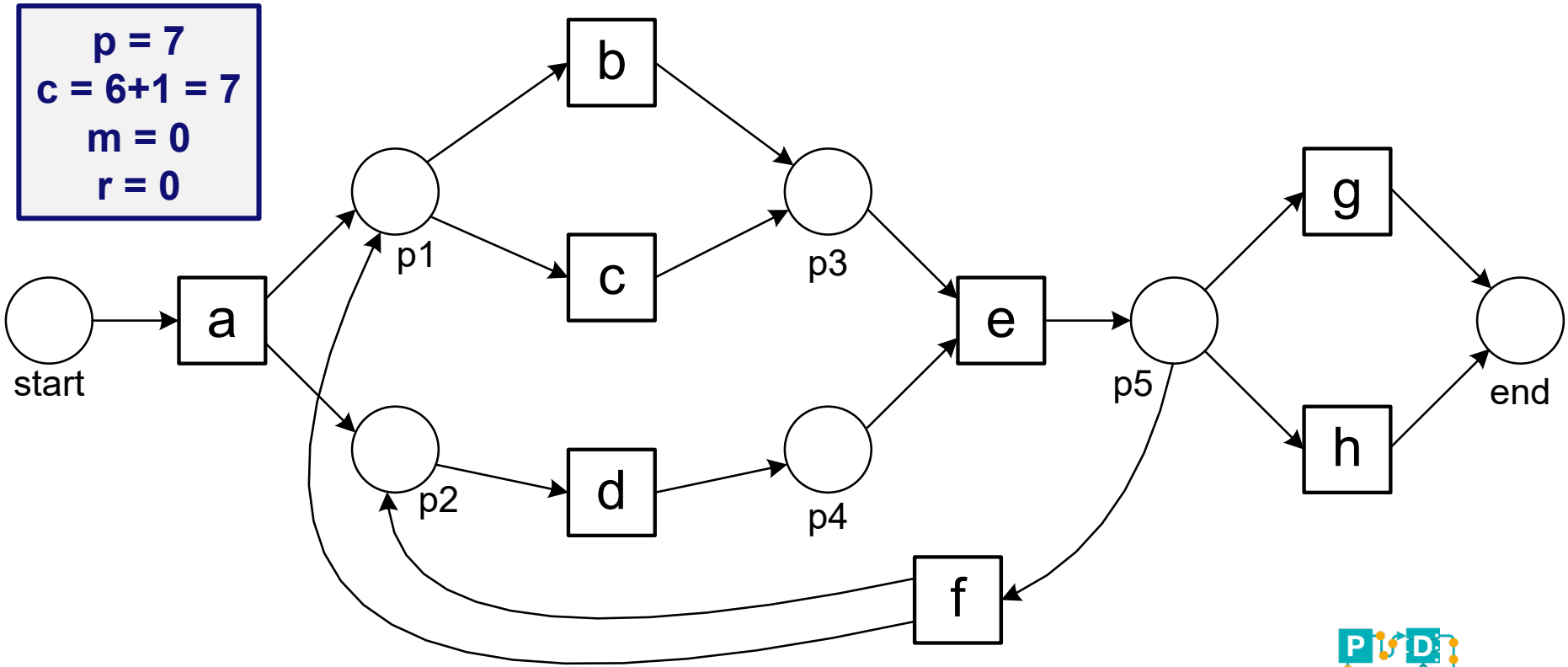


Replaying $\sigma_1 = \langle a, c, d, e, h \rangle$



Chair of Process and Data Science

Replaying $\sigma_1 = \langle a, c, d, e, h \rangle$

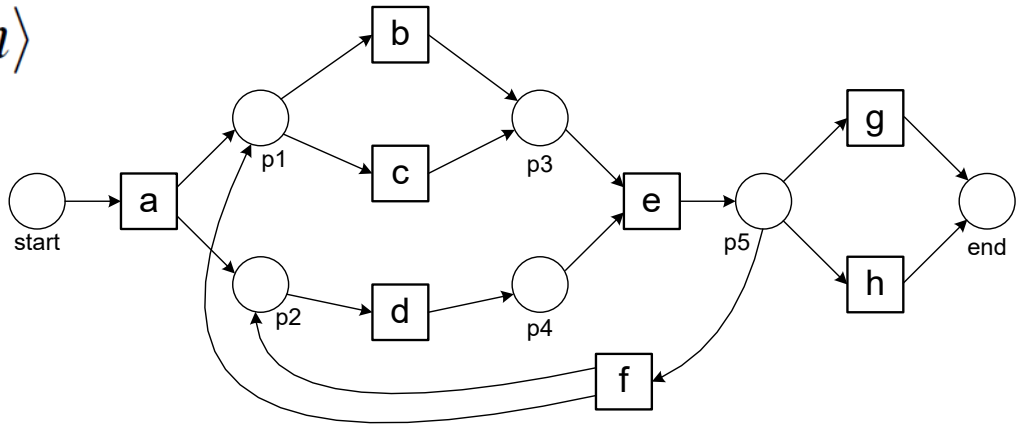


Chair of Process and Data Science

Quantifying fitness at the trace level

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

$$\sigma_1 = \langle a, c, d, e, h \rangle$$



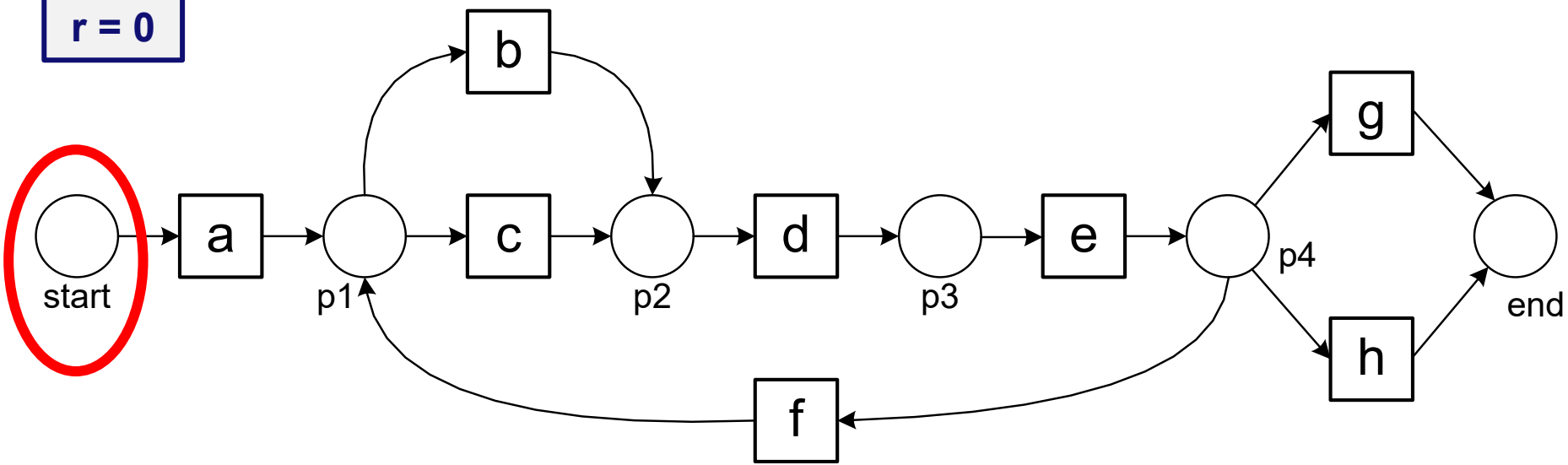
$$fitness(\sigma, N) = \frac{1}{2}\left(1 - \frac{0}{7}\right) + \frac{1}{2}\left(1 - \frac{0}{7}\right) = 1$$

Chair of Process and Data Science

Replaying

$$\sigma_3 = \langle a, d, c, e, h \rangle$$

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

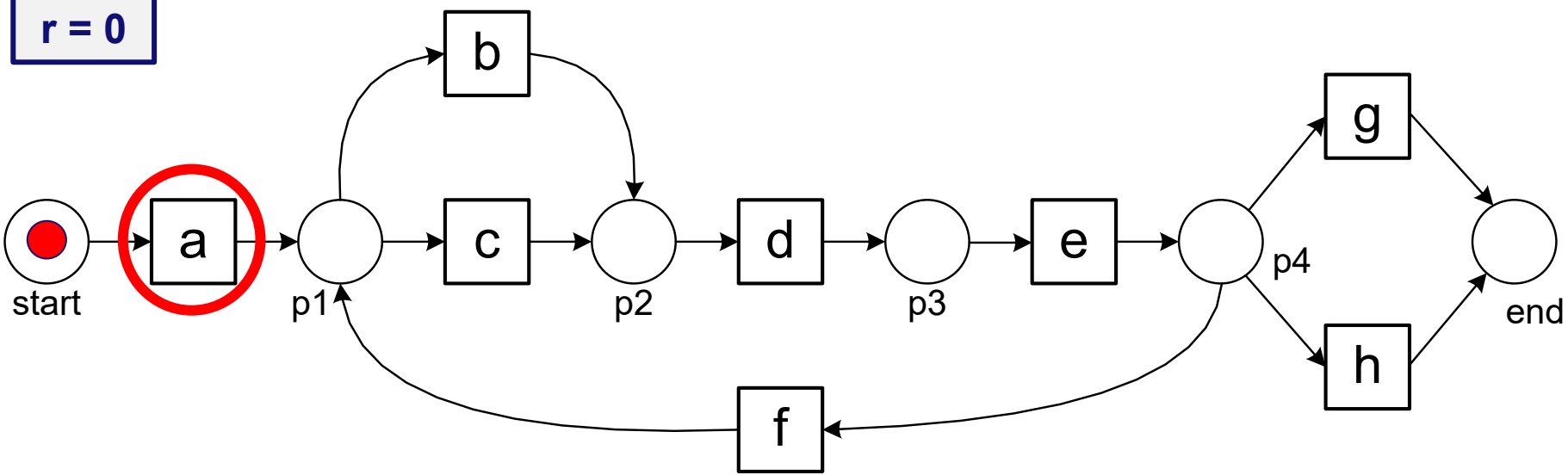


Chair of Process and Data Science

Replaying

$$\sigma_3 = \langle a, d, c, e, h \rangle$$

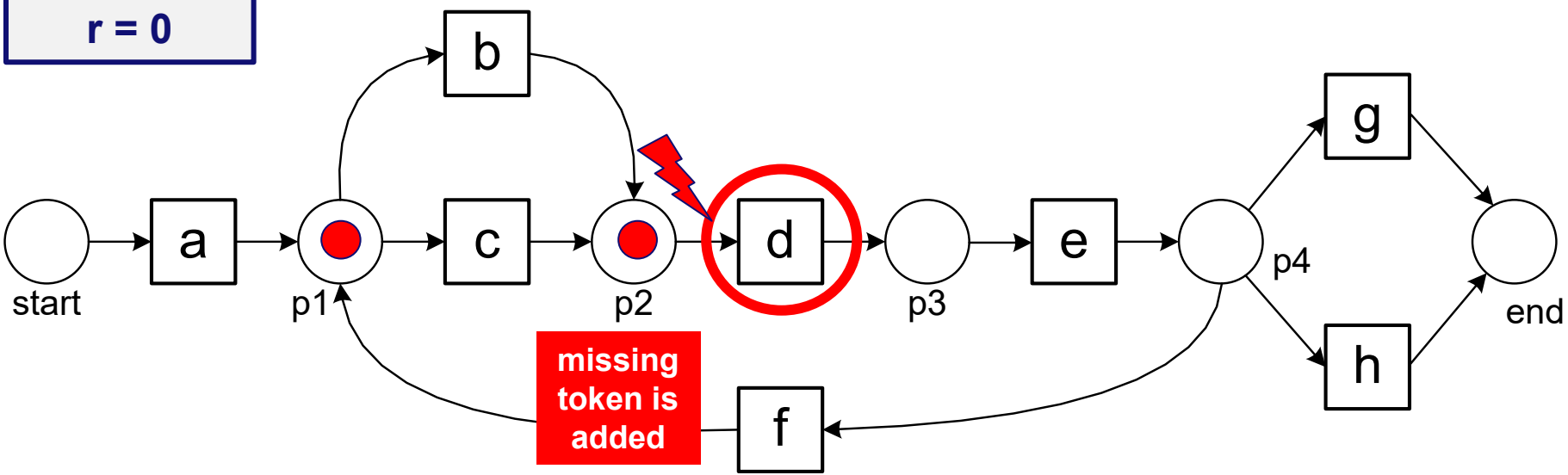
p = 1
c = 0
m = 0
r = 0



Replaying

$$\sigma_3 = \langle a, d, c, e, h \rangle$$

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

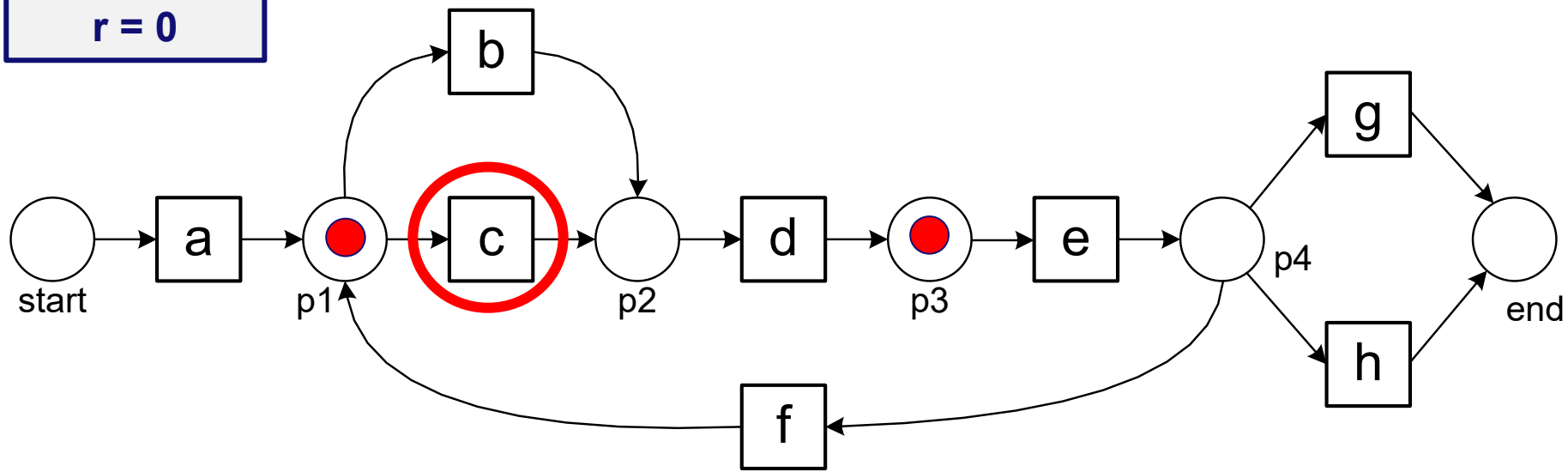


Chair of Process and Data Science

Replaying

$$\sigma_3 = \langle a, d, c, e, h \rangle$$

p = 2+1 = 3
c = 1+1 = 2
m = 0+1 = 1
r = 0

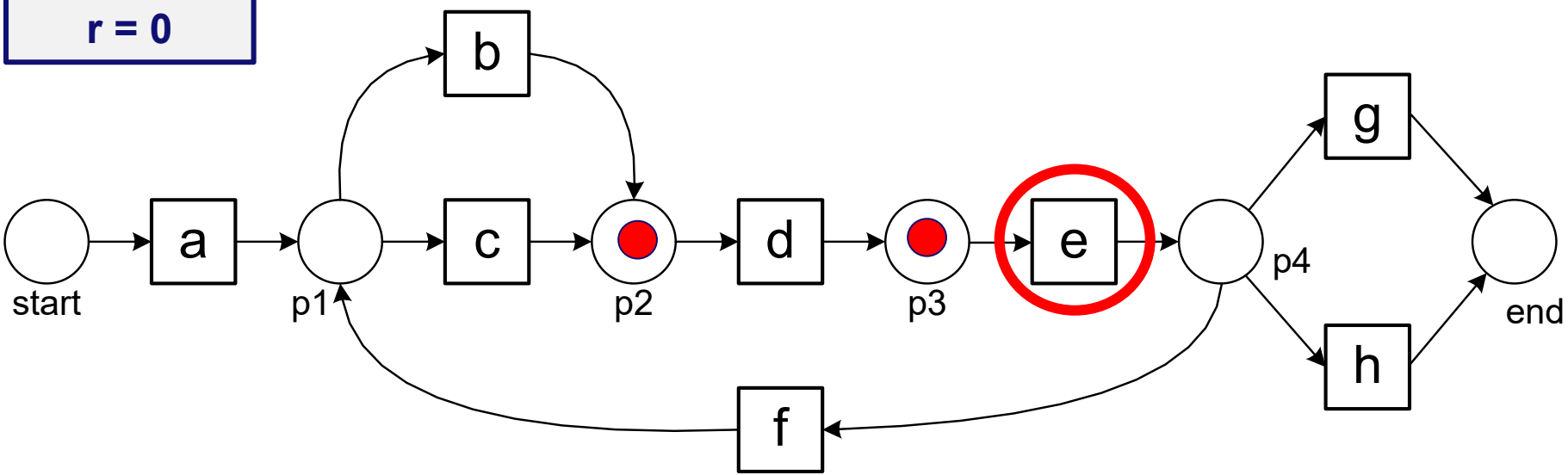


Chair of Process and Data Science

Replaying

$$\sigma_3 = \langle a, d, c, e, h \rangle$$

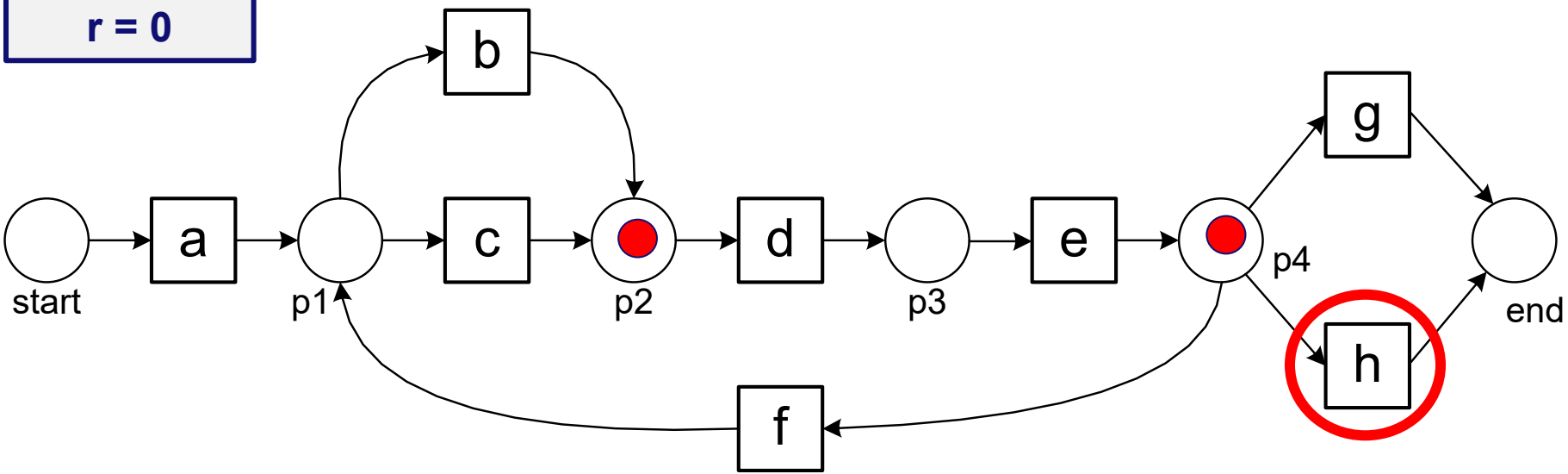
p = 3+1 = 4
c = 2+1 = 3
m = 1
r = 0



Replaying

$$\sigma_3 = \langle a, d, c, e, h \rangle$$

p = 4+1 = 5
c = 3+1 = 4
m = 1
r = 0

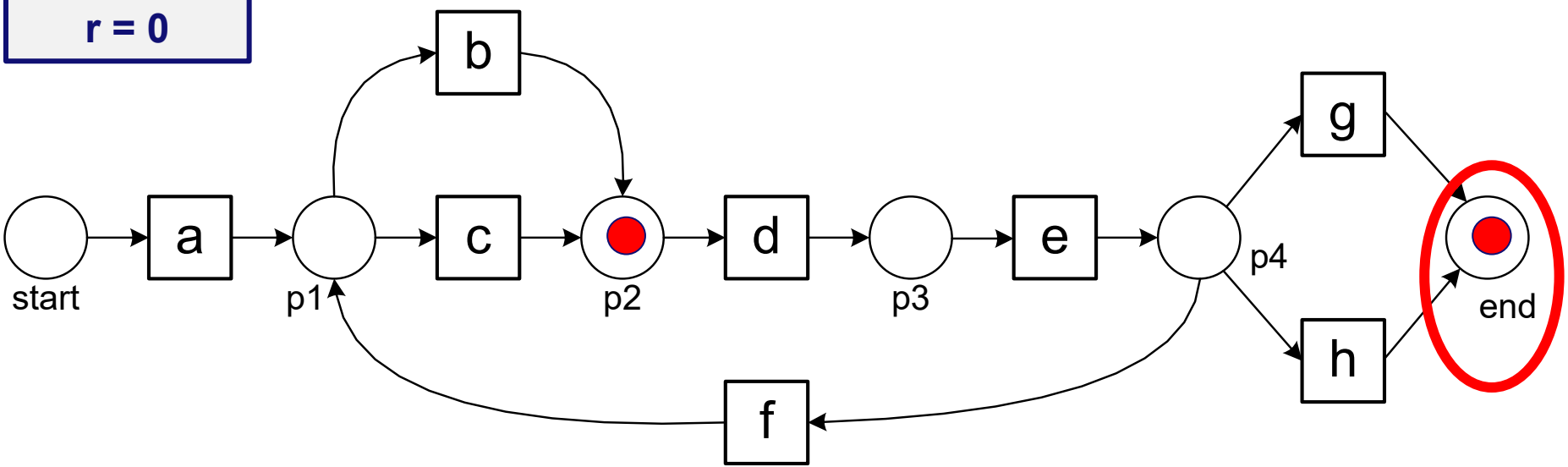


Chair of Process and Data Science

Replaying

$$\sigma_3 = \langle a, d, c, e, h \rangle$$

p = 5+1 = 6
c = 4+1 = 5
m = 1
r = 0

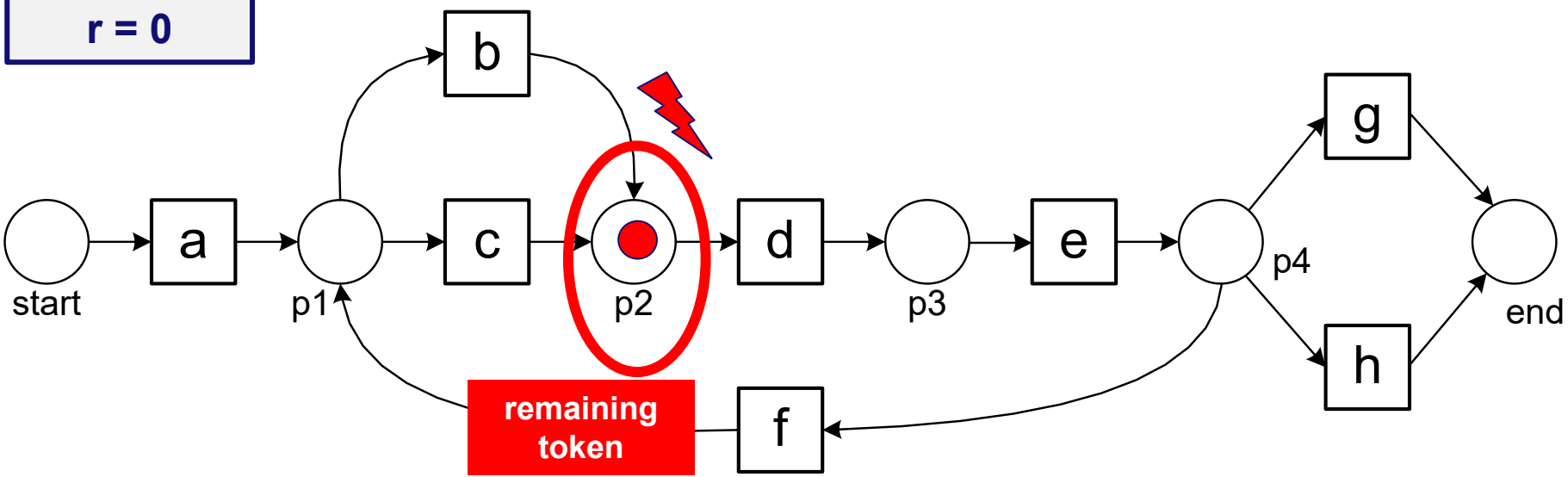


Chair of Process and Data Science

Replaying

$$\sigma_3 = \langle a, d, c, e, h \rangle$$

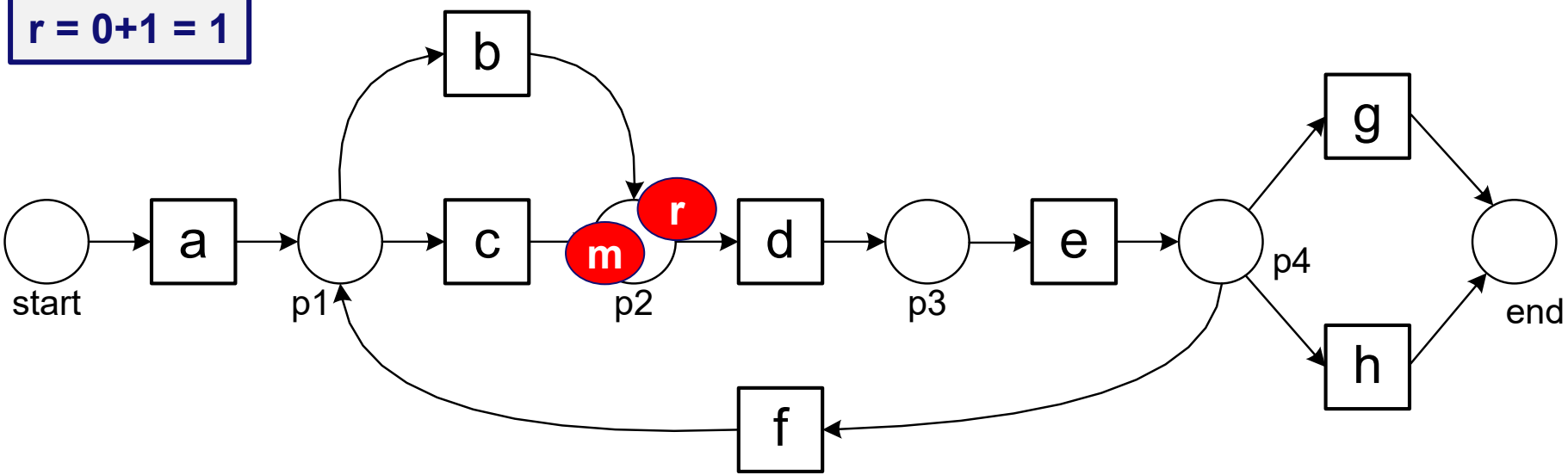
p = 6
c = 5+1 = 6
m = 1
r = 0



Replaying

$$\sigma_3 = \langle a, d, c, e, h \rangle$$

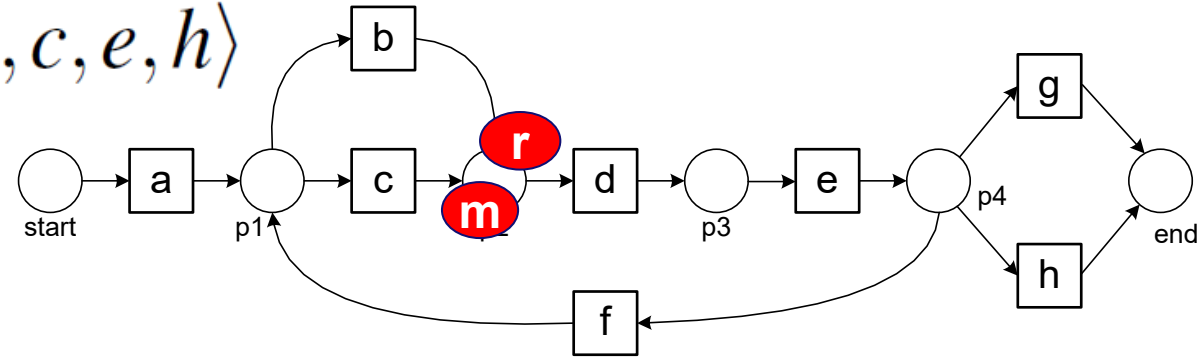
p = 6
c = 6
m = 1
r = 0+1 = 1



Quantifying fitness at the trace level

p = 6
c = 6
m = 1
r = 1

$$\sigma_3 = \langle a, d, c, e, h \rangle$$



$$fitness(\sigma, N) = \frac{1}{2}\left(1 - \frac{1}{6}\right) + \frac{1}{2}\left(1 - \frac{1}{6}\right) = 0.8333$$

Chair of Process and Data Science

Fitness at the log level

missing tokens

consumed tokens

remaining tokens

produced tokens

$$fitness(L,N) = \frac{1}{2}\left(1 - \frac{\sum_{\sigma \in L} L(\sigma) \times m_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times c_{N,\sigma}}\right) + \frac{1}{2}\left(1 - \frac{\sum_{\sigma \in L} L(\sigma) \times r_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times p_{N,\sigma}}\right)$$

Looks scar
just needs t
sums of p, c, m, and r
over the multiset of
traces in de

#	trace
455	acdeh
191	abdeg
177	adceh
144	abdeh
111	acdeg
82	adceg
56	adbeh
47	acdefdbeh
38	adbeg
33	acdefbdeh
14	acdefbdeg
11	acdefdbeg
9	adcefcdeh
8	adcefdbeh
5	adcefbdeg
3	acdefbdefdbeg
2	adcefdbeg
2	adcefbdefbdeg
1	adcefdbefbdeh
1	adbefbdefdbeg
1	adcefdbefcdefdbeg
1391	

?

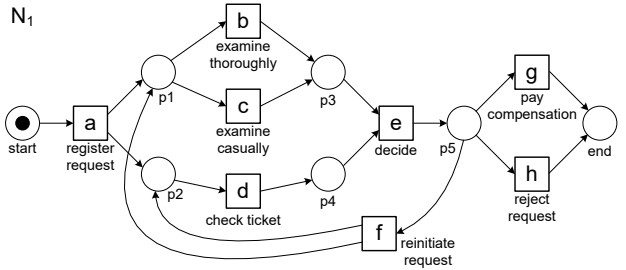
$$fitness(L,N) = \frac{1}{2}\left(1 - \frac{\sum_{\sigma \in L} L(\sigma) \times m_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times c_{N,\sigma}}\right) + \frac{1}{2}\left(1 - \frac{\sum_{\sigma \in L} L(\sigma) \times r_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times p_{N,\sigma}}\right)$$

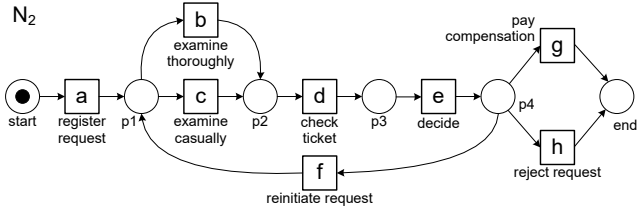
$$fitness(L_{full}, N_1) = 1$$

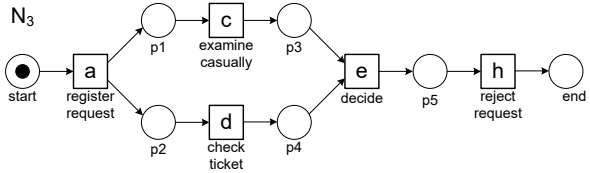
$$fitness(L_{full}, N_2) = 0.9504$$

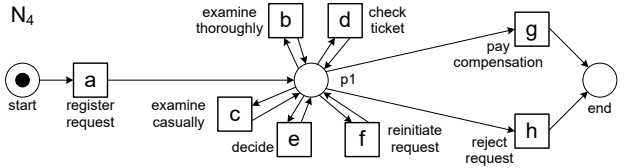
$$fitness(L_{full}, N_3) = 0.8797$$

$$fitness(L_{full}, N_4) = 1$$

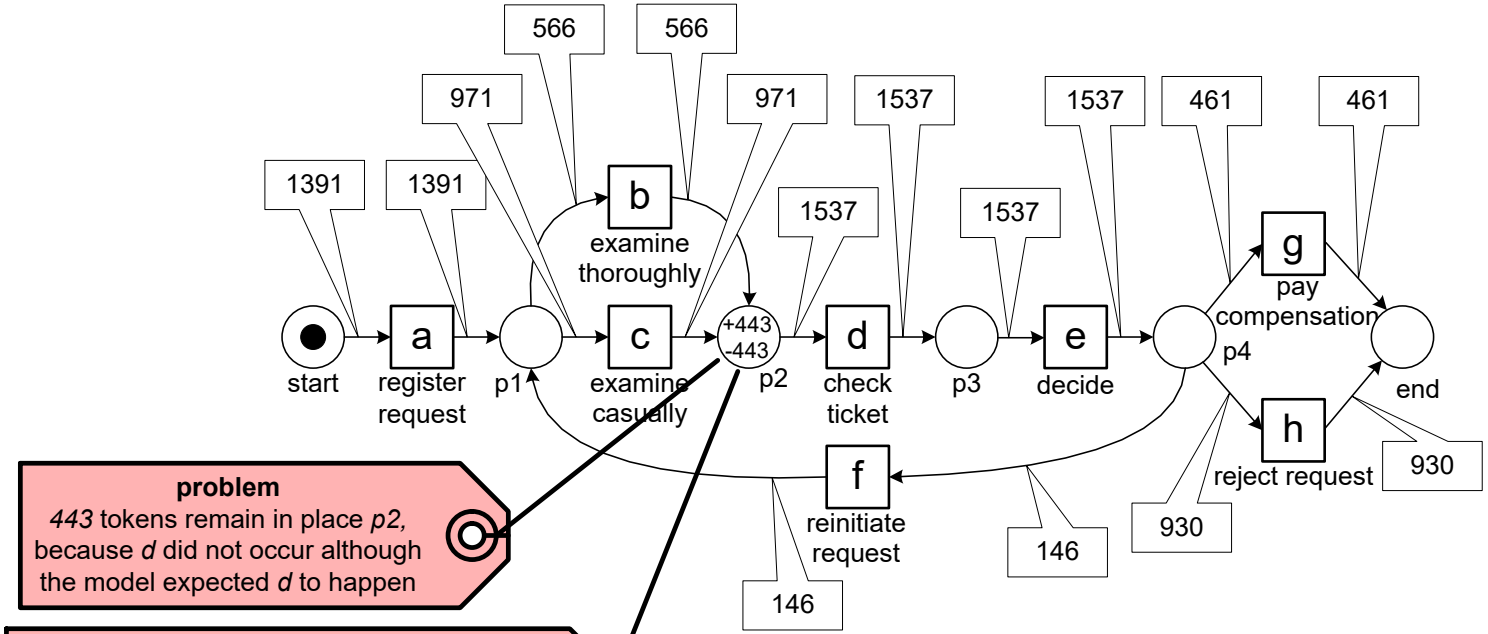








Diagnostics

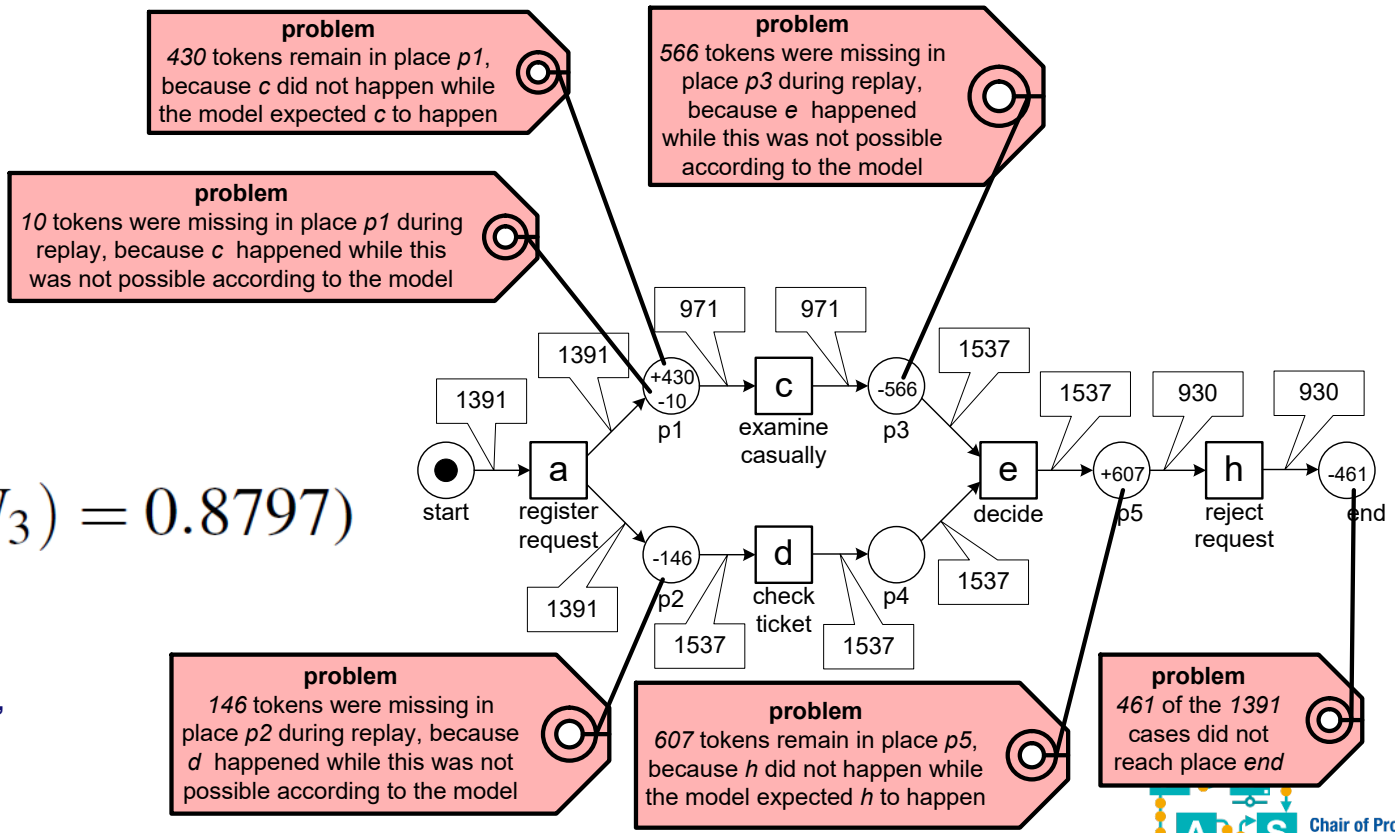


$$(fitness(L_{full}, N_2) = 0.9504)$$

Chair of Process and Data Science

Diagnostics

$$(fitness(L_{full}, N_3) = 0.8797)$$



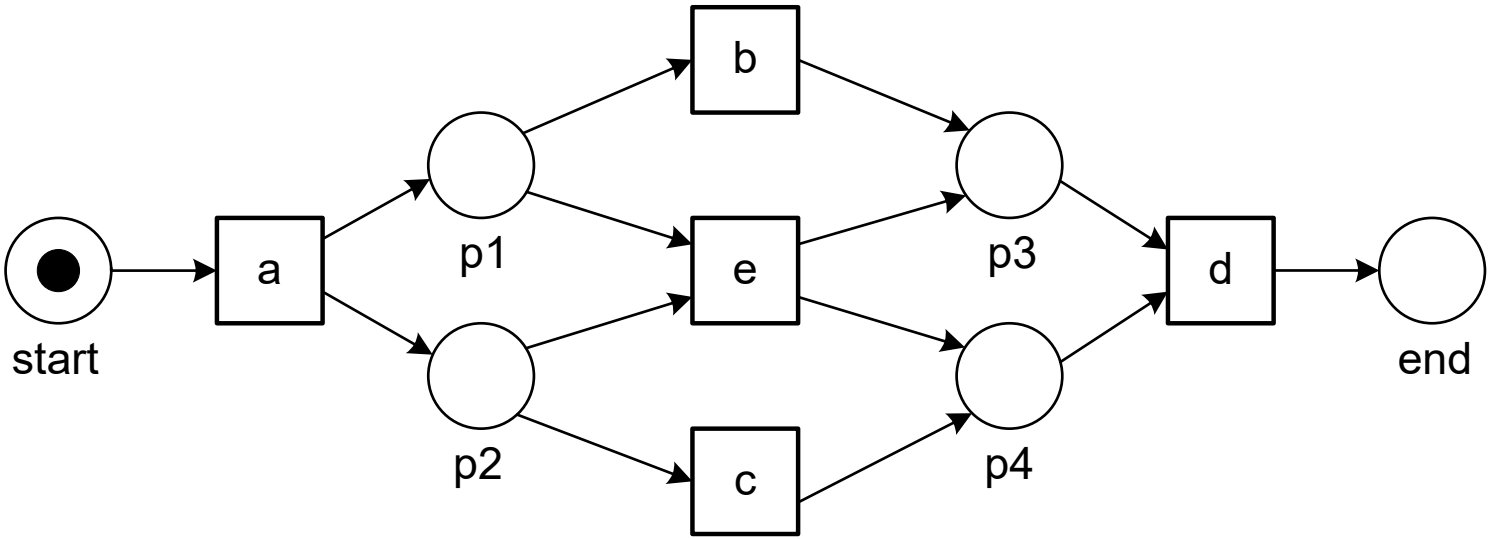
Remark: If event log and model consider different sets of activities, this should be addressed first.
Activities in log, but not in model, are simply ignored.

Examples

Question (may take some time)

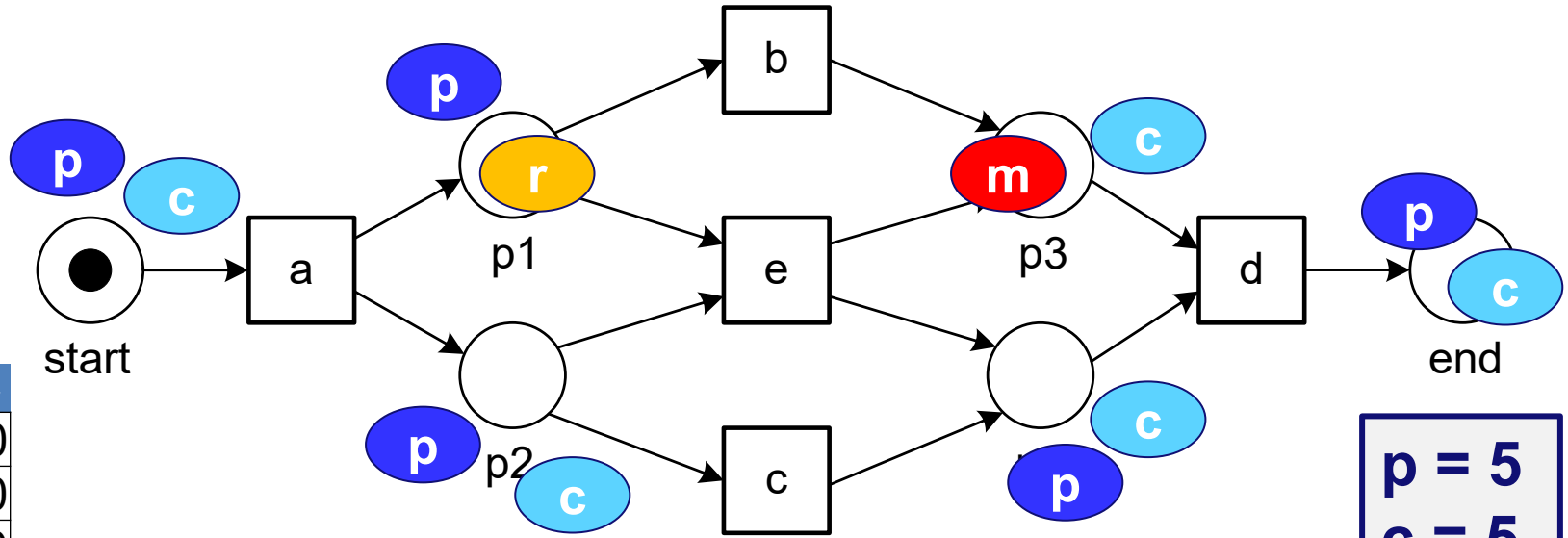
Compute fitness using missing and remaining tokens

trace	frequency
abcd	10
acbd	10
aed	10
abd	2
acd	1
ad	1
abbd	1



- **Consider the event log containing 35 cases.**
- **What is the fitness?**

Let us pick one trace: acd



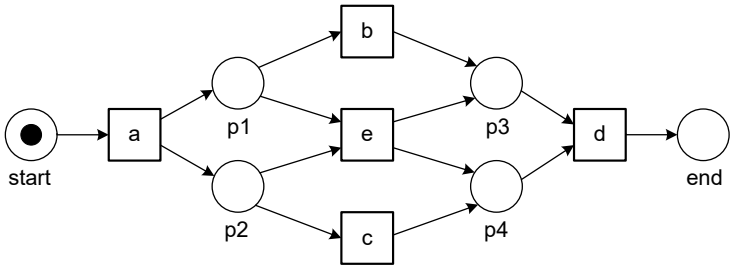
trace	frequency
abcd	10
acbd	10
aed	10
abd	2
acd	1
ad	1
abbd	1

$$fitness(L,N) = \frac{1}{2}\left(1 - \frac{\sum_{\sigma \in L} L(\sigma) \times m_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times c_{N,\sigma}}\right) + \frac{1}{2}\left(1 - \frac{\sum_{\sigma \in L} L(\sigma) \times r_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times p_{N,\sigma}}\right)$$

p = 5
c = 5
m = 1
r = 1

Fitness = 0.9658

trace	frequency	produced tokens (p)	remaining tokens (r)	consumed tokens (c)	missing tokens (m)	produced tokens (all)	remaining tokens (all)	consumed tokens (all)	missing tokens (all)
abcd	10	6	0	6	0	60	0	60	0
acbd	10	6	0	6	0	60	0	60	0
aed	10	6	0	6	0	60	0	60	0
abd	2	5	1	5	1	10	2	10	2
acd	1	5	1	5	1	5	1	5	1
ad	1	4	2	4	2	4	2	4	2
abbd	1	6	2	6	2	6	2	6	2



205	7	205	7
sum p	sum r	sum c	sum m

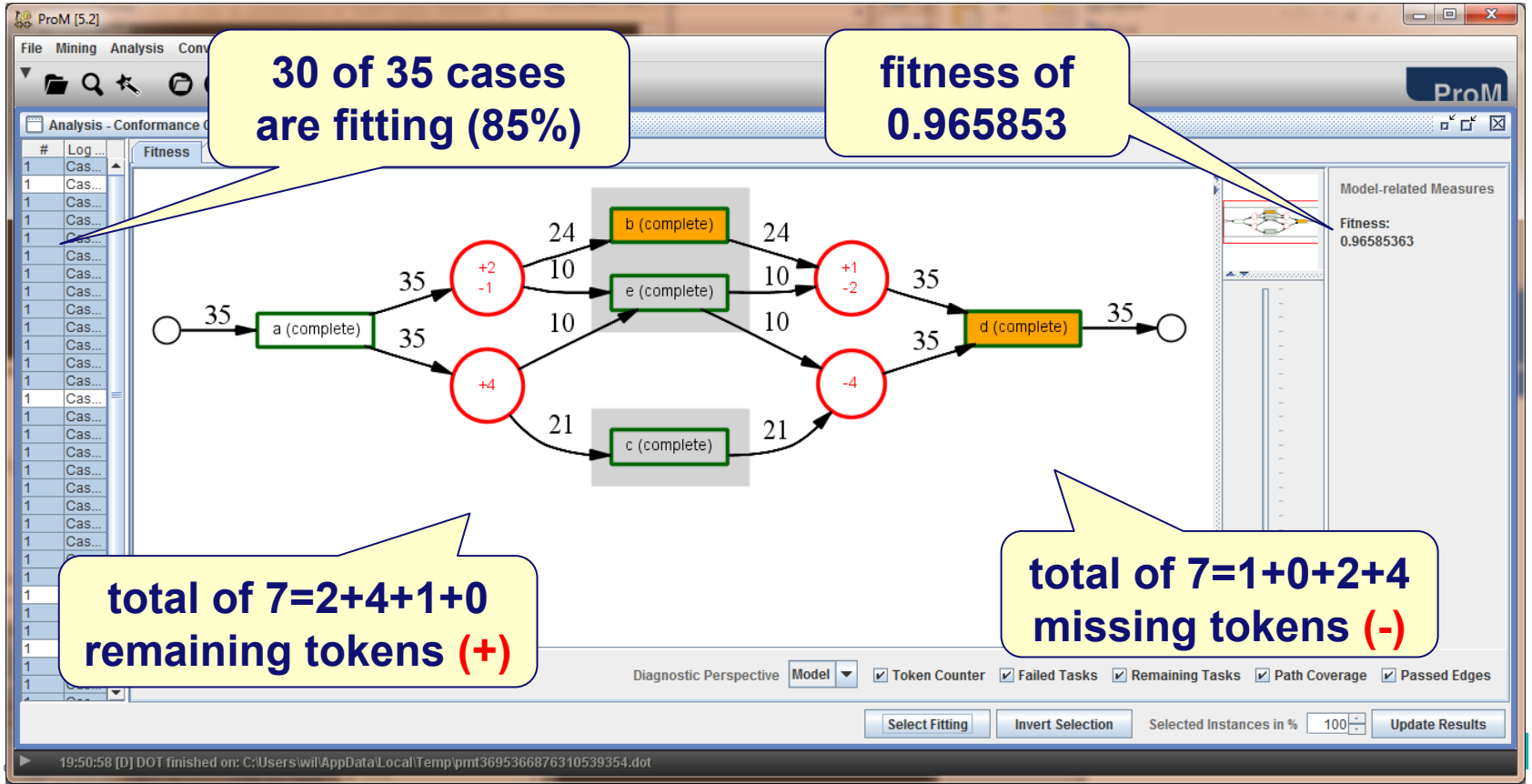
fitness	0.965853659

$$fitness(L,N) = \frac{1}{2}\left(1 - \frac{\sum_{\sigma \in L} L(\sigma) \times m_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times c_{N,\sigma}}\right) + \frac{1}{2}\left(1 - \frac{\sum_{\sigma \in L} L(\sigma) \times r_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times p_{N,\sigma}}\right)$$

ProM 5.2 output

(ProM 6 only supports more advanced conformance checking techniques like alignments)

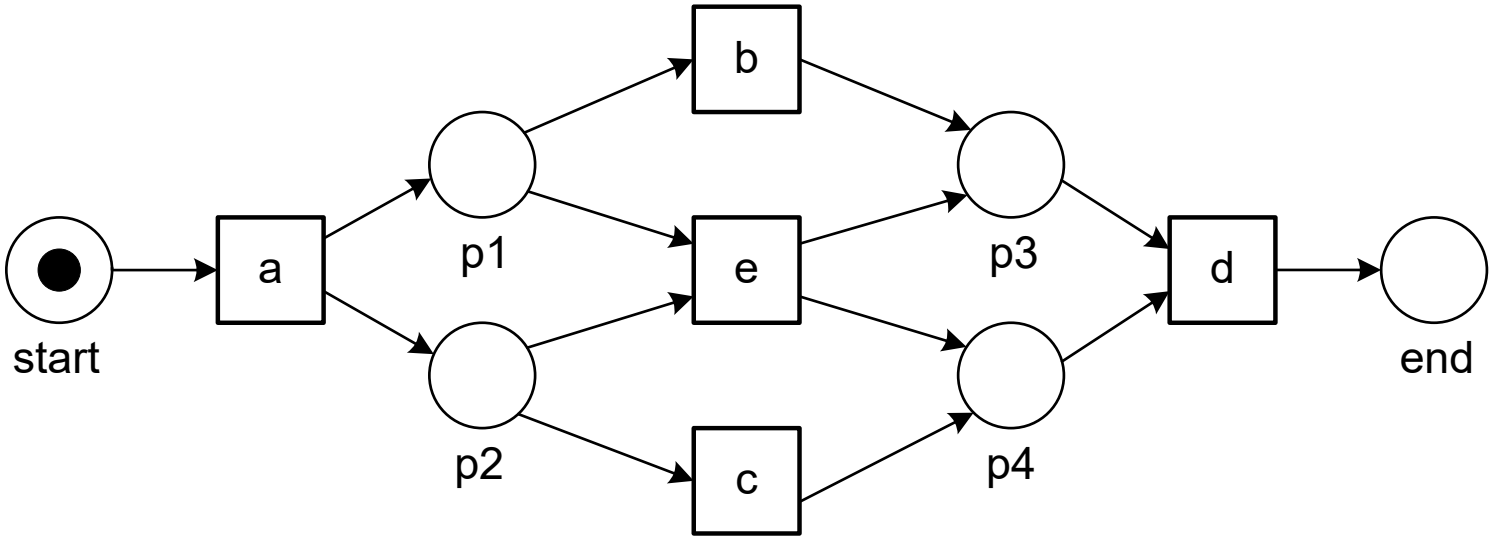
Note that PM4Py and Celonis support variants of token-based replay.



Question

Compute fitness using missing and remaining tokens

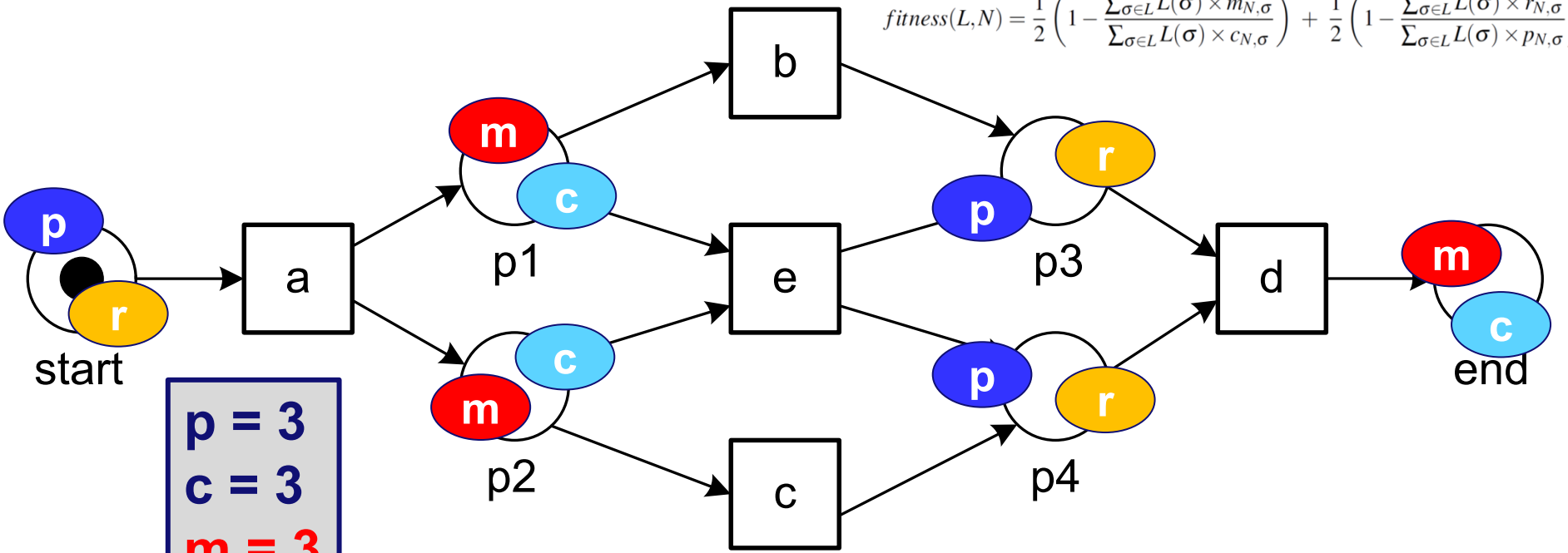
trace	frequency
e	1



- **Consider the event log containing just one case: L = [⟨e⟩].**
- **What is the fitness (using token-based replay)?**

Answer obtained by replaying ⟨e⟩

$$fitness(L,N) = \frac{1}{2}\left(1 - \frac{\sum_{\sigma \in L} L(\sigma) \times m_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times c_{N,\sigma}}\right) + \frac{1}{2}\left(1 - \frac{\sum_{\sigma \in L} L(\sigma) \times r_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times p_{N,\sigma}}\right)$$

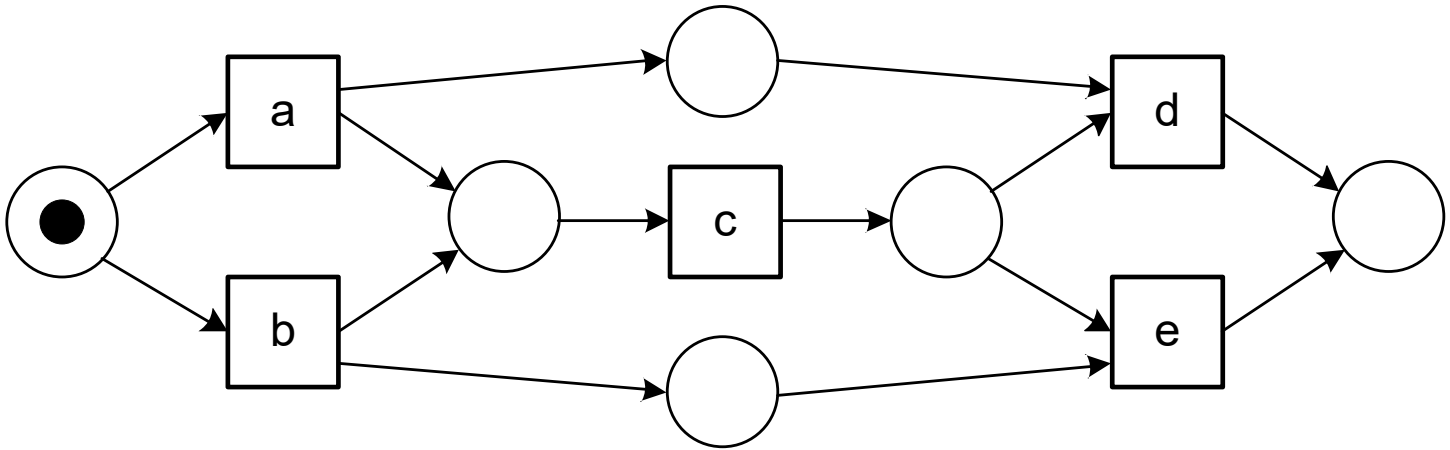


p = 3
c = 3
m = 3
r = 3

fitness = 0 !

Another example

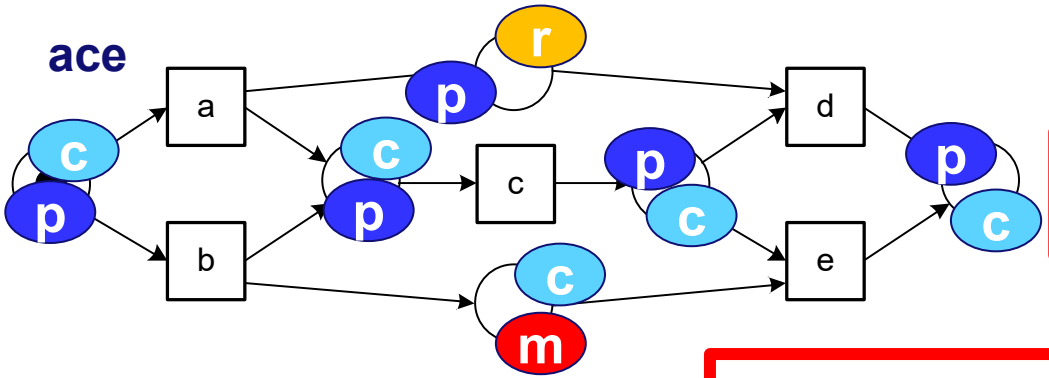
trace	frequency
acd	10
bce	10
ace	5
bcd	5
dca	1
abd	1
d	1



- **Consider the event log containing 33 cases.**
- **What is the fitness?**

Fitness = 0.895705521

trace	frequency	produced tokens (p)	remaining tokens (r)	consumed tokens (c)	missing tokens (m)	produced tokens (all)	remaining tokens (all)	consumed tokens (all)	missing tokens (all)
acd	10	5	0	5	0	50	0	50	0
bce	10	5	0	5	0	50	0	50	0
ace	5	5	1	5	1	25	5	25	5
bcd	5	5	1	5	1	25	5	25	5
dca	1	5	3	5	3	5	3	5	3
abd	1	6	3	5	2	6	3	5	2
d	1	2	1	3	2	2	1	3	2

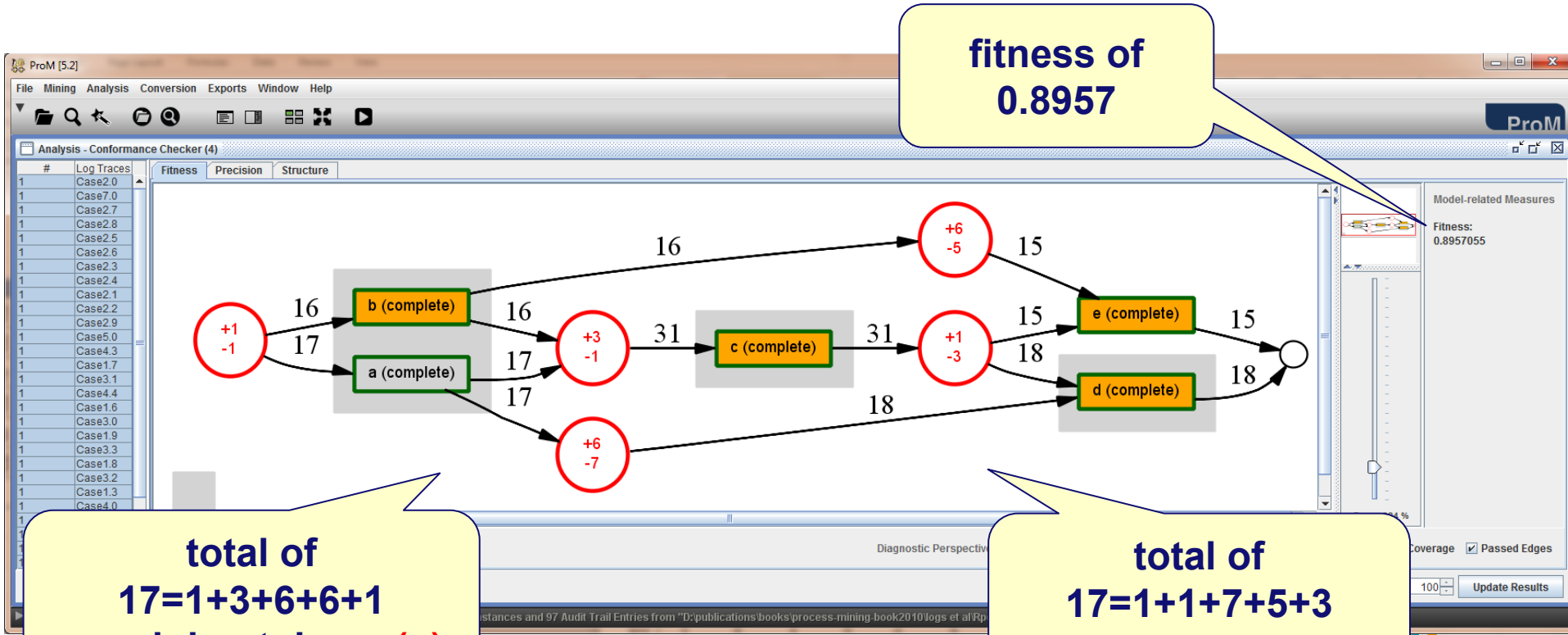


163	17	163	17
sum p	sum r	sum c	sum m

fitness **0.895705521**

$$fitness(L,N) = \frac{1}{2}\left(1 - \frac{\sum_{\sigma \in L} L(\sigma) \times m_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times c_{N,\sigma}}\right) + \frac{1}{2}\left(1 - \frac{\sum_{\sigma \in L} L(\sigma) \times r_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times p_{N,\sigma}}\right)$$

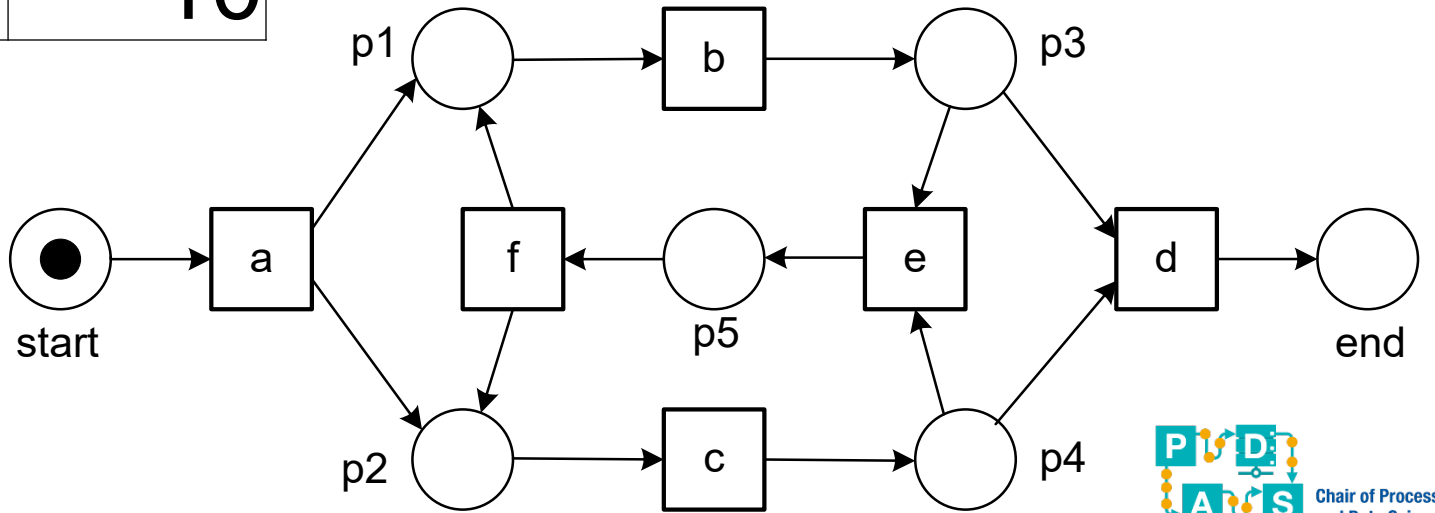
ProM 5.2 diagnostics



Another example

trace	frequency
abefcd	10
abbefccd	10

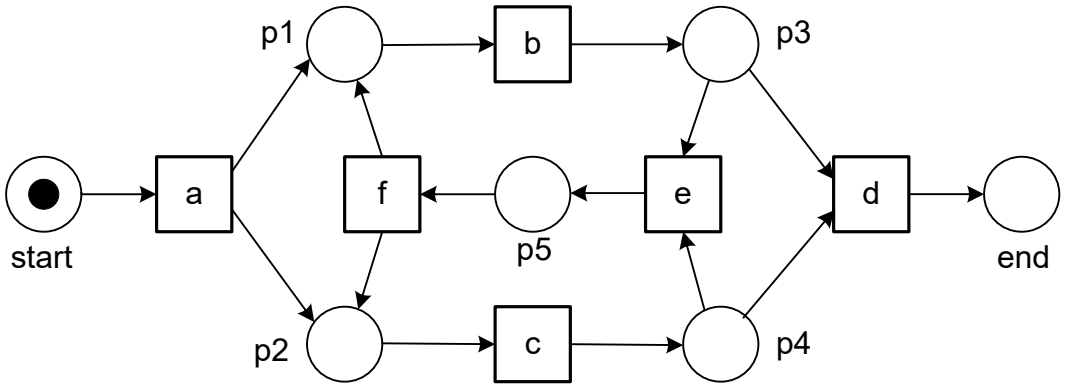
- Consider the event log containing 20 cases.
- What is the fitness?



Chair of Process and Data Science

Fitness = 0.8

trace	frequency	produced tokens (p)	remaining tokens (r)	consumed tokens (c)	missing tokens (m)	produced tokens (all)	remaining tokens (all)	consumed tokens (all)	missing tokens (all)
abefcd	10	9	2	9	2	90	20	90	20
abbefccd	10	11	2	11	2	110	20	110	20



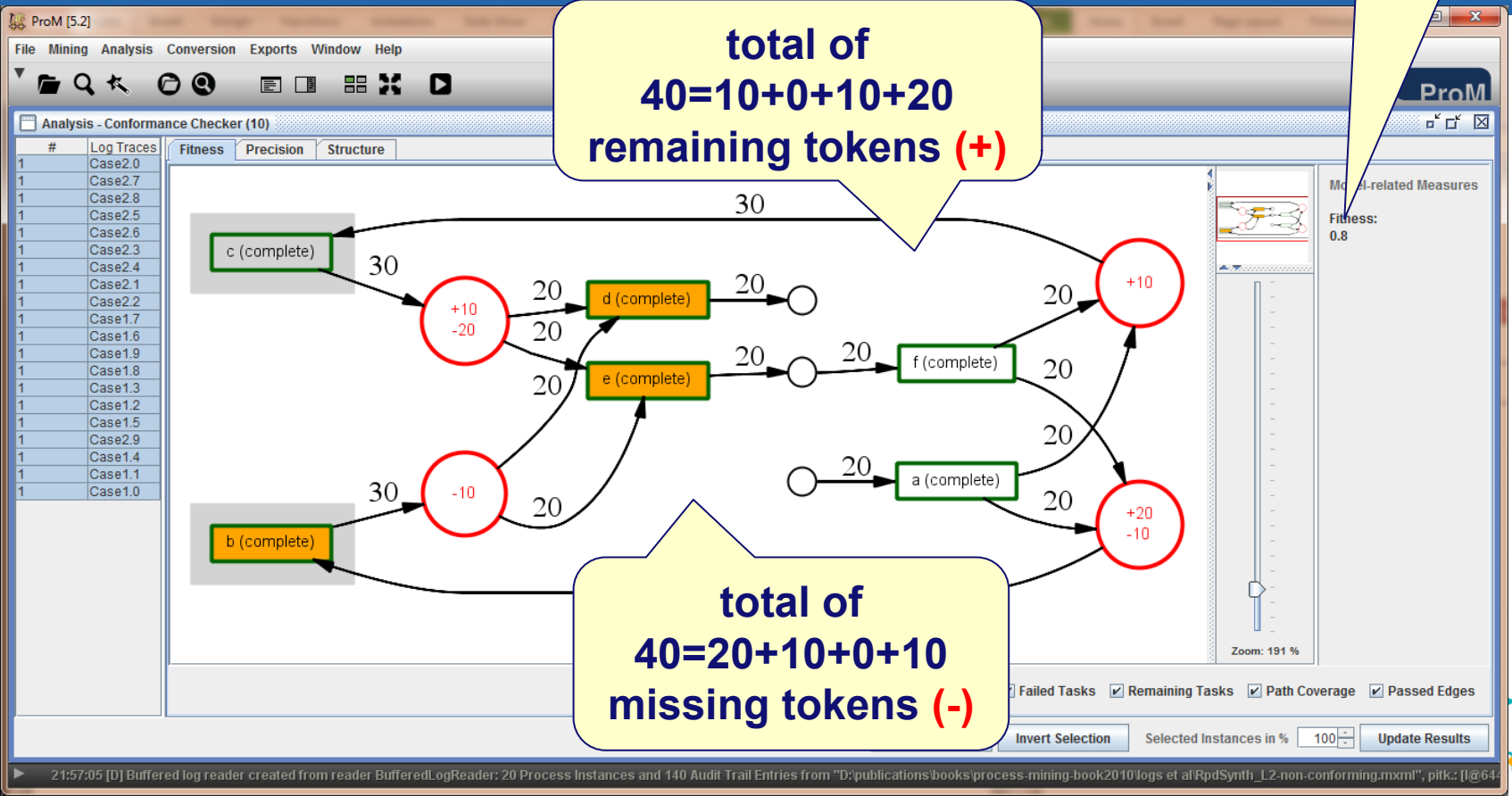
200	40	200	40
sum p	sum r	sum c	sum m

fitness **0.8**

$$fitness(L,N) = \frac{1}{2}\left(1 - \frac{\sum_{\sigma \in L} L(\sigma) \times m_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times c_{N,\sigma}}\right) + \frac{1}{2}\left(1 - \frac{\sum_{\sigma \in L} L(\sigma) \times r_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times p_{N,\sigma}}\right)$$

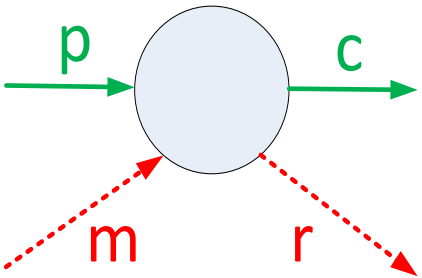
ProM 5.2 diagnostics

fitness of 0.8



Thus far c=p and m=r. Always?

- **Provide, if possible, a log and model such that c≠p and m≠r at end.**

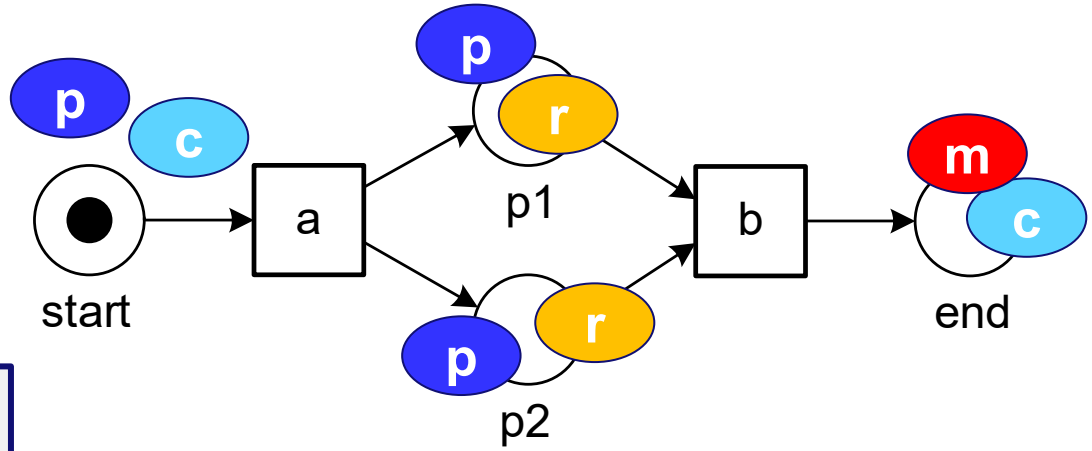


p = produced tokens
c = consumed tokens
m = missing tokens
(consumed while not there)
r = remaining tokens
(produced but not consumed)

Hint: Recall that r = p+m-c (i.e., p+m=c+r) at end.

Example: Model below and event log $[\langle a \rangle]$

r+c = p+m



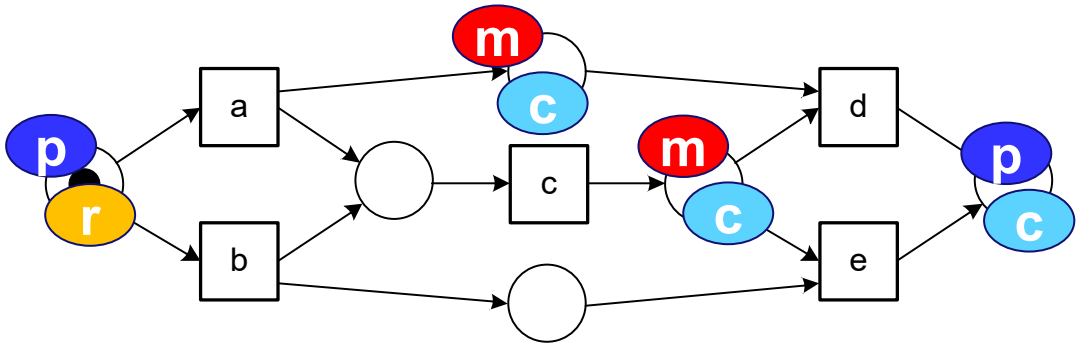
p=3 > c=2

p = 3
c = 2
m = 1
r = 2

$$fitness(\sigma, N) = \frac{1}{2}\left(1 - \frac{1}{2}\right) + \frac{1}{2}\left(1 - \frac{2}{3}\right) = 0.4166$$

Chair of Process and Data Science

Example: Model below and event log $[\langle d \rangle]$



r+c = p+m

p = 2
c = 3
m = 2
r = 1

p=2 < c=3

$$fitness(\sigma, N) = \frac{1}{2}\left(1 - \frac{2}{3}\right) + \frac{1}{2}\left(1 - \frac{1}{2}\right) = 0.4166$$

Chair of Process and Data Science

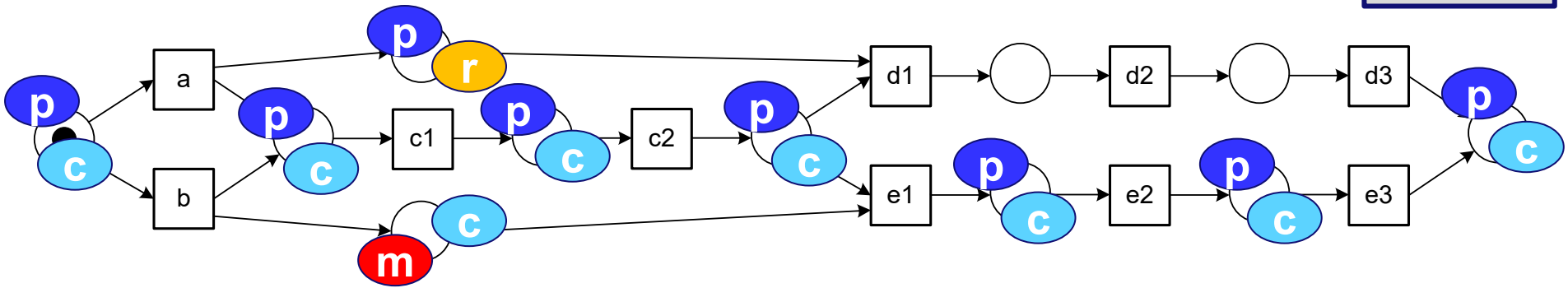
Limitations

- Basic replay approach assumes visible & uniquely labeled transitions.
- ProM implementation uses heuristics to deal with silent transitions and multiple transitions having the same label.
- Conformance values sometimes too optimistic due to "token flooding".
- Local decision making may lead to misleading results.

Local decision making is not enough …

p = 8
c = 8
m = 1
r = 1
f = 0.875

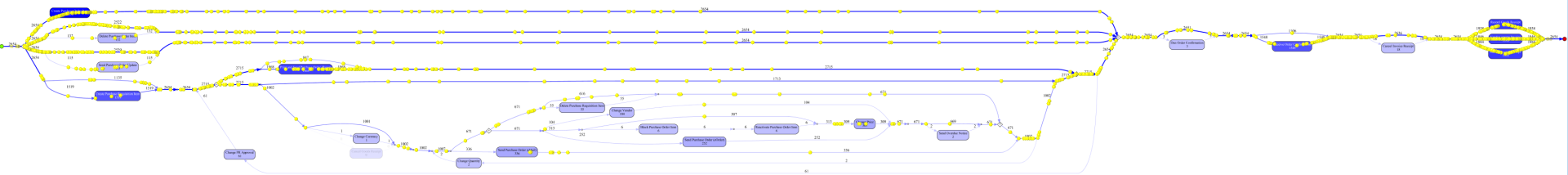
⟨a,c1,c2,e1,e2,e3⟩



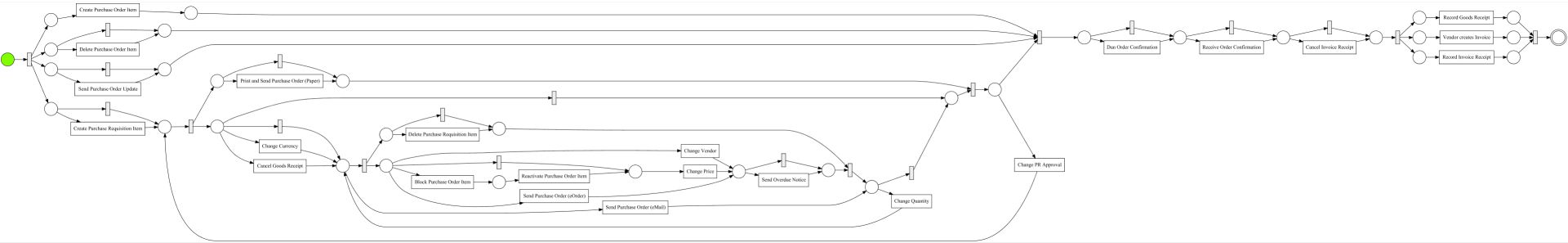
- Replay technique does not provide a corresponding path through the model (vital for conformance/performance analysis and other diagnostics).
- We would like to see the "closest path", i.e., ⟨b,c1,c2,e1,e2,e3⟩.

Chair of Process and Data Science

Challenge: Silent and duplicate transitions

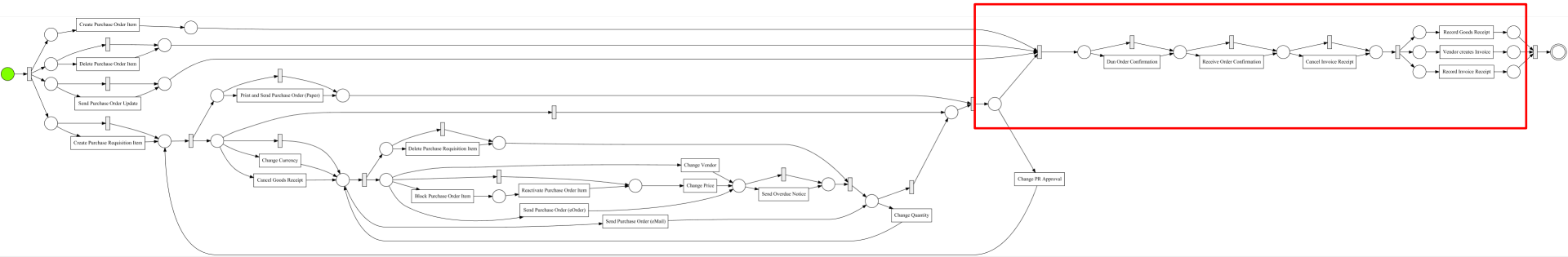


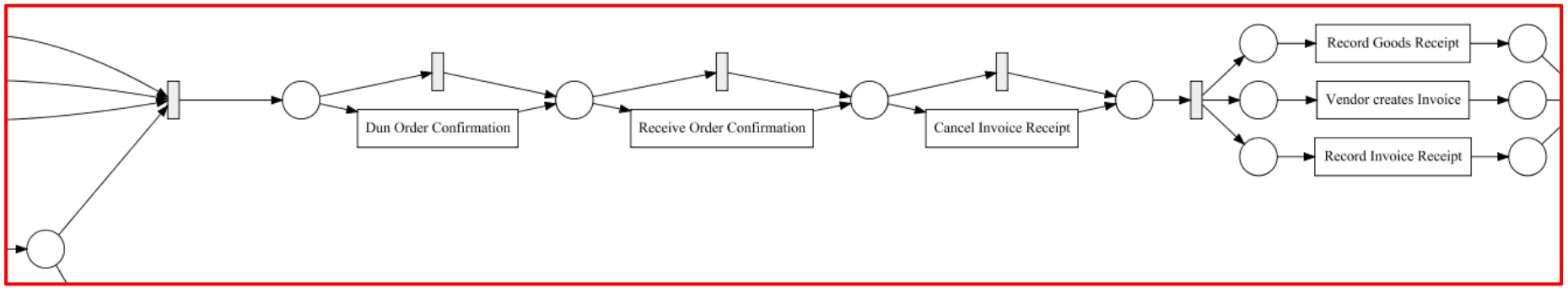
The Inductive Miner may produce many silent transitions …



Chair of Process and Data Science

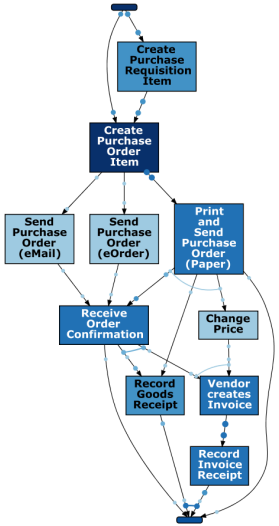
Challenge: Silent and duplicate transitions

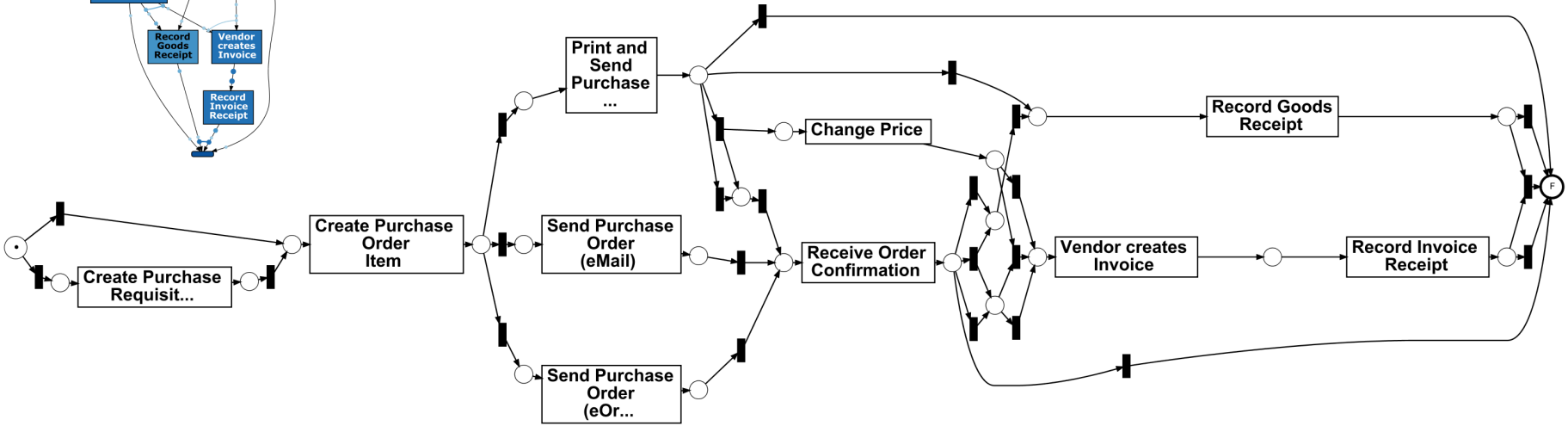




Challenge: Silent and duplicate transitions

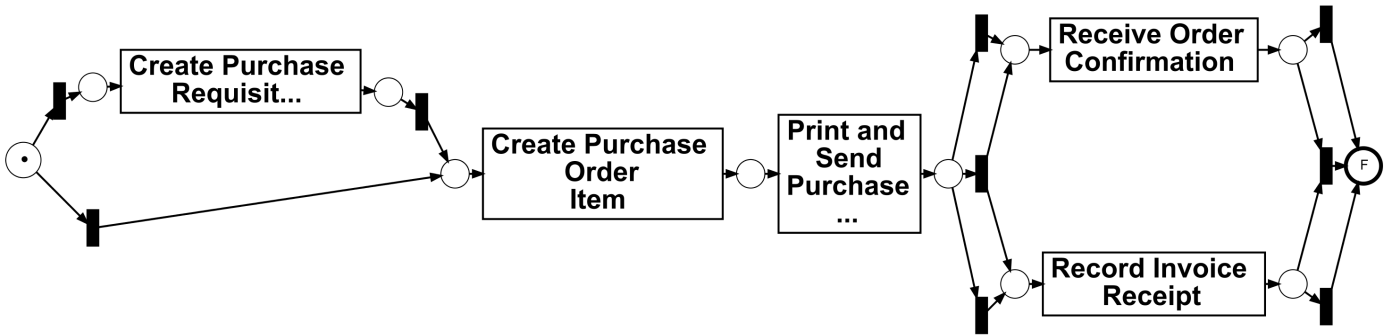
The Heuristic Miner may also produce many silent transitions



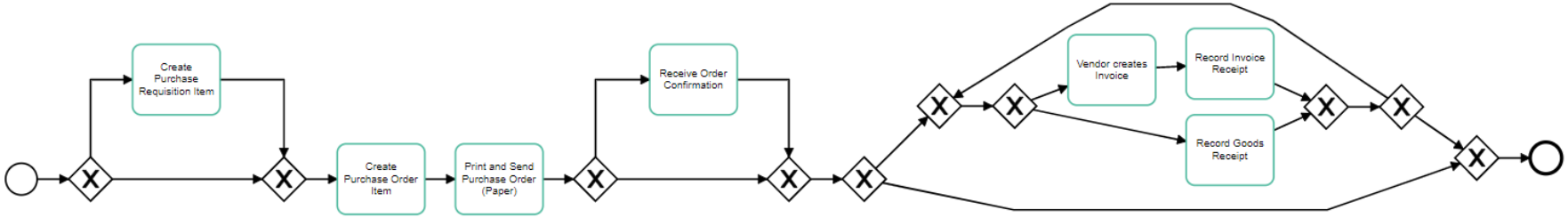


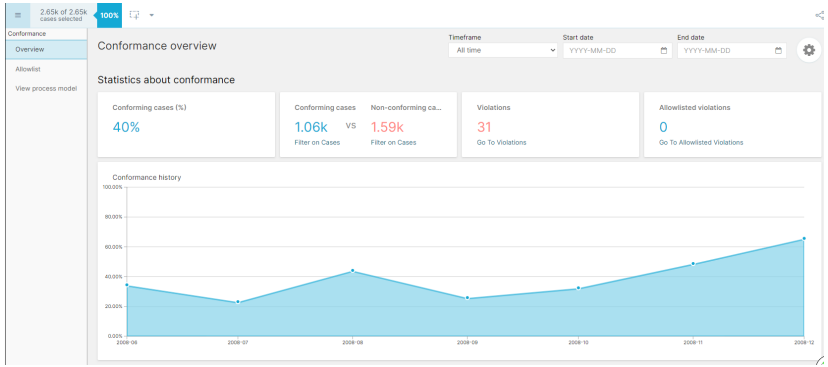
Challenge: Silent and duplicate transitions

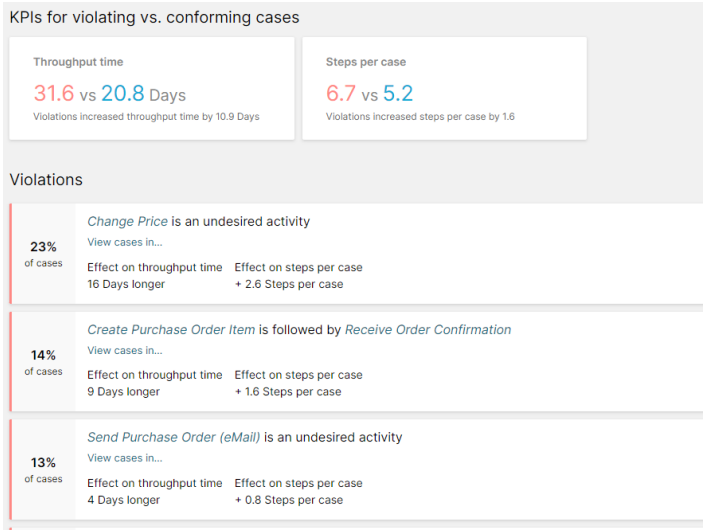
- Some form of state-space exploration is needed to decide which silent transitions to fire.
- If a trace is non-fitting, the set of possibilities grows further.



Celonis's classical conformance checker uses a variant of token-based replay

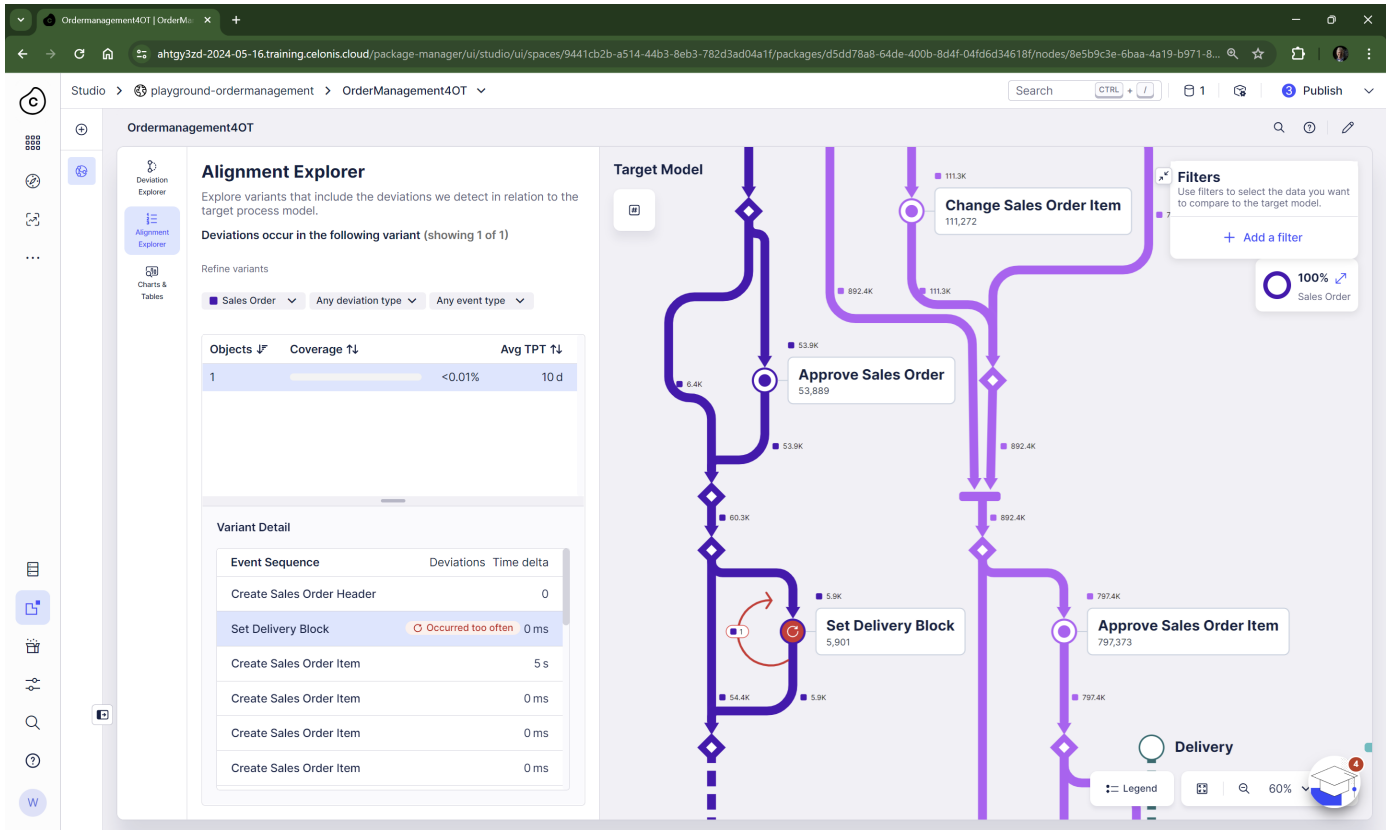






Chair of Process and Data Science

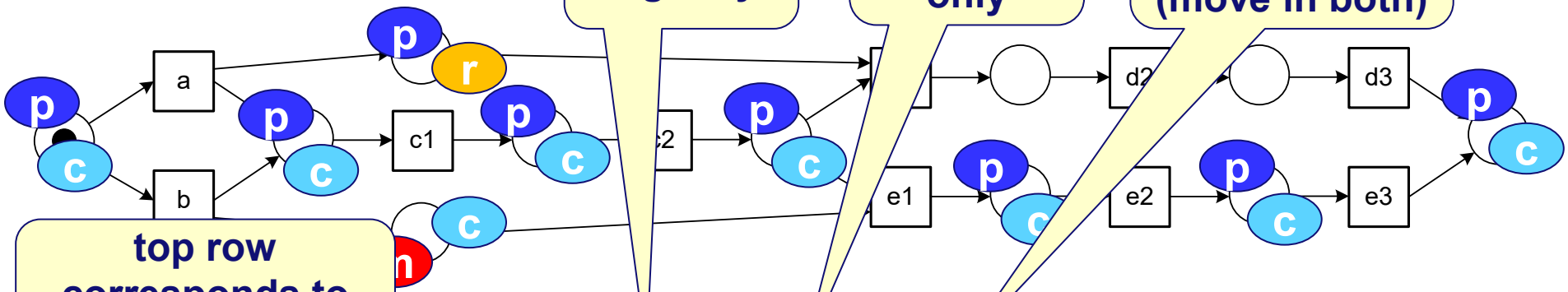
Newer capabilities of Celonis (e.g. PAM) use alignments



Chair of Process and Data Science

Next: alignments

⟨a,c1,c2,e1,e2,e3⟩



move in log only

move in model only

synchronous move (move in both)

top row corresponds to trace in event log

bottom row corresponds to path in model

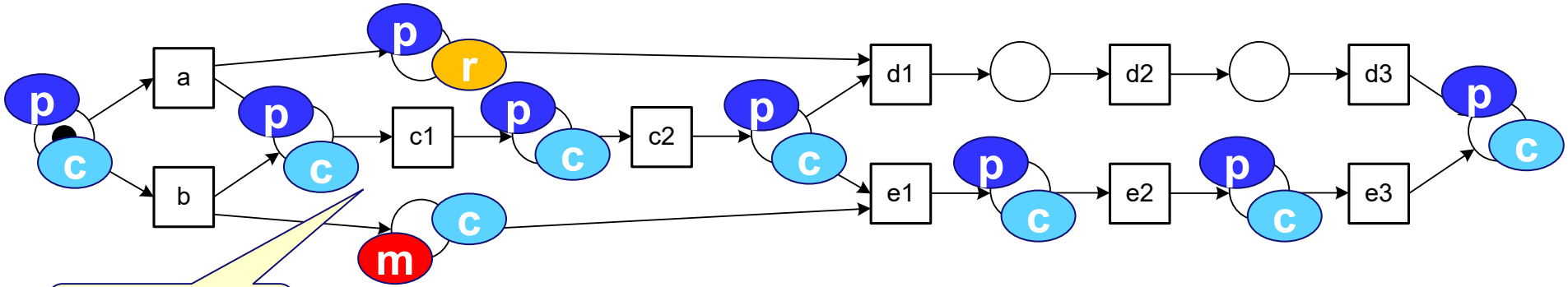
a	»	c1	c2	e1	e2	e3
»	b	c1	c2	e1	e2	e3

"no move"

"no move"

Alignments provide better diagnostics

⟨a,c1,c2,e1,e2,e3⟩



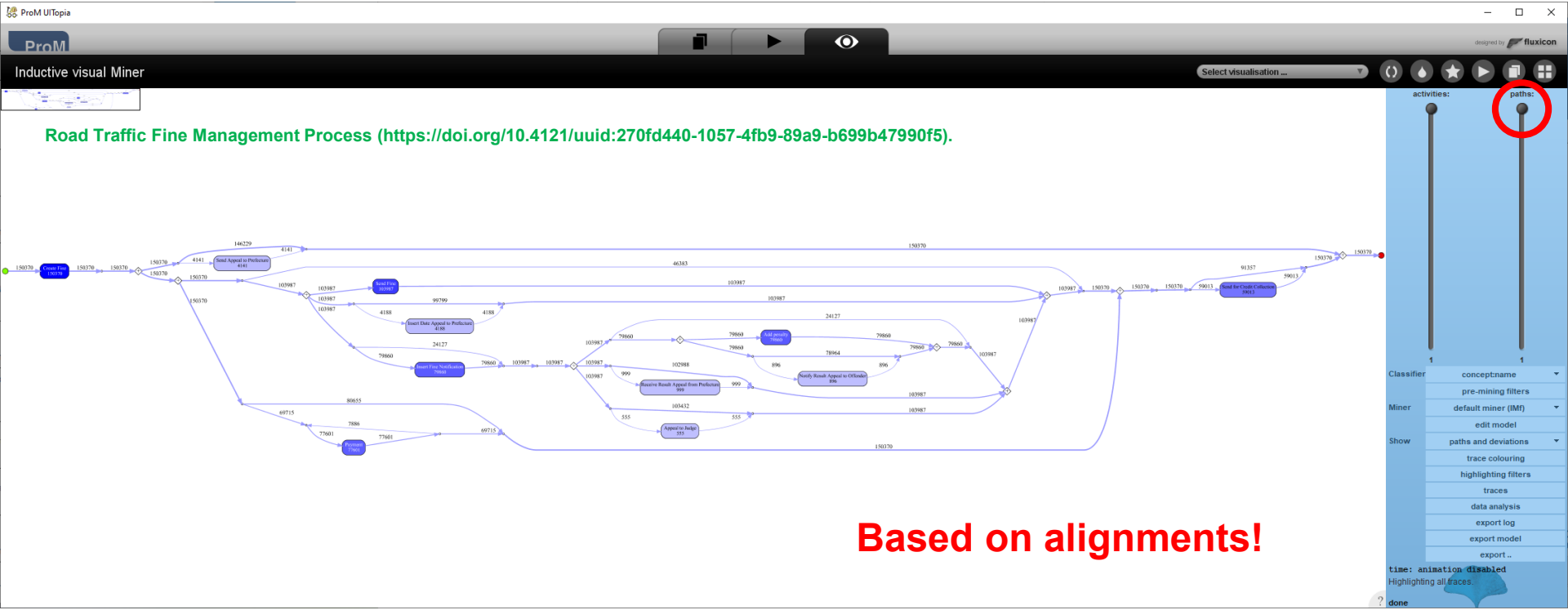
???

b was replaced by a

a	»	c1	c2	e1	e2	e3
»	b	c1	c2	e1	e2	e3

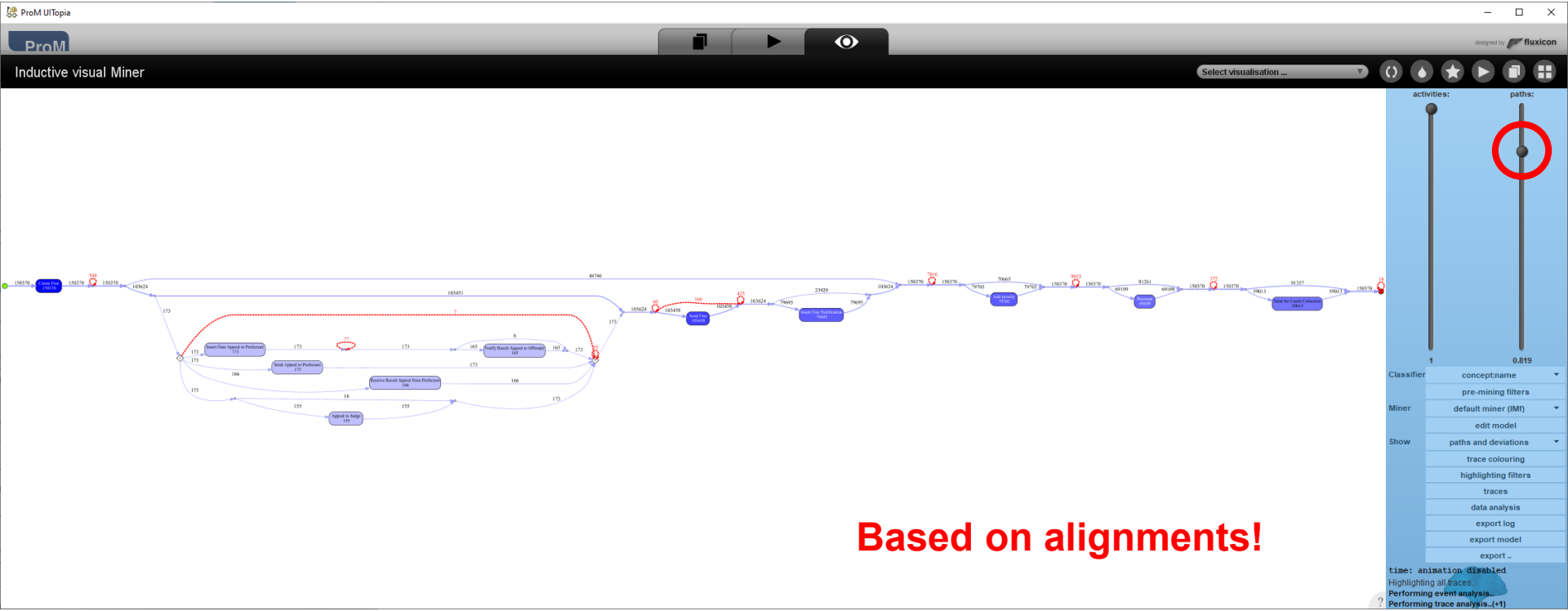
Chair of Process and Data Science

Example: Inductive Visual Miner



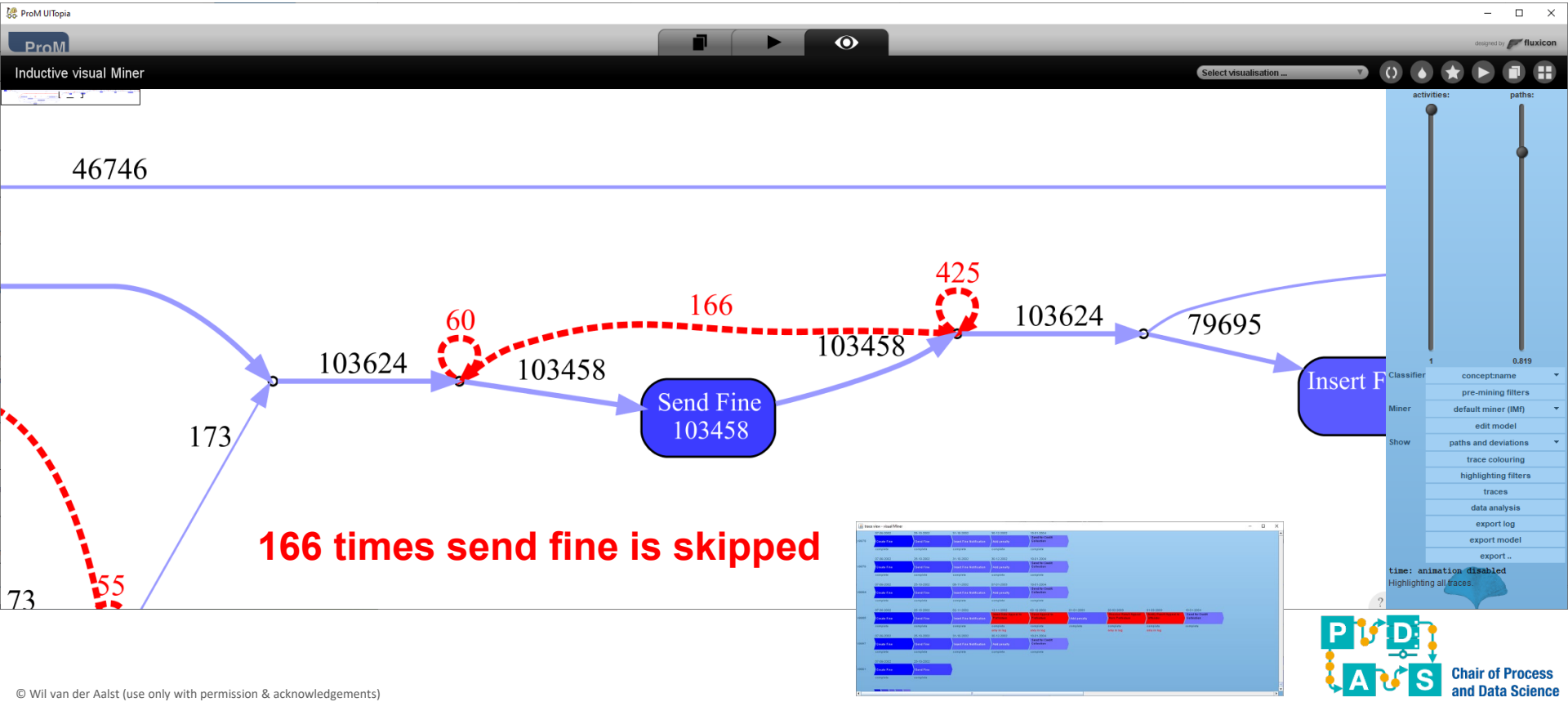
Chair of Process and Data Science

Example: Inductive Visual Miner



Chair of Process and Data Science

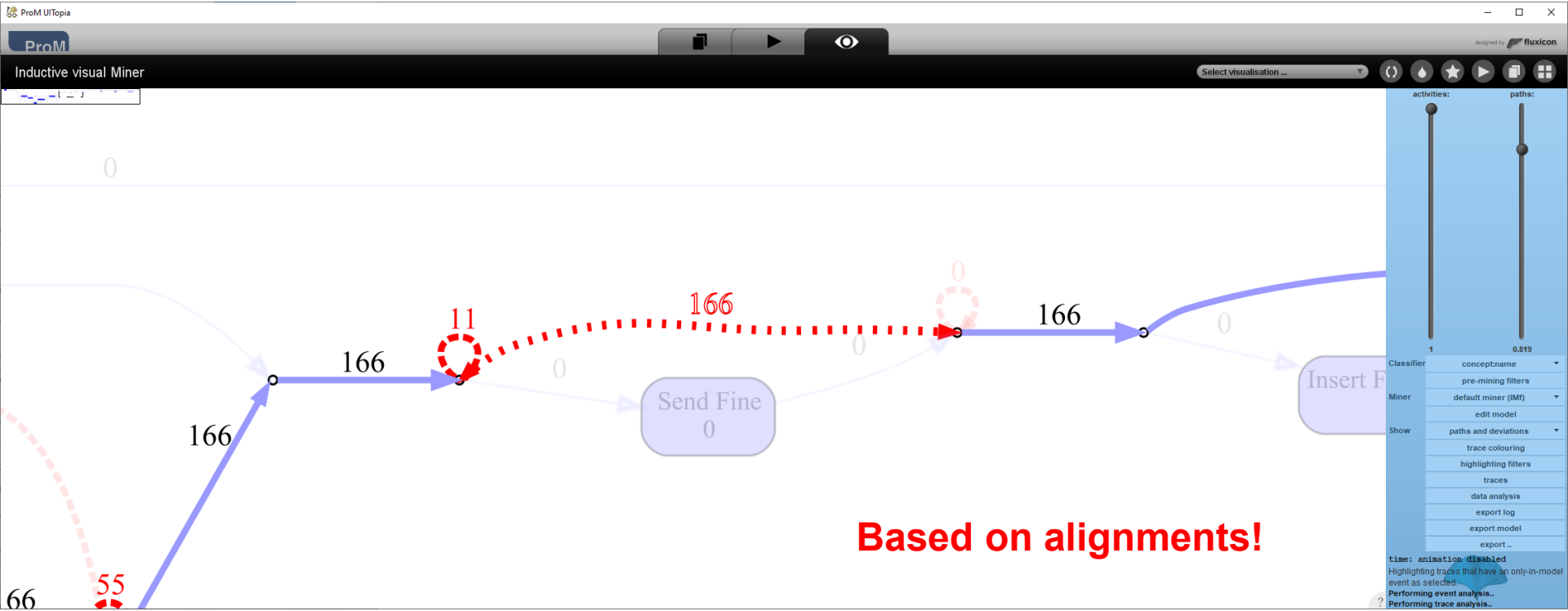
Example: Inductive Visual Miner



Chair of Process and Data Science

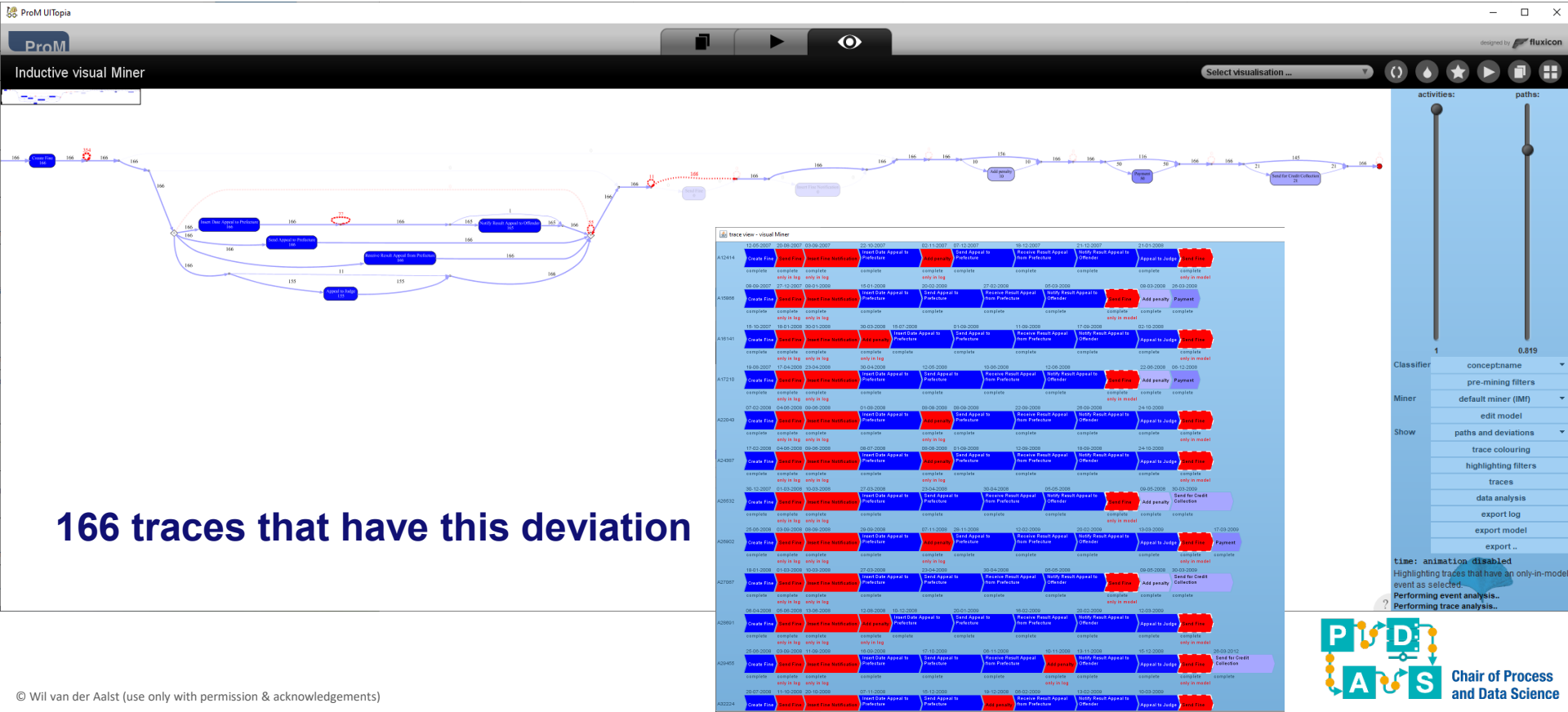
Example: Inductive Visual Miner

Example: Inductive Visual Miner



Chair of Process and Data Science

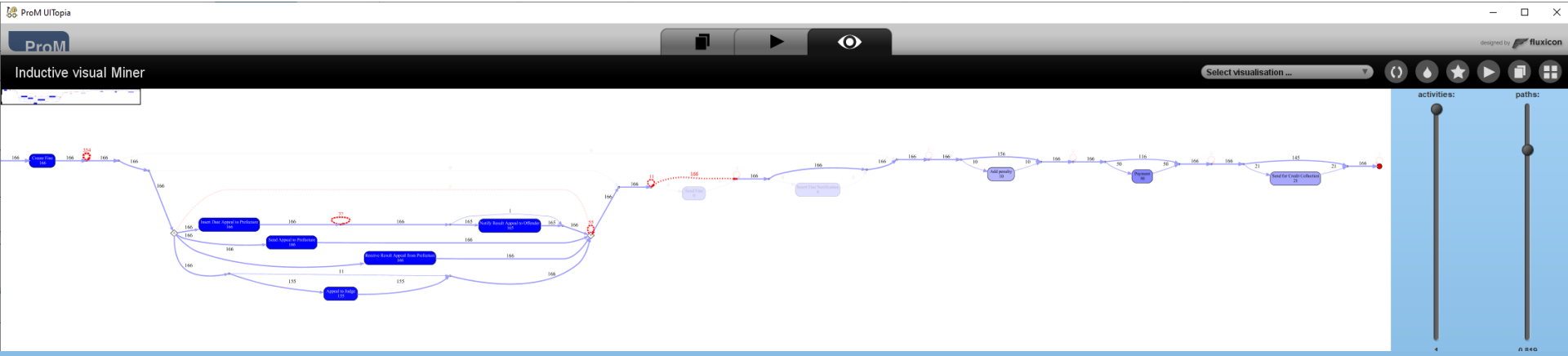
Example: Inductive Visual Miner



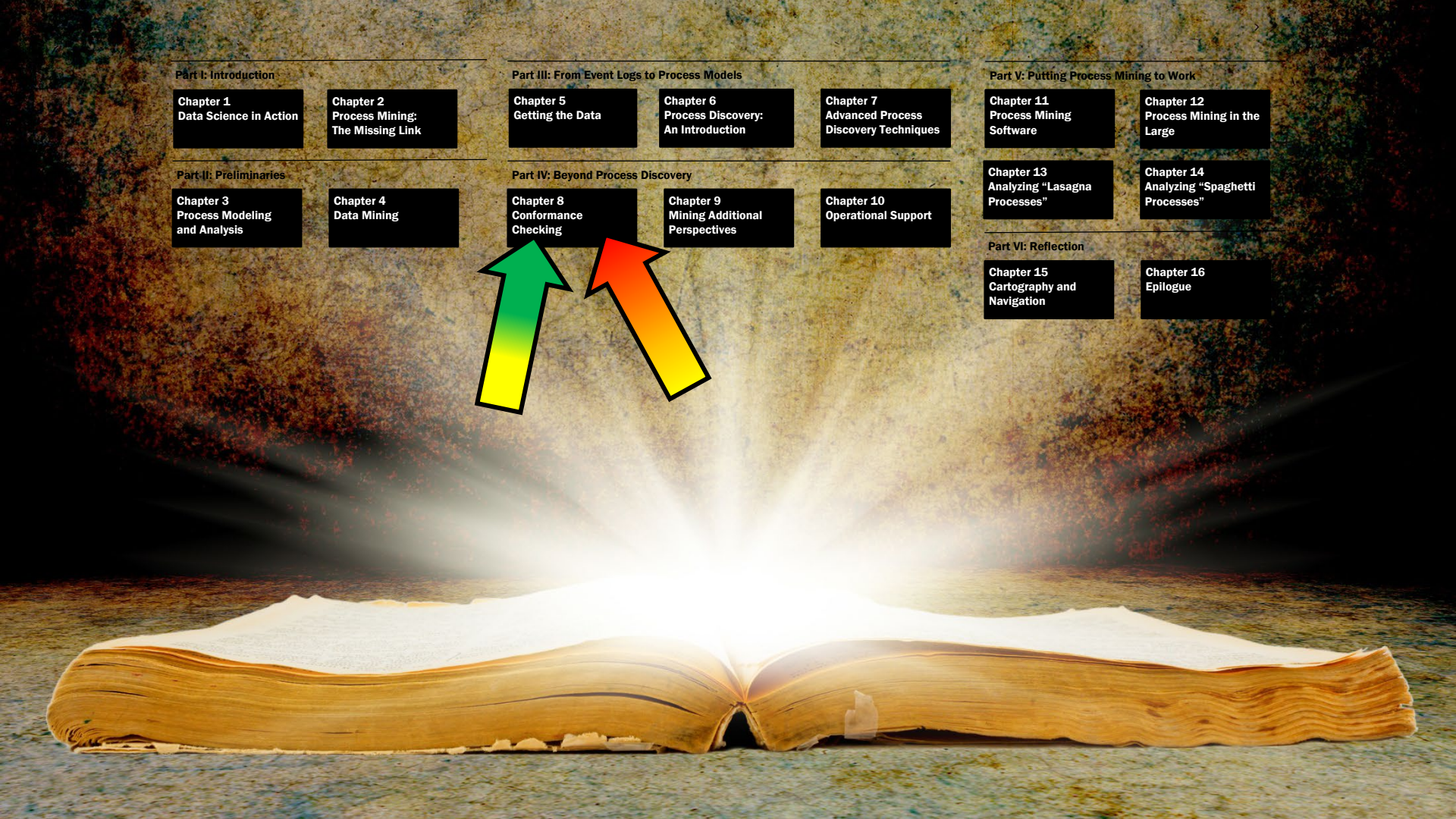
166 traces that have this deviation

Chair of Process and Data Science

Example: Inductive Visual Miner



03-03-2007	27-06-2007	06-07-2007	30-07-2007	24-08-2007	04-09-2007	26-11-2007	19-03-2008	28-03-2008		
Create Fine	Send Fine	Insert Fine Notification	Insert Date Appeal to Prefecture	Send Appeal to Prefecture	Add penalty	Receive Result Appeal from Prefecture	Notify Result Appeal to Offender	Appeal to Judge	Send Fine	
complete	complete	complete	complete	complete	complete	complete	complete	complete	complete	
	only in log	only in log			only in log				only in model	
17-09-2003	17-09-2003	17-09-2003	07-10-2003	27-10-2003	16-11-2003	15-12-2003	20-02-2004	21-02-2004	10-03-2005	12-04-2005		12-05-2005
Create Fine	Send Fine	Insert Fine Notification	Insert Date Appeal to Prefecture	Send Appeal to Prefecture	Add penalty	Receive Result Appeal from Prefecture	Notify Result Appeal to Offender	Appeal to Judge	Payment	Payment	Send Fine	Payment
complete	complete	complete	complete	complete	complete	complete	complete	complete	complete	complete	complete	complete
	only in log	only in log			only in log				only in log	only in log	only in model	
28-06-2006	16-11-2006	29-11-2006	27-12-2006	28-01-2007	02-02-2007	11-04-2007	19-04-2007	07-05-2007				
Create Fine	Send Fine	Insert Fine Notification	Insert Date Appeal to Prefecture	Add penalty	Send Appeal to Prefecture	Receive Result Appeal from Prefecture	Notify Result Appeal to Offender	Appeal to Judge	Send Fine			
complete	complete	complete	complete	complete	complete	complete	complete	complete	complete			
	only in log	only in log		only in log					only in model			


Part I: Introduction
Chapter 1
Data Science in Action
Chapter 2
Process Mining:
The Missing Link
Part II: Preliminaries
Chapter 3
Process Modeling
and Analysis
Chapter 4
Data Mining
Part III: From Event Logs to Process Models
Chapter 5
Getting the Data
Chapter 6
Process Discovery:
An Introduction
Chapter 7
Advanced Process
Discovery Techniques
Part IV: Beyond Process Discovery
Chapter 8
Conformance
Checking
Chapter 9
Mining Additional
Perspectives
Chapter 10
Operational Support
Part V: Putting Process Mining to Work
Chapter 11
Process Mining
Software
Chapter 12
Process Mining in the
Large
Chapter 13
Analyzing "Lasagna
Processes"
Chapter 14
Analyzing "Spaghetti
Processes"
Part VI: Reflection
Chapter 15
Cartography and
Navigation
Chapter 16
Epilogue

ID	Topic	Date	Date	Place
Lecture 1	Introduction to Process Mining	08.04.24	Monday	AH V
Lecture 2	Data Science: Supervised Learning	09.04.24	Tuesday	AH V
Exercise 1	*Tool Introduction*	09.04.24	Tuesday	AH III
Lecture 3	Data Science: Unsupervised Learning and Evaluation	15.04.24	Monday	AH V
Lecture 4	Introduction to Process Discovery	16.04.24	Tuesday	AH V
Exercise 2	*Data Mining*	16.04.24	Tuesday	AH III
Lecture 5	Alpha Algorithm 1	22.04.24	Monday	AH V
Lecture 6	Alpha Algorithm 2	23.04.24	Tuesday	AH V
Exercise 3	*Petri Nets*	23.04.24	Tuesday	AH III
Lecture 7	Model Quality Representation	29.04.24	Monday	AH V
Lecture 8	Heuristic Mining	30.04.24	Tuesday	AH V
Exercise 4	*Alpha Miner*	30.04.24	Tuesday	AH III
Lecture 9	Region-Based Mining	06.05.24	Monday	AH V
Exercise 5	*Heuristic Mining and Region-Based Mining*	07.05.24	Tuesday	AH III
Lecture 10	Inductive Mining	13.05.24	Monday	AH V
Lecture 11	Event Data and Exploration	14.05.24	Tuesday	AH V
Exercise 6	*Inductive Mining*	14.05.24	Tuesday	AH III
Lecture 12	Conformance Checking 1	27.05.24	Monday	AH V
Lecture 13	Conformance Checking 2	28.05.24	Tuesday	AH V
	Q&A Session Assignment Part I	28.05.24	Tuesday	AH III
	Deadline Assignment Part I	02.06.24	Sunday	
Exercise 7	*Footprint and Token-Based Replay (Exercise)*	03.06.24	Monday	AH V
Exercise 8	*Alignments (Exercise)*	04.06.24	Tuesday	AH V
Lecture 14	Decision Mining	10.06.24	Monday	AH V
Lecture 15	*Celonis Guest Lecture*	11.06.24	Tuesday	AH V
Exercise 9	*Decision Mining*	11.06.24	Tuesday	AH III
Lecture 16	Performance Analysis and Organizational Mining	17.06.24	Monday	AH V
Exercise 10	*Performance Analysis (Exercise)*	18.06.24	Tuesday	AH V
Exercise 11	*Organizational Mining*	18.06.24	Tuesday	AH III
Exercise 12	*Celonis Case Study*	24.06.24	Monday	AH V
Lecture 17	Operational Support and Process Mining Applications	01.07.24	Monday	AH V
Lecture 18	Distributed, Streaming, and Comparative Process Mining	02.07.24	Tuesday	AH V
Exercise 13	*Operational Process Mining*	02.07.24	Tuesday	AH III
Lecture 19	Closing	08.07.24	Monday	AH V
	Q&A Session Assignment Part II	09.07.24	Tuesday	AH III
	Deadline Assignment Part II	14.07.24	Sunday	
	Q&A Session Exam	16.07.24	Tuesday	AH III

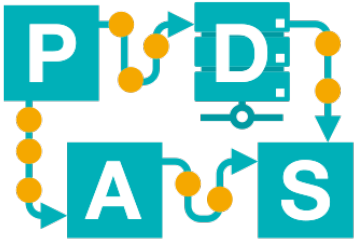
Chair of Process and Data Science

Conformance Checking (2/2)

Lecture 13

prof.dr.ir. Wil van der Aalst
www.vdaalst.com @wvdaalst
www.pads.rwth-aachen.de

BPI-L13



Chair of Process and Data Science

RWTH AACHEN UNIVERSITY

Outline

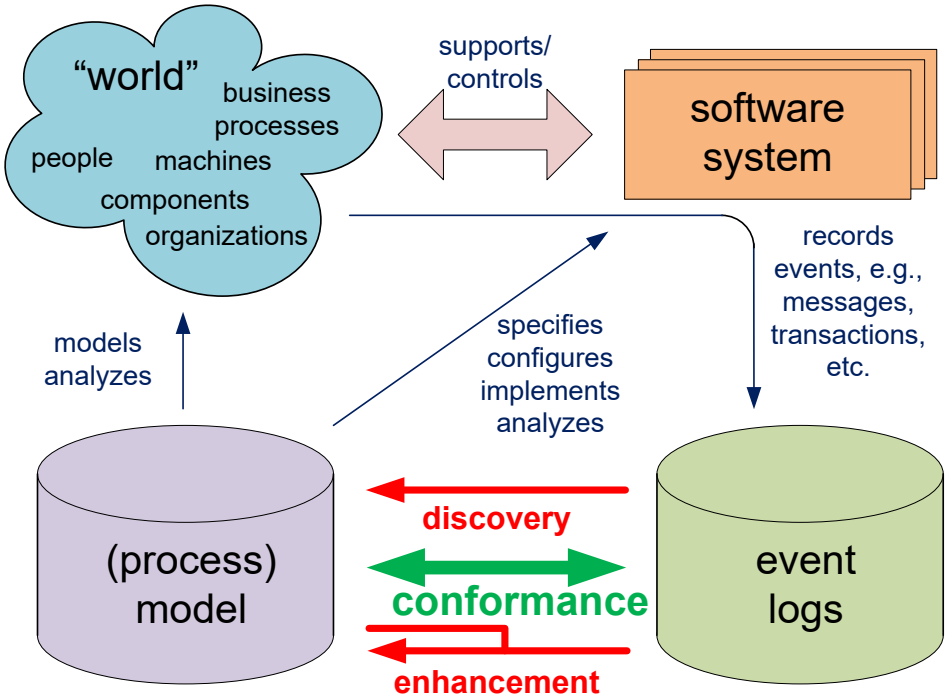
- A few end-to-end examples using toke-based replay.
- Alignment-based conformance checking.
- Tool support for conformance checking.
- Applications of process mining.

Chair of Process and Data Science

RWTH AACHEN UNIVERSITY

Token-based replay revisited

Conformance checking



1. Conformance checking using causal footprints.
2. **Conformance checking based on token-based replay.**
3. Alignment-based conformance checking.

Chair of Process and Data Science

Last lecture: Token-based replay

#	trace
455	acdeh
191	abdeg
177	adceh
144	abdeh
111	acdeg
82	adceg
56	adbeh
47	acdefdbeh
38	adbeg
33	acdefbdeh
14	acdefbdeg
11	acdefdbeg
9	adcefcdeh
8	adcefdbeh
5	adcefbdeg
3	acdefbdefdbeg
2	adcefdbeg
2	adcefbdefbdeg
1	adcefdbefbdeh
1	adbefbdefdbeg
1	adcefdbefcdefdbeg
1391	

missing tokens

?

remaining tokens

$$fitness(L,N) = \frac{1}{2}\left(1 - \frac{\sum_{\sigma \in L} L(\sigma) \times m_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times c_{N,\sigma}}\right) + \frac{1}{2}\left(1 - \frac{\sum_{\sigma \in L} L(\sigma) \times r_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times p_{N,\sigma}}\right)$$

consumed tokens

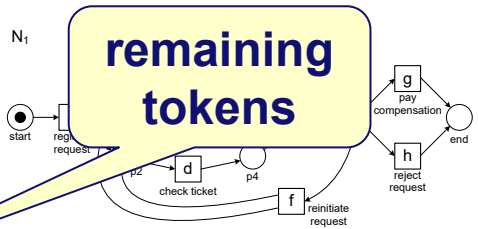
produced tokens

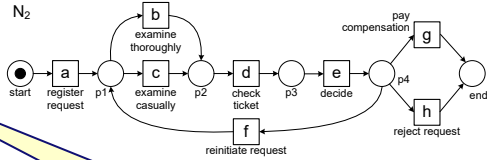
$$fitness(L_{full}, N_1) = 1$$

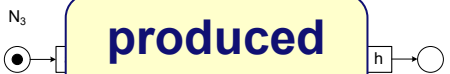
$$fitness(L_{full}, N_2) = 0.9504$$

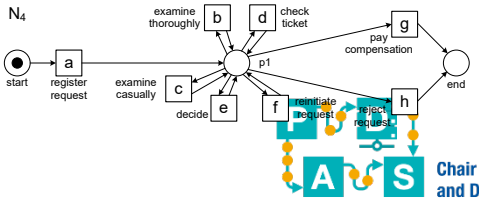
$$fitness(L_{full}, N_3) = 0.8797$$

$$fitness(L_{full}, N_4) = 1$$









Chair of Process and Data Science

End-to-end examples

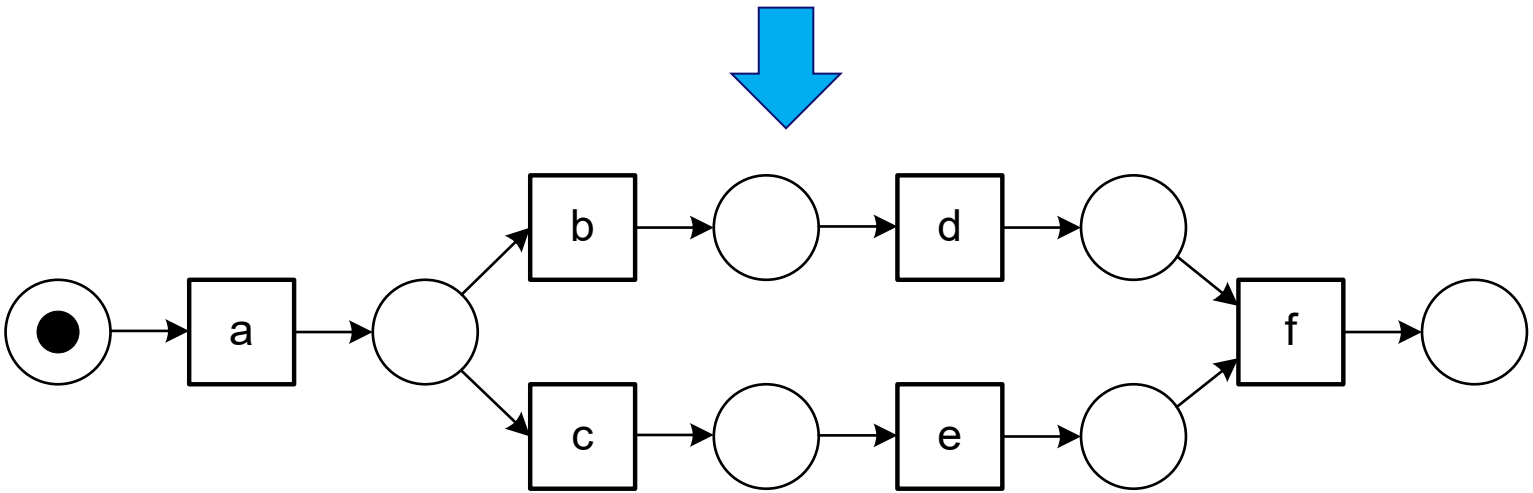
Question

$$L = [\langle a, b, d, e, f \rangle^{10}, \langle a, c, e, d, f \rangle^{10}]$$

- Consider the above event log.
- Give the model that the Alpha algorithm generates.
- Compute fitness using missing and remaining tokens.
- Comment on the findings.

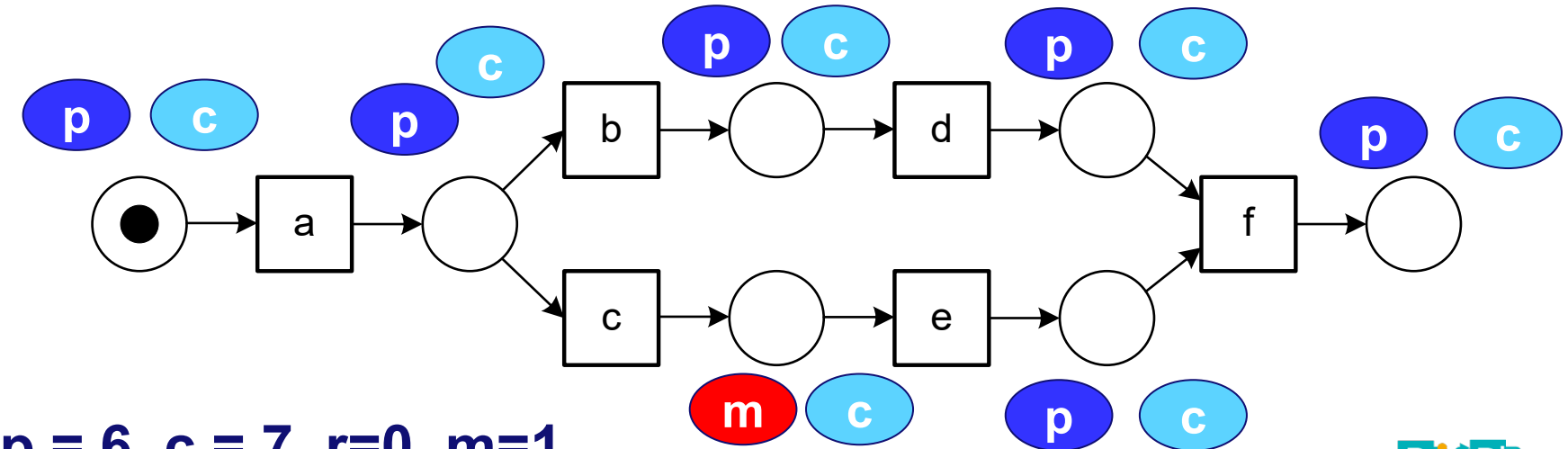
Model generated by the Alpha Algorithm

$$L = [\langle a, b, d, e, f \rangle^{10}, \langle a, c, e, d, f \rangle^{10}]$$



Trace abdef

$$L = [\langle a, b, d, e, f \rangle^{10}, \langle a, c, e, d, f \rangle^{10}]$$

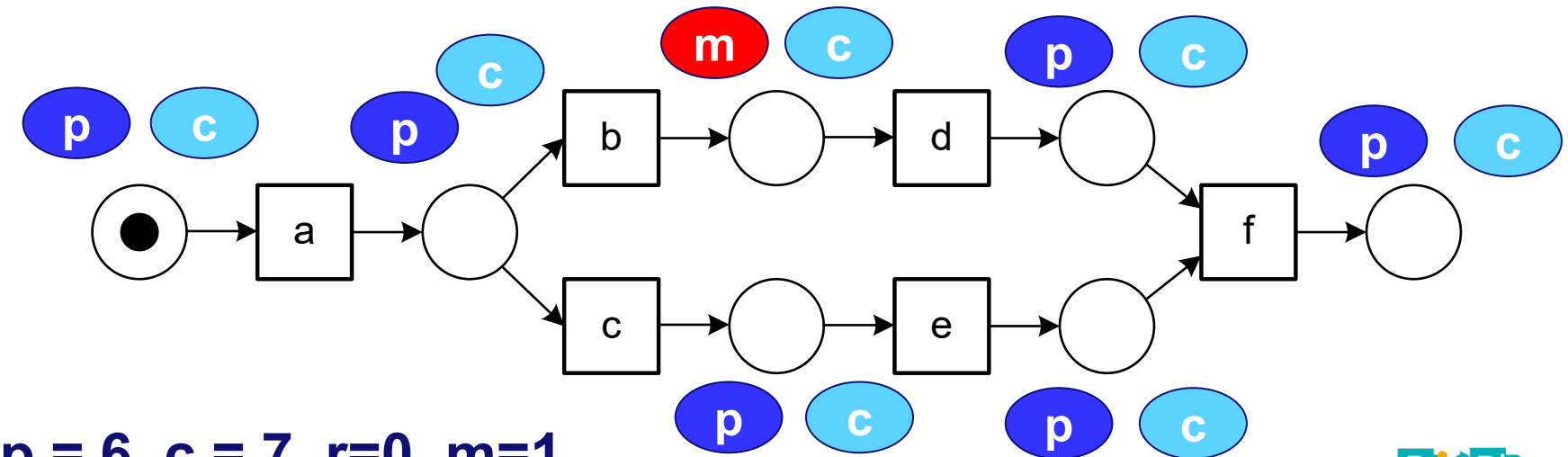


p = 6, c = 7, r=0, m=1

Chair of Process and Data Science

Trace acedf

$$L = [\langle a, b, d, e, f \rangle^{10}, \langle a, c, e, d, f \rangle^{10}]$$



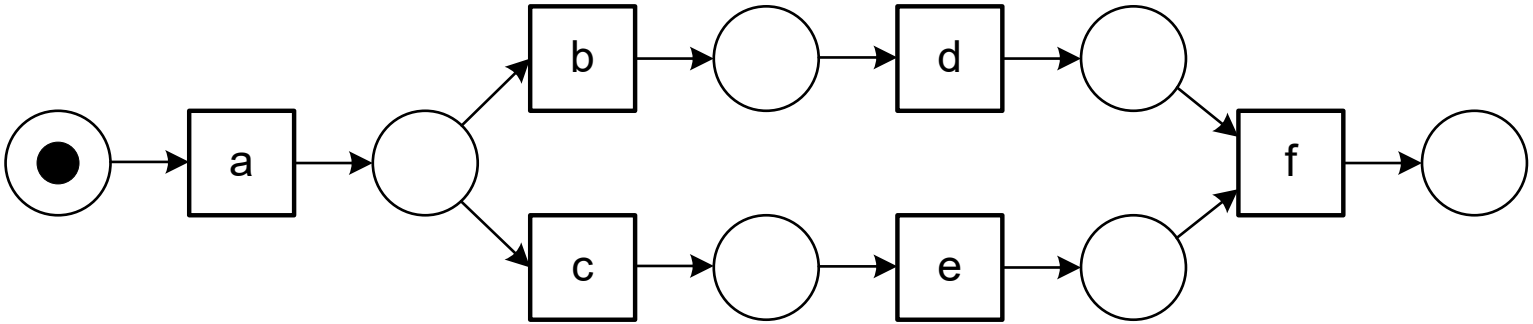
p = 6, c = 7, r=0, m=1

Chair of Process and Data Science

Overall fitness

$$fitness(L,N) = \frac{1}{2}\left(1 - \frac{\sum_{\sigma \in L} L(\sigma) \times m_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times c_{N,\sigma}}\right) + \frac{1}{2}\left(1 - \frac{\sum_{\sigma \in L} L(\sigma) \times r_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times p_{N,\sigma}}\right)$$

$$L = [\langle a, b, d, e, f \rangle^{10}, \langle a, c, e, d, f \rangle^{10}]$$

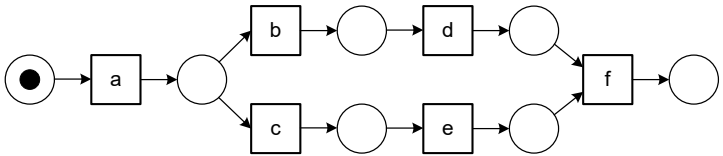


p = 2*10*6 =120, c = 2*10*7=140, r=2*10*0=0, m=2*10*1=20

$$\frac{1}{2}\left(1-\frac{20}{140}\right)+\frac{1}{2}\left(1-\frac{0}{120}\right)=\frac{13}{14}\approx 0.93$$

Model is not sound!

$$L = [\langle a, b, d, e, f \rangle^{10}, \langle a, c, e, d, f \rangle^{10}]$$



$$\frac{1}{2}\left(1-\frac{20}{140}\right)+\frac{1}{2}\left(1-\frac{0}{120}\right)=\frac{13}{14}\approx 0.93$$

- The model is not sound. Actually there is no firing sequence leading to the target marking!
- How to interpret the result? Therefore, we typically require “relaxed soundness”, i.e., there is at least one firing sequence leading to the target marking.

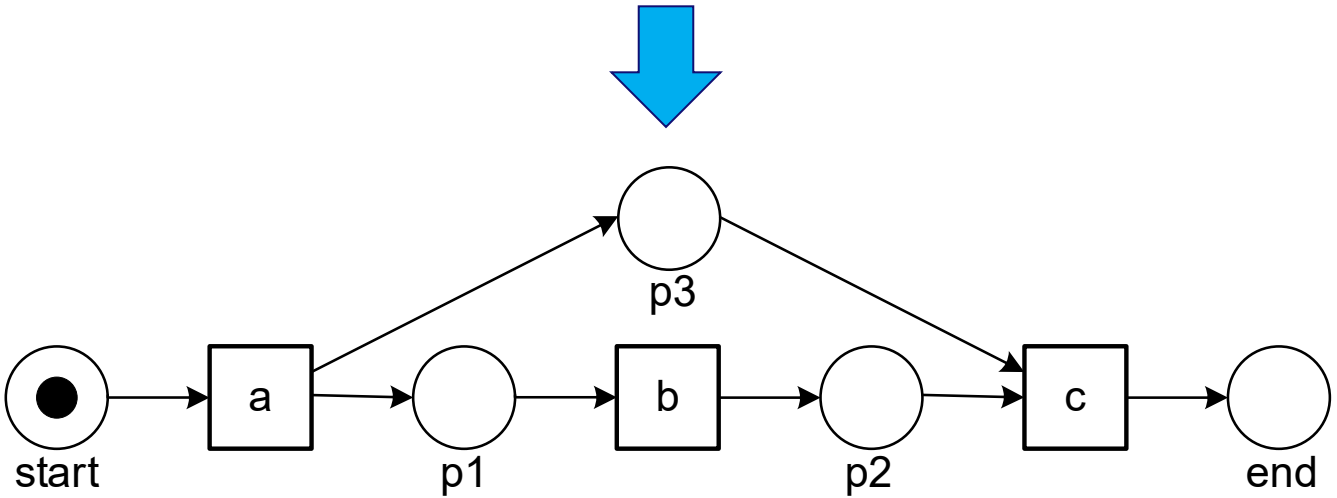
Question

$$L_{11} = [\langle a, b, c \rangle^{20}, \langle a, c \rangle^{30}]$$

- Consider the above event log.
- Give the model that the Alpha algorithm generates
- Compute fitness using missing and remaining tokens.
- Comment on the findings.

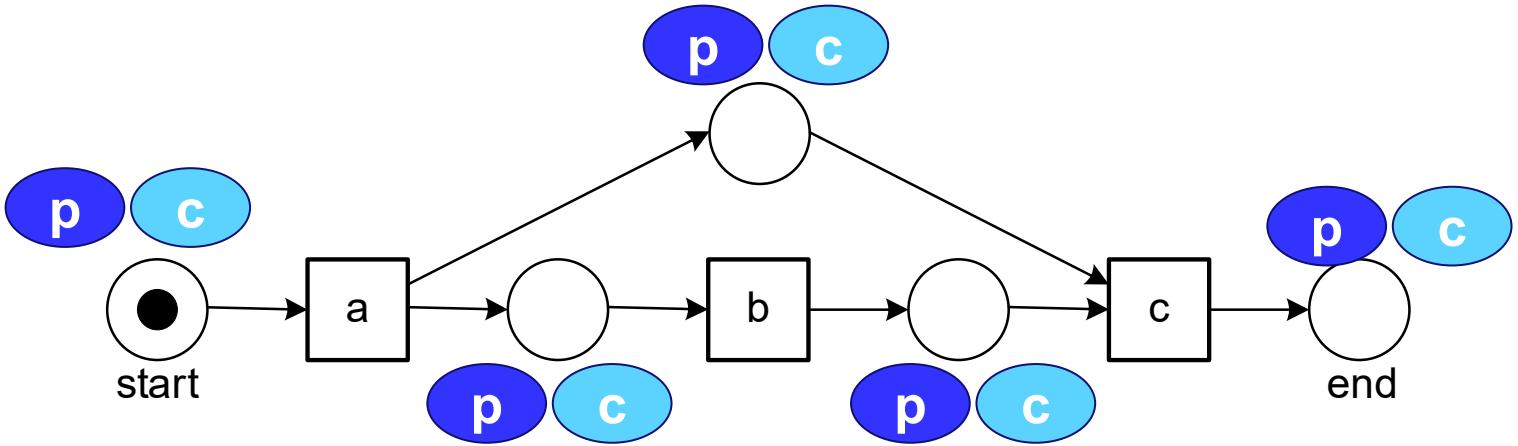
Model generated by the Alpha Algorithm

$$L_{11} = [\langle a, b, c \rangle^{20}, \langle a, c \rangle^{30}]$$



Trace abc

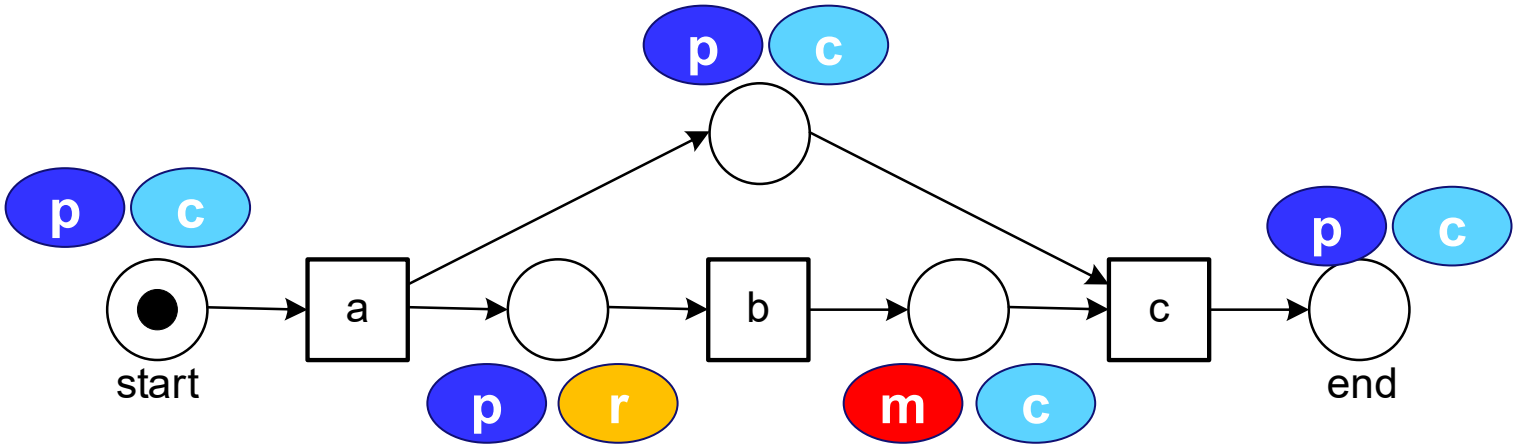
$$L_{11} = [\langle a, b, c \rangle^{20}, \langle a, c \rangle^{30}]$$



p = 5, c = 5, r=0, m=0

Trace ac

$$L_{11} = [\langle a, b, c \rangle^{20}, \langle a, c \rangle^{30}]$$



p = 4, c = 4, r=1, m=1

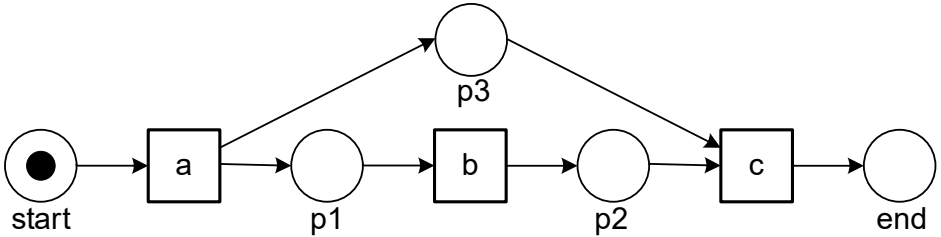
Overall fitness

$$fitness(L,N) = \frac{1}{2}\left(1 - \frac{\sum_{\sigma \in L} L(\sigma) \times m_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times c_{N,\sigma}}\right) + \frac{1}{2}\left(1 - \frac{\sum_{\sigma \in L} L(\sigma) \times r_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times p_{N,\sigma}}\right)$$

$$L_{11} = [\langle a,b,c\rangle^{20}, \langle a,c\rangle^{30}]$$

p = 5, c = 5, r=0, m=0

p = 4, c = 4, r=1, m=1



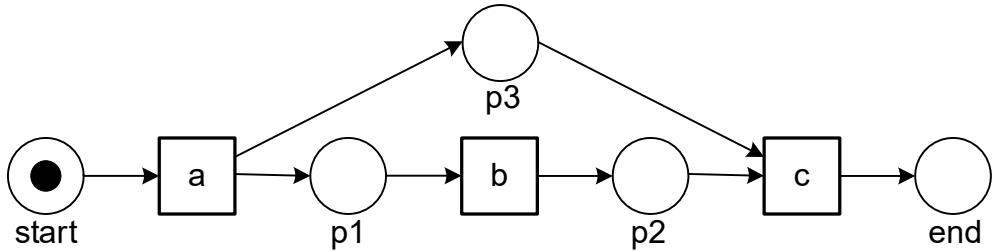
p = 20*5+30*4 =220, c = 20*5+30*4=220,
r = 20*0+30*1=30, m=20*0+30*1=30

$$\frac{1}{2}\left(1-\frac{30}{220}\right)+\frac{1}{2}\left(1-\frac{30}{220}\right)=\frac{19}{22}\approx 0.86$$

Chair of Process and Data Science

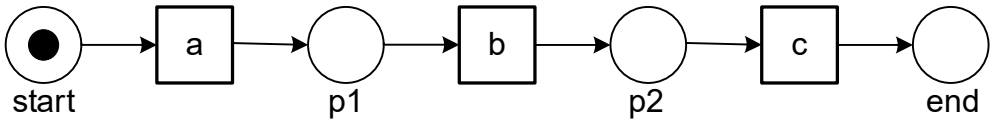
Redundant places impact fitness (1/2)

$$L_{11} = [\langle a, b, c \rangle^{20}, \langle a, c \rangle^{30}]$$



$$\frac{1}{2}\left(1 - \frac{30}{220}\right) + \frac{1}{2}\left(1 - \frac{30}{220}\right) = \frac{19}{22} \approx 0.86$$

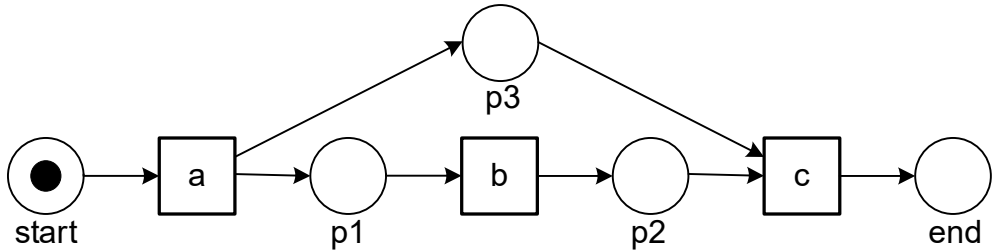
$$L_{11} = [\langle a, b, c \rangle^{20}, \langle a, c \rangle^{30}]$$



$$\frac{1}{2}\left(1 - \frac{30}{170}\right) + \frac{1}{2}\left(1 - \frac{30}{170}\right) = \frac{14}{17} \approx 0.82$$

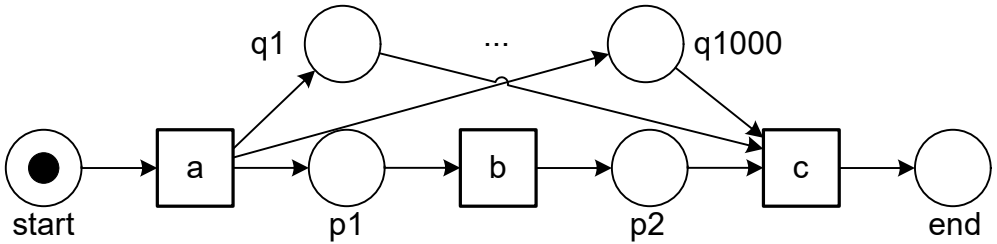
Redundant places impact fitness (2/2)

$$L_{11} = [\langle a,b,c\rangle^{20}, \langle a,c\rangle^{30}]$$



$$\frac{1}{2}\left(1-\frac{30}{220}\right)+\frac{1}{2}\left(1-\frac{30}{220}\right)=\frac{19}{22}\approx 0.86$$

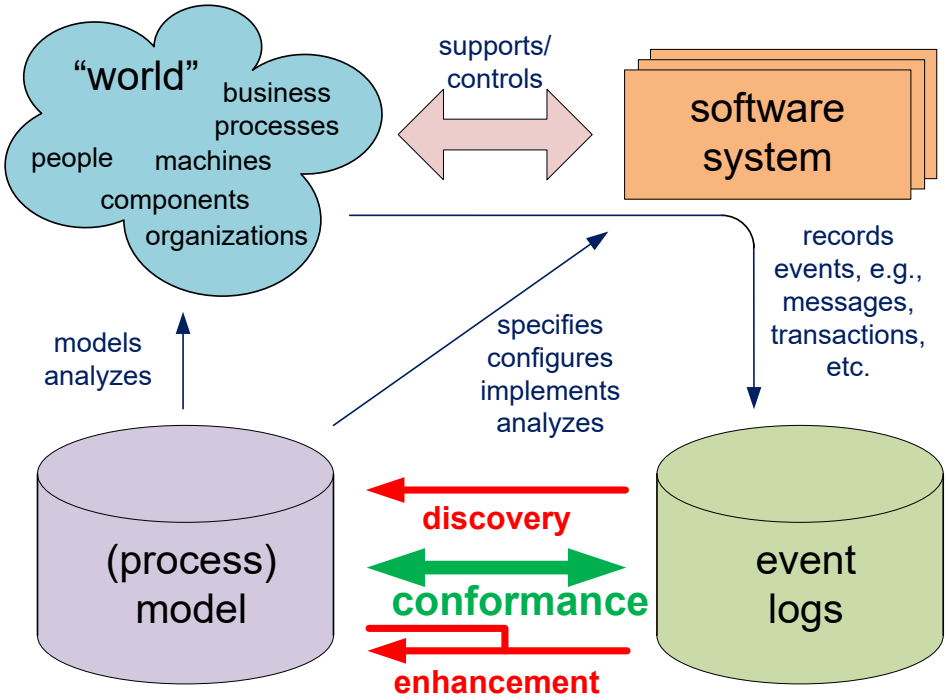
$$L_{11} = [\langle a,b,c\rangle^{20}, \langle a,c\rangle^{30}]$$



$$\frac{1}{2}\left(1-\frac{30}{50170}\right)+\frac{1}{2}\left(1-\frac{30}{50170}\right)=\frac{5014}{5017}\approx 0.999$$

Aligning Observed and Modeled Behavior

Conformance checking



1. Conformance checking using causal footprints.
2. Conformance checking based on token-based replay.
3. **Alignment-based conformance checking.**

Chair of Process and Data Science

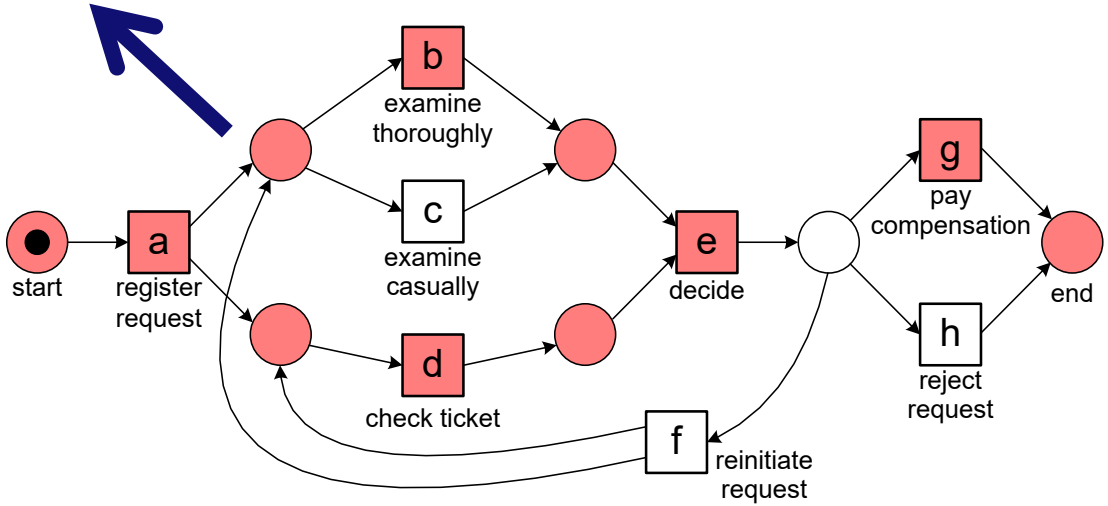
Requirements

- Conformance checking should not impose restrictions on the process notation (e.g., silent transitions and two transitions with the same label should be possible).
- Two semantically equivalent models should have the same conformance value.
- Should provide a "closest matching path" through the process model for any trace in the event log.
  - Also required for performance analysis!
  - Beyond the analysis of replay fitness (advanced diagnostics, precision, generalization, etc.).

Alignments

a	b	d	e	g
a	b	d	e	g

⟨a,b,d,e,g⟩

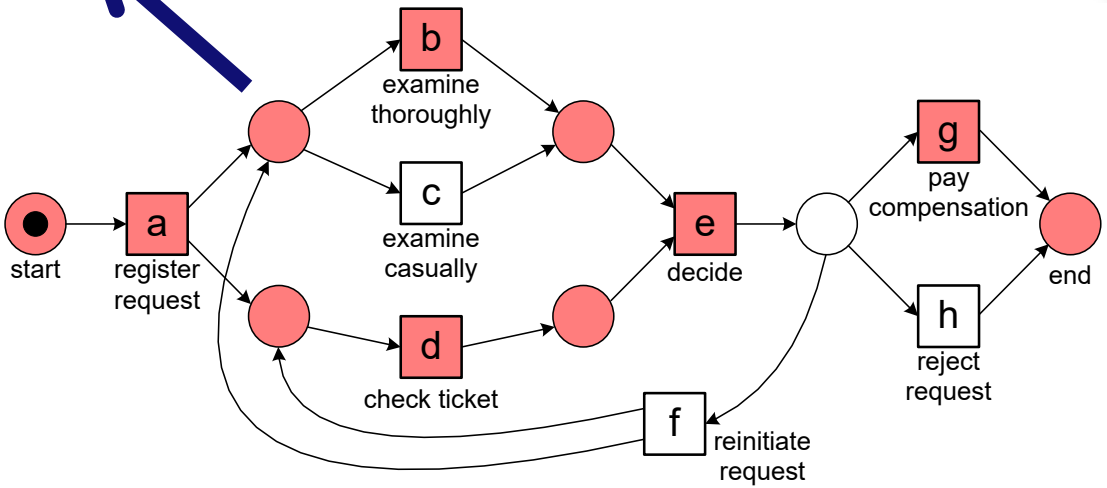


#	trace
455	acdeh
191	abdeg
177	adceh
144	abdeh
111	acdeg
82	adceg
56	adbeh
47	acdefdbeh
38	adbeg
33	acdefbdeh
14	acdefbdeg
11	acdefdbeg
9	adcefcdeh
8	adcefdbeh
5	adcefbdeg
3	acdefbdefdbeg
2	adcefdbeg
2	adcefbdefbdeg
1	adcefdbefbdeh
1	adbefbdefdbeg
1	adcefdbefcdefdbeg
1391	

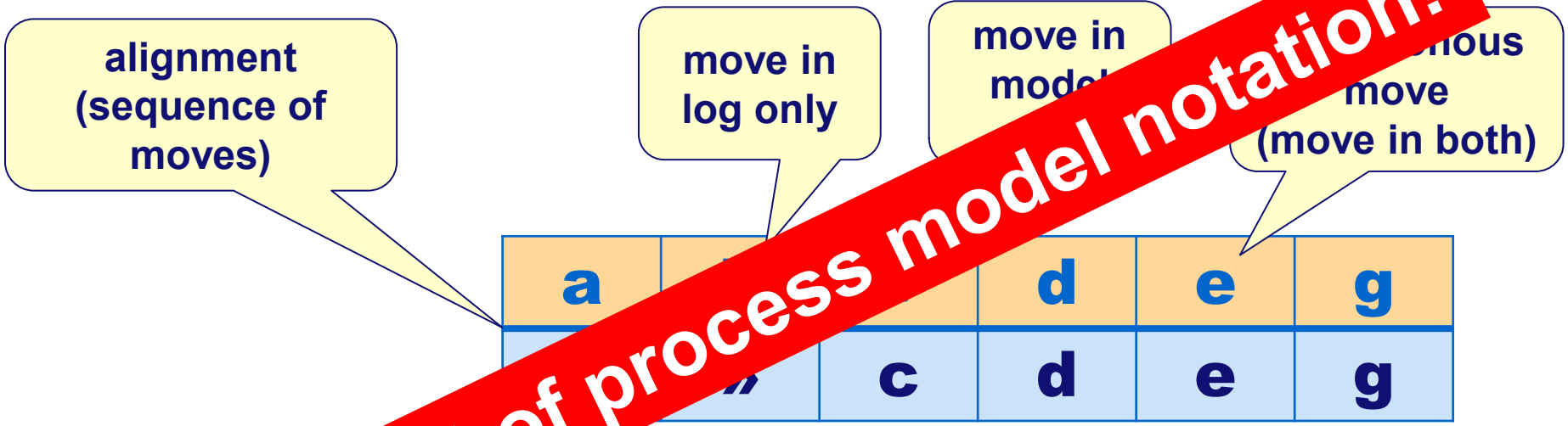
Alignments

a	»	d	e	g	h
a	b	d	e	g	»

⟨a,d,e,g,h⟩



Terminology



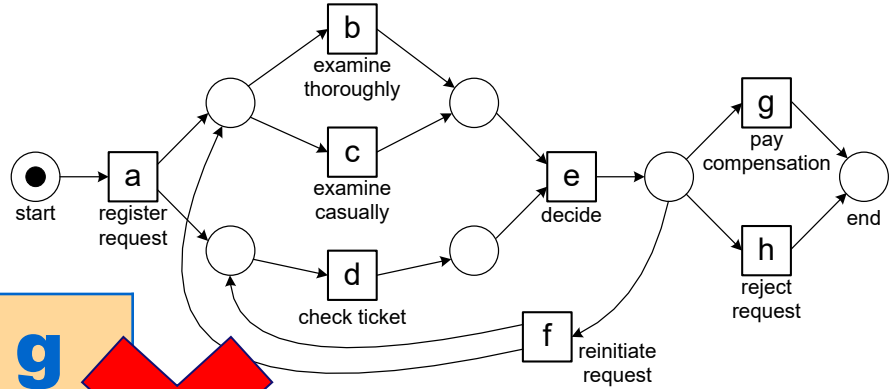
Independent of process model notation!

- Projecti… row (remove "no moves") corresponds to t… in the event log.
  …tion on bottom row (remove "no moves") …rresponds to a run of the model.

Chair of Process and Data Science

Optimal alignment for ⟨a,b,d,e,g⟩

a	b	d	e	g
a	b	d	e	g

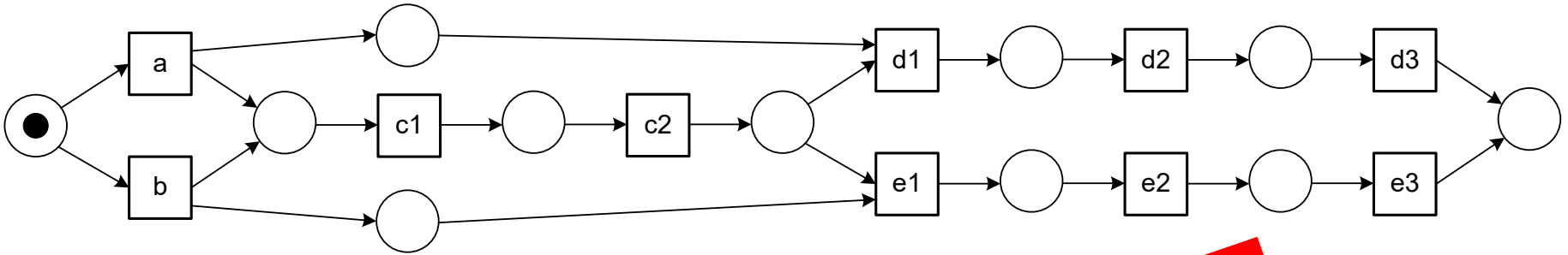


a	b	»	d	e	g
a	»	c	d	e	g

a	b	d	e	g	»	»	»	»	»
»	»	»	»	»	a	c	d	e	g

Chair of Process and Data Science

Optimal alignment for ⟨a,c1,c2,e1,e2,e3⟩ ?



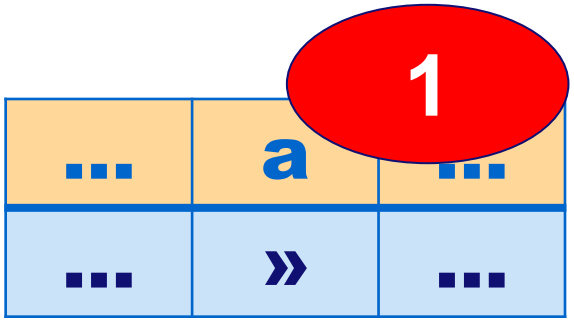
a	»	c1	c2	e1	e2	e3
»	b	c1	c2	e1		

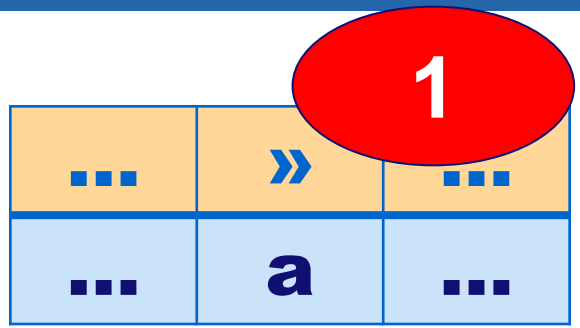
a	c1			»	e1	e2	e3
a	c	d1	d2	d3	»	»	»

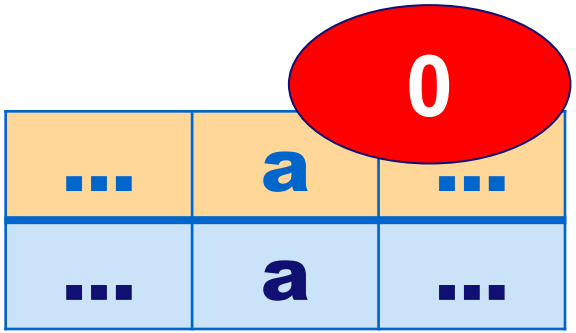
Depends on cost function!

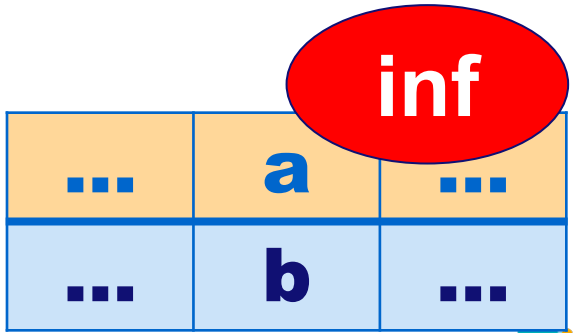
Chair of Process and Data Science

Standard cost function
count »'s in alignment







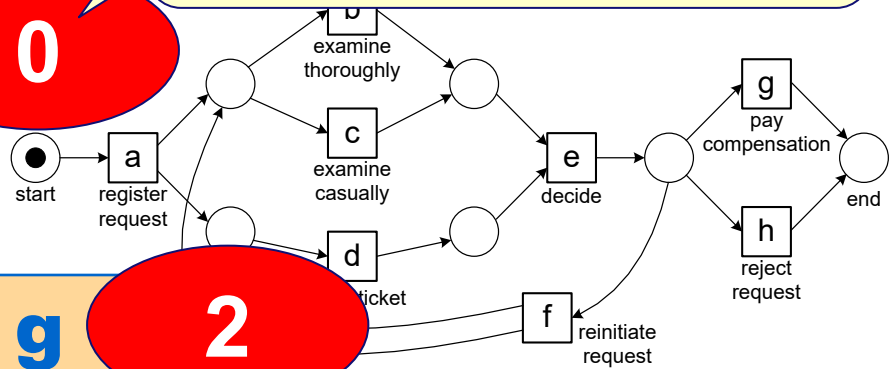


Using the standard cost f

optimal
there is no other alignment that has lower costs

a	b	d	e	g
a	b	d	e	g

0



a	b	»	d	e	g
a	»	c	d	e	g

2

a	b	d	e	g	»	»	»	»	»
»	»	»	»	»	a	c	d	e	g

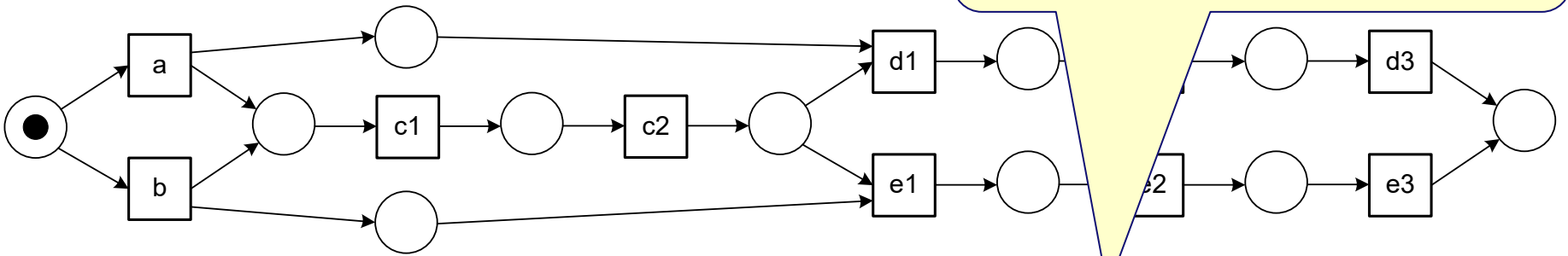
10

Chair of Process and Data Science

Using the standard cost fu

trace in log: ⟨a,c1,c2,e1,e2,e3⟩

optimal
there is no other alignment that has lower costs



a	»	c1	c2	e1	e2	e3
»	b	c1	c2	e1	e2	e3

2

a	c1	c2	»	»	»	e1	e2	e3
a	c1	c2	d1	d2	d3	»	»	»

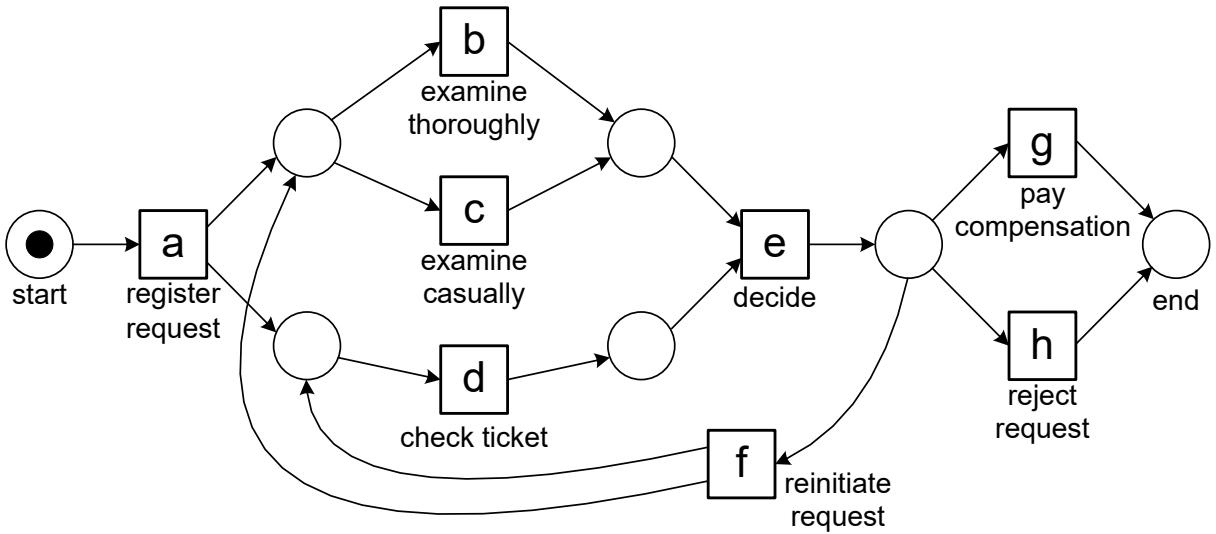
6

Chair of Process and Data Science

Optimal alignment for ⟨a,b,e,f,d,e,g⟩ (1/2)

a	b	e	f	d	e	g
a	b	»	»	d	e	g

2

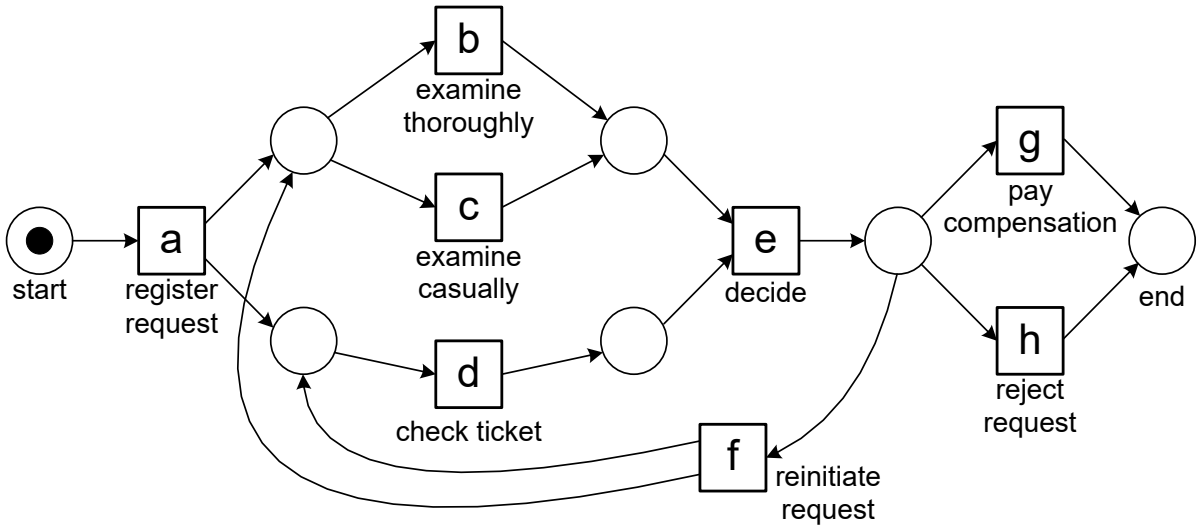


loop is not taken: e and f in event log are discarded

Optimal alignment for ⟨a,b,e,f,d,e,g⟩ (2/2)

a	b	»	e	f	d	»	e	g
a	b	d	e	f	d	b	e	g

2



loop is taken: d and b are missing in event log

Chair of Process and Data Science

Not one unique optimal alignment

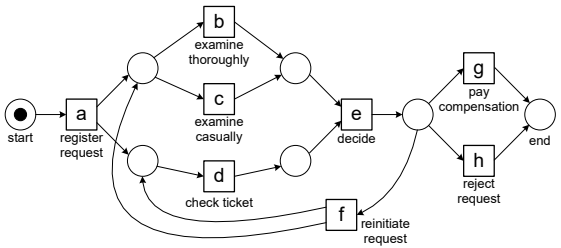
a	b	»	e	f	d	»	e	g
a	b	d	e	f	d	b	e	g

2

a	b	e	f	d	e	g
a	b	»	»	d	e	g

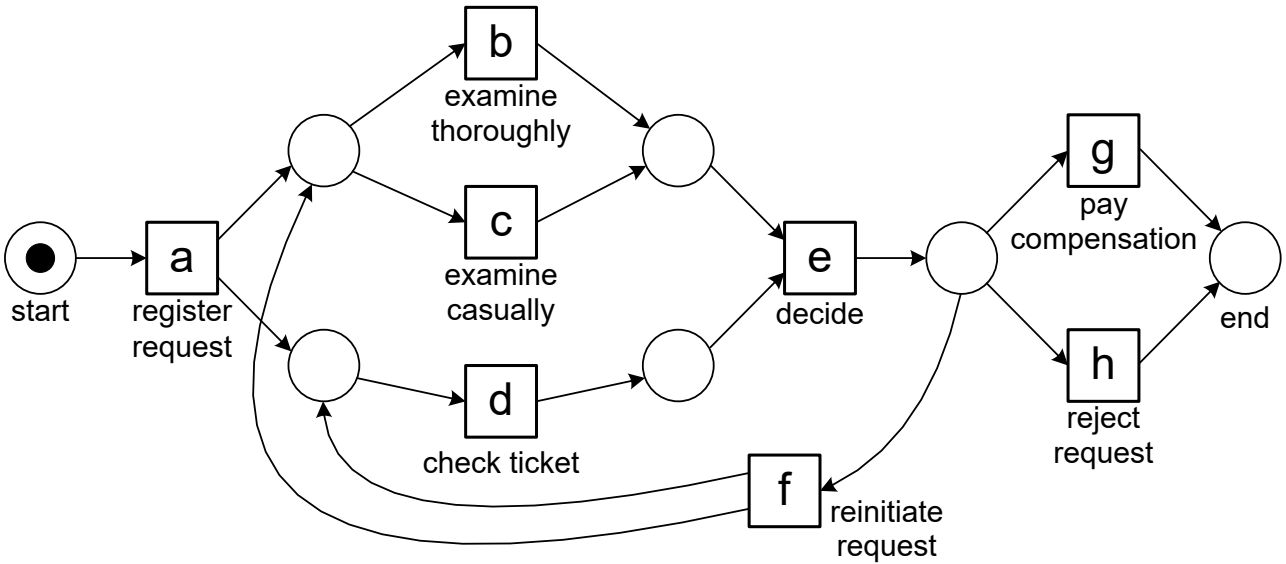
2

...



Chair of Process and Data Science

Question: How many optimal alignments are there for ⟨a,b,e,f,d,e,g⟩ ?



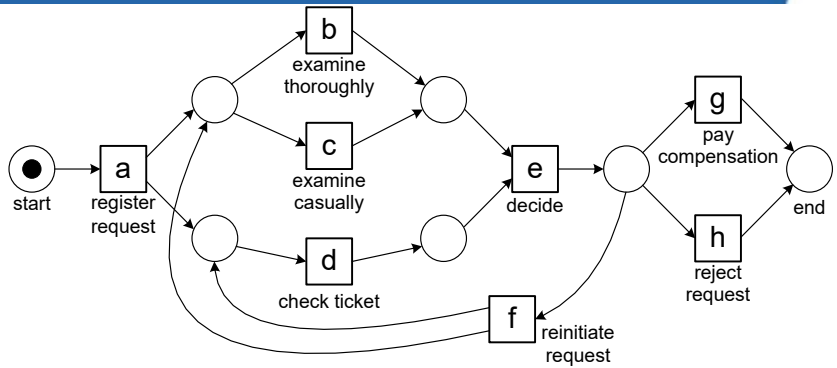
⟨a,b,e,f,d,e,g⟩

Answer: 9

1+(2x2)+(2x2)=9 optimal alignments having cost 2

a	b	e	f	d	e	g
a	b	»	»	d	e	g

1x



move in model can be reordered in concurrent part

4x

a	b	»	e	f	d	»	e	g
a	b	d	e	f	d	b	e	g

4x

a	b	»	e	f	d	»	e	g
a	b	d	e	f	d	c	e	g

Any cost structure is possible

...	send-letter(John,4 weeks, $400)	...
...	send-email(Sue,3 weeks,$500)	...

Any cost structure is possible

...	send-letter(John,4 weeks, $400)	...
...	send-email(Sue,3 weeks,$500)	...

similar activities (lower costs for related activities)

Any cost structure is possible

...	send-letter(John,4 weeks, $400)	...
...	send-email(Sue,3 weeks,$50	...

resource-related conformance costs
(done by someone that does or does not have the specified role)

Any cost structure is possible

...	send-letter(John,4 weeks, $400)	...
...	send-email(Sue,3 weeks,$500)	...

time-related conformance costs (activity should happen within a preset deadline)

Any cost structure is possible

...	send-letter(John,4 weeks, $400)	...
...	send-email(Sue,3 weeks,$500)	...

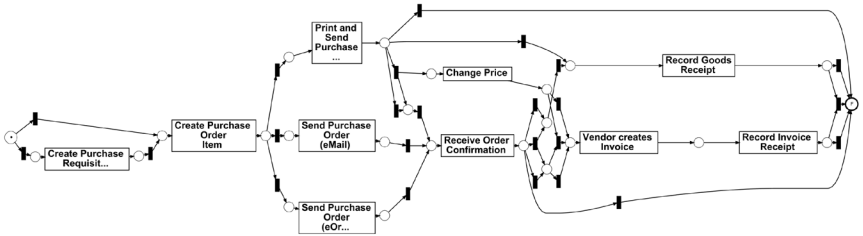
data-related conformance costs (routing condition is violated, e.g., path only for more valuable orders)

Any cost structure is possible

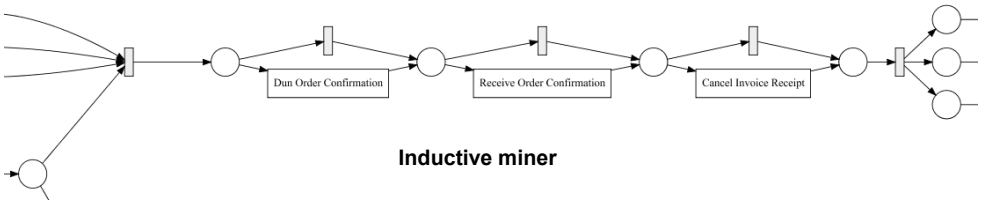
...	send-letter(John,4 weeks, $400)	...
...	send-email(Sue,3 weeks,$500)	...

risk-related conformance costs, context-dependent conformance costs, …

Side note: Silent transitions



Heuristic miner



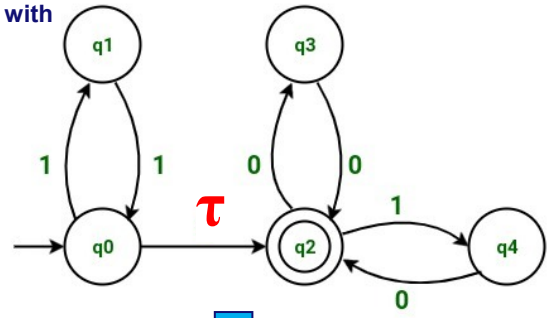
Inductive miner

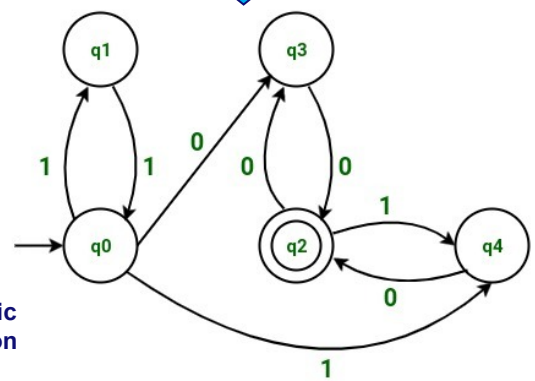
- **Two options:**
  - **Remove these transitions when considering the model's behavior** (Model reduction: any Nondeterministic Finite Automaton (NFA) with silent activities, can be translated into an NFA without silent activities, which, in turn, can be translated into a Deterministic Finite Automaton (DFA).)
  - **The corresponding moves on model have cost 0** (or a very small epsilon in case of loops)

Chair of Process and Data Science

Example Transformations (outside scope course)

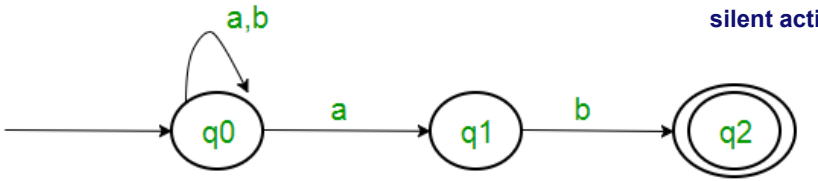
Nondeterministic Finite Automaton (NFA) with silent activities

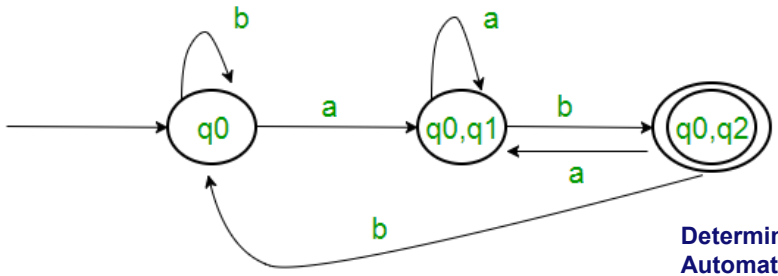




Nondeterministic Finite Automaton NFA without silent activities

Nondeterministic Finite Automaton NFA without silent activities





Deterministic Finite Automaton (DFA)

Examples taken from https://www.geeksforgeeks.org/

Chair of Process and Data Science

Computing fitness

C[illegible]n optimal alignment with a

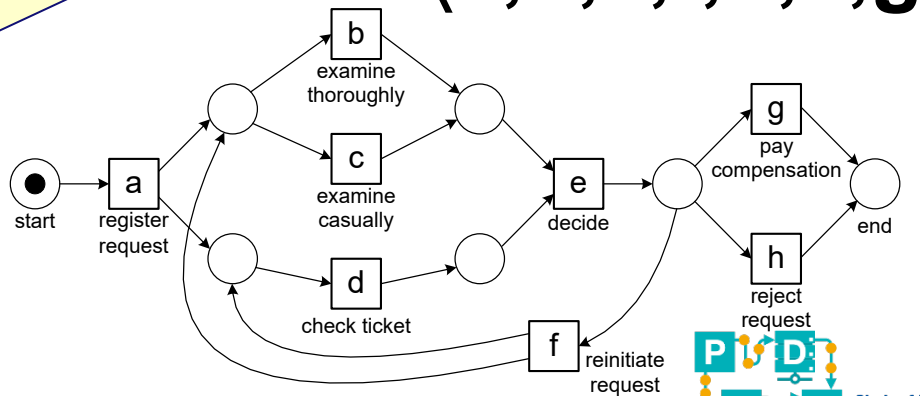
mo[illegible]model.

all events cause a move in log only

shortest path from initial state to final state

$$1 - \frac{2}{7+5} = 0.833$$

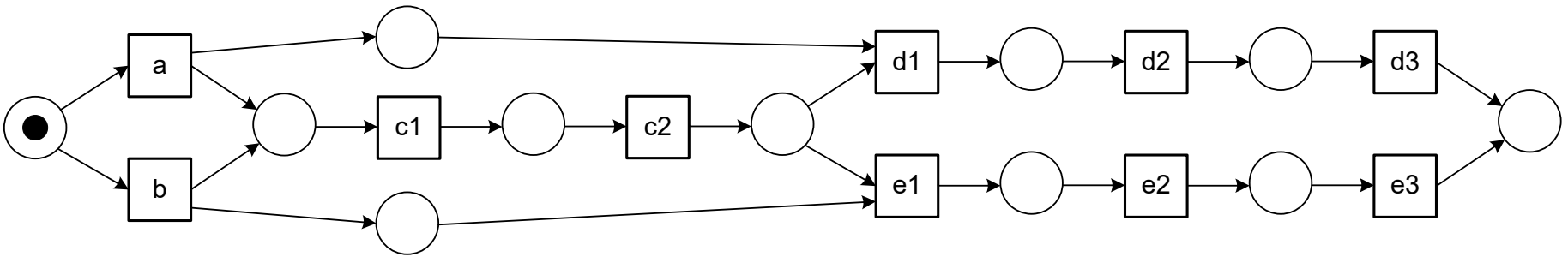
⟨a,b,e,f,d,e,g⟩



Chair of Process and Data Science

Question: Compute alignment-based fitness

⟨a,c1,c2,e1,e2,e3⟩



Answer

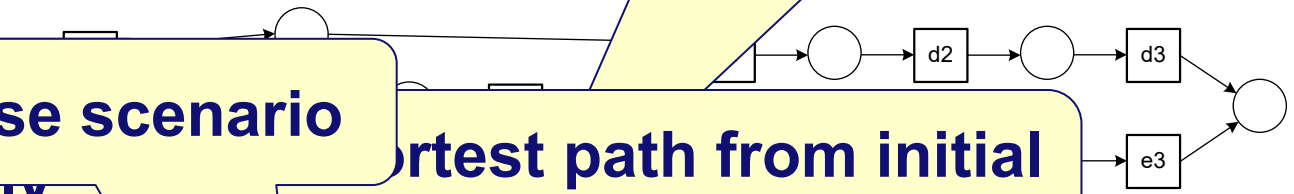
cost of optimal alignment = 2

⟨a,c1,c2,e1,e2,e3⟩

all ev… move in log only

worst-case scenario

…ortest path from initial state to final state



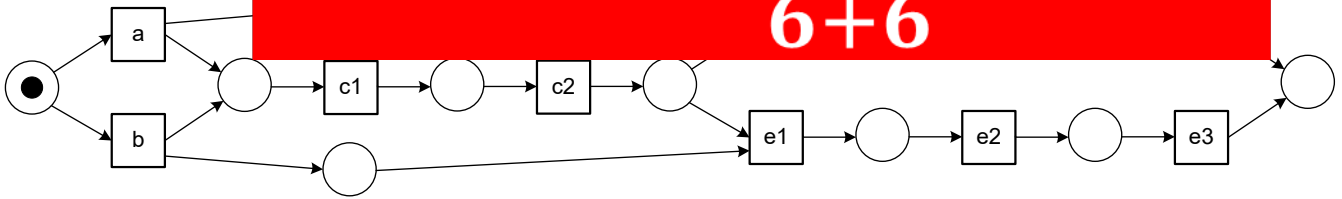
a		c1	c2		e…	
»		c1	c2	e…	e2	e…

a	c1	c2	e1	e2	e3	»	»	»	»	»	»
»	»	»	»	»	»	a	c1	c2	d1	d2	d3

Answer

$$\text{fitness} = 1 - \frac{2}{6+6} = 0.833$$

⟨a,c1,c2,e1,e2,e3⟩



2

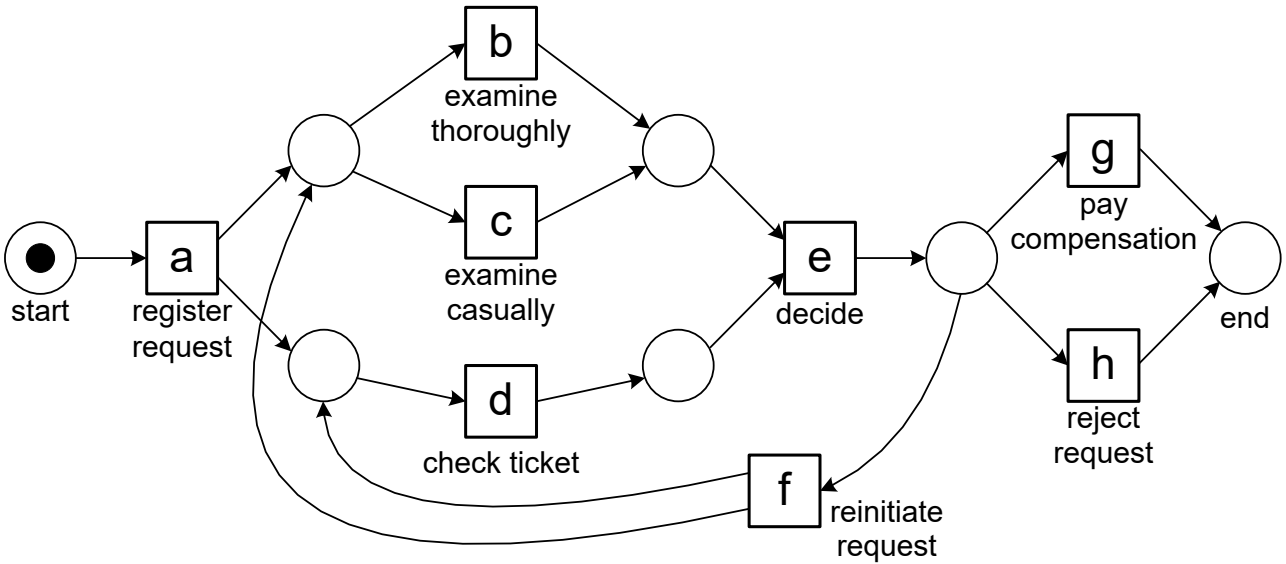
a	»	c1	c2	e1	e2	e3
»	b	c1	c2	e1	e2	e3

6

a	c1	c2	e1	e2	e3	»	»	»	»	»	»
»	»	»	»	»	»	a	c1	c2	d1	d2	d3

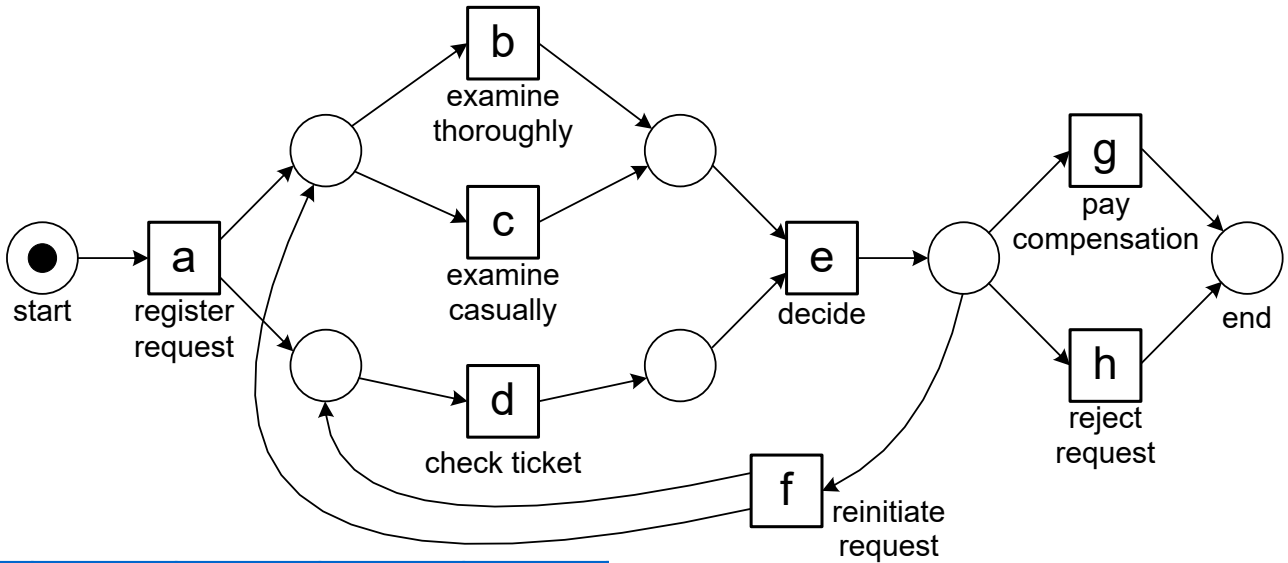
6

Another example: What is the alignment-based fitness of the trace ⟨b,c⟩ ?



⟨b,c⟩

What is the alignment-based fitness of the trace ⟨b,c⟩ ?

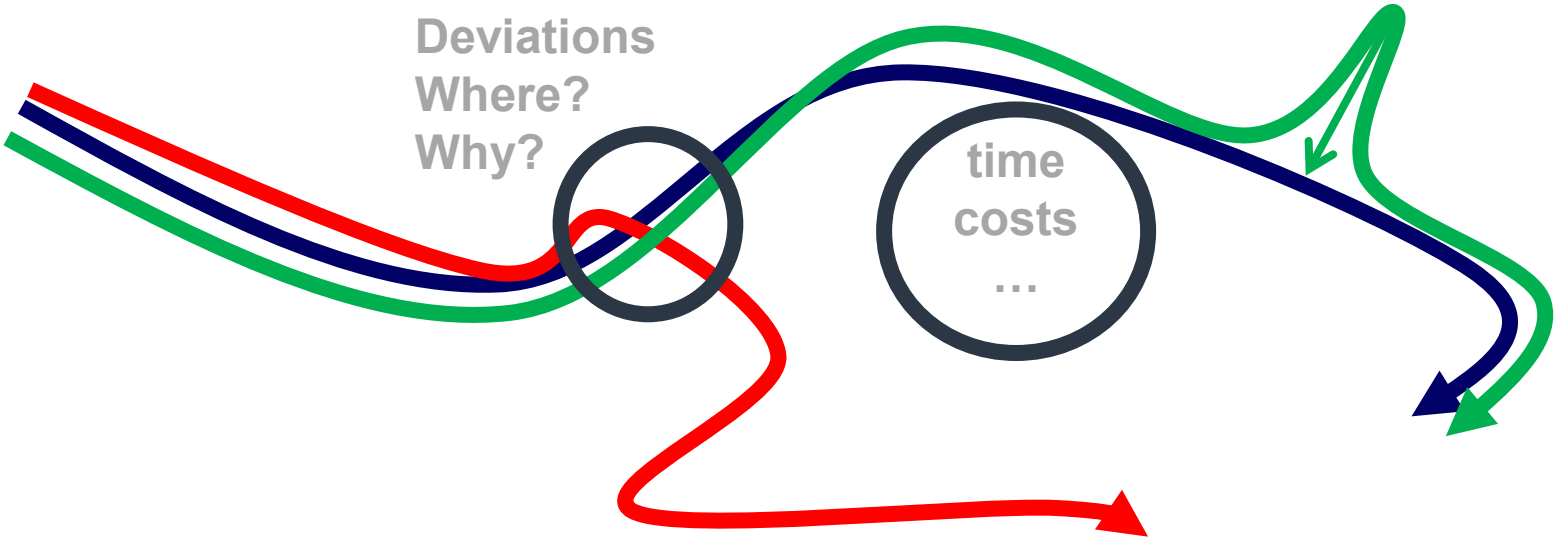


»	b	c	»	»	»
a	b	»	d	e	h

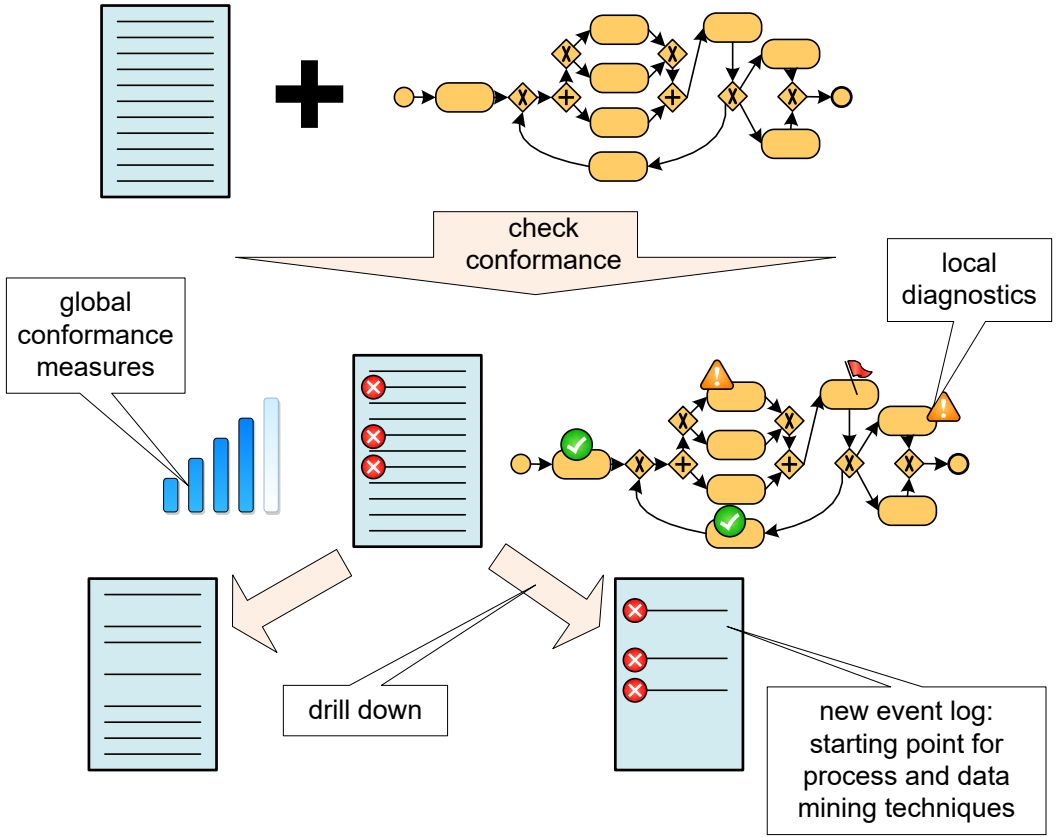
$$1 - \frac{5}{2+5} = \frac{2}{7} = 0.286$$

Advantages of aligning log and model

- Observed behavior is directly related to modeled behavior.
- Very flexible (any cost structure).
- Detailed diagnostics.
- After aligning log and model, other quality dimensions can be investigated (separation of concerns).



Drilling down



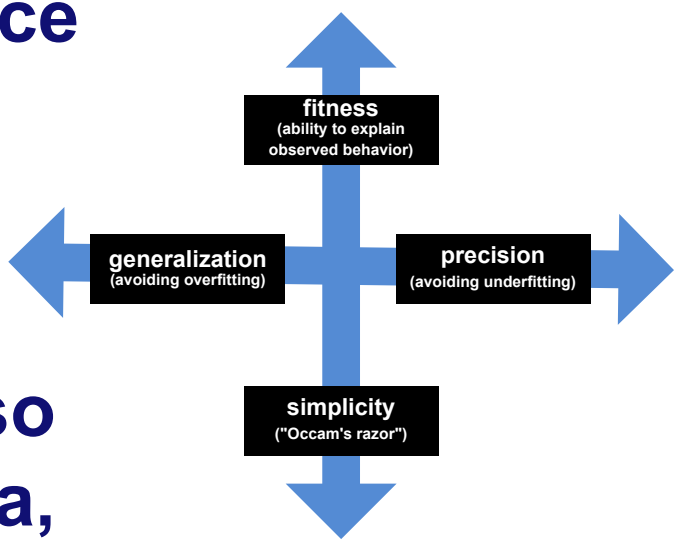
Example approach

- Create event log containing deviating (or non-deviating) cases.
- Apply process discovery to new log.
- Compare process models.

Later more on comparative process mining.

Beyond fitness and control-flow

- There are also solid conformance measures for precision, generalization, and simplicity.
- Multiple definitions possible.
- Conformance checking may also include other perspectives (data, resources, time, cost, etc.).
- Example: data-aware alignments.



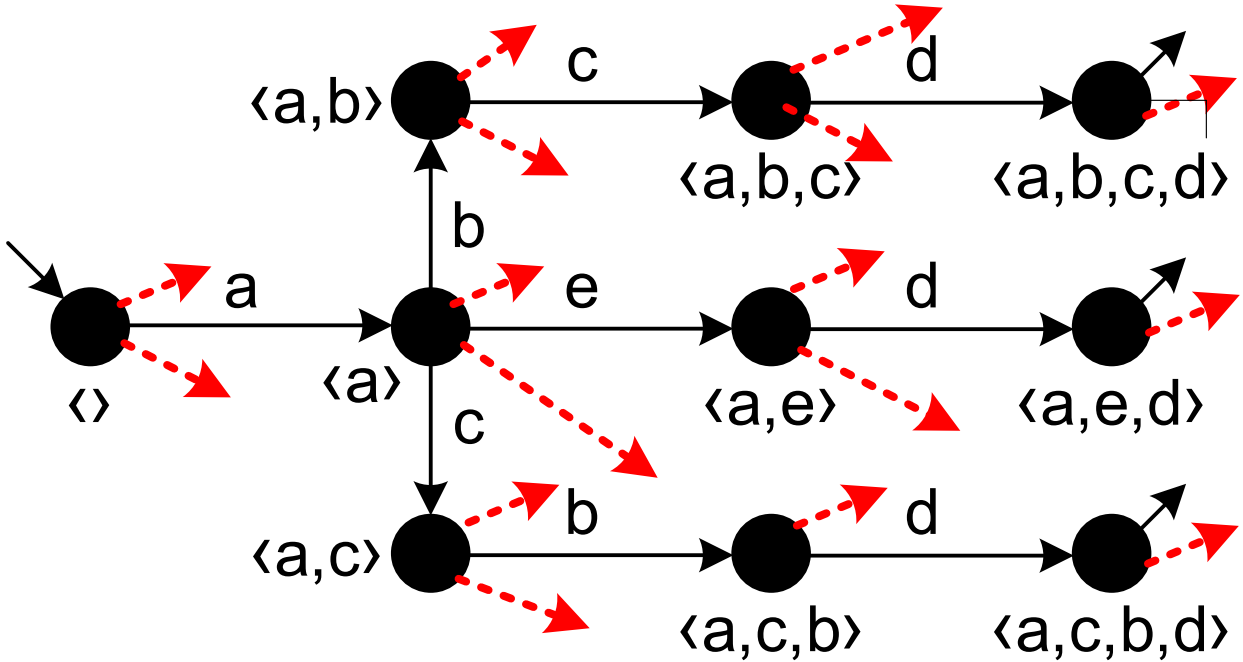
Example: Precision (1/2)

- Naïve approach: Compute the fraction of modeled behavior actually observed in the event log.
- Problems:
  - If “behavior” = “trace”, then the model with loops has a precision of 0 (because there are ∞-many traces).
  - Sensitive to the size of the event log. Precision gets better when a longer period is taken.
  - Recall that an event log contains example behavior and often many unique traces.

Example: Precision (2/2)

- Many smarter approaches, e.g., using escaping edges.
- Assume the event log has been aligned, i.e., the remaining events are synchronous moves or model moves.
- Build a prefix automaton (see lecture on region-based mining) based on the aligned event log.
- Extend the prefix automaton with escaping edges, i.e., situation where the model allows for more behavior.
- Quantify such escaping edges.

Prefix Automaton with escaping edges



$$L_1 = [\langle a,b,c,d\rangle^3, \langle a,c,b,d\rangle^2, \langle a,e,d\rangle]$$

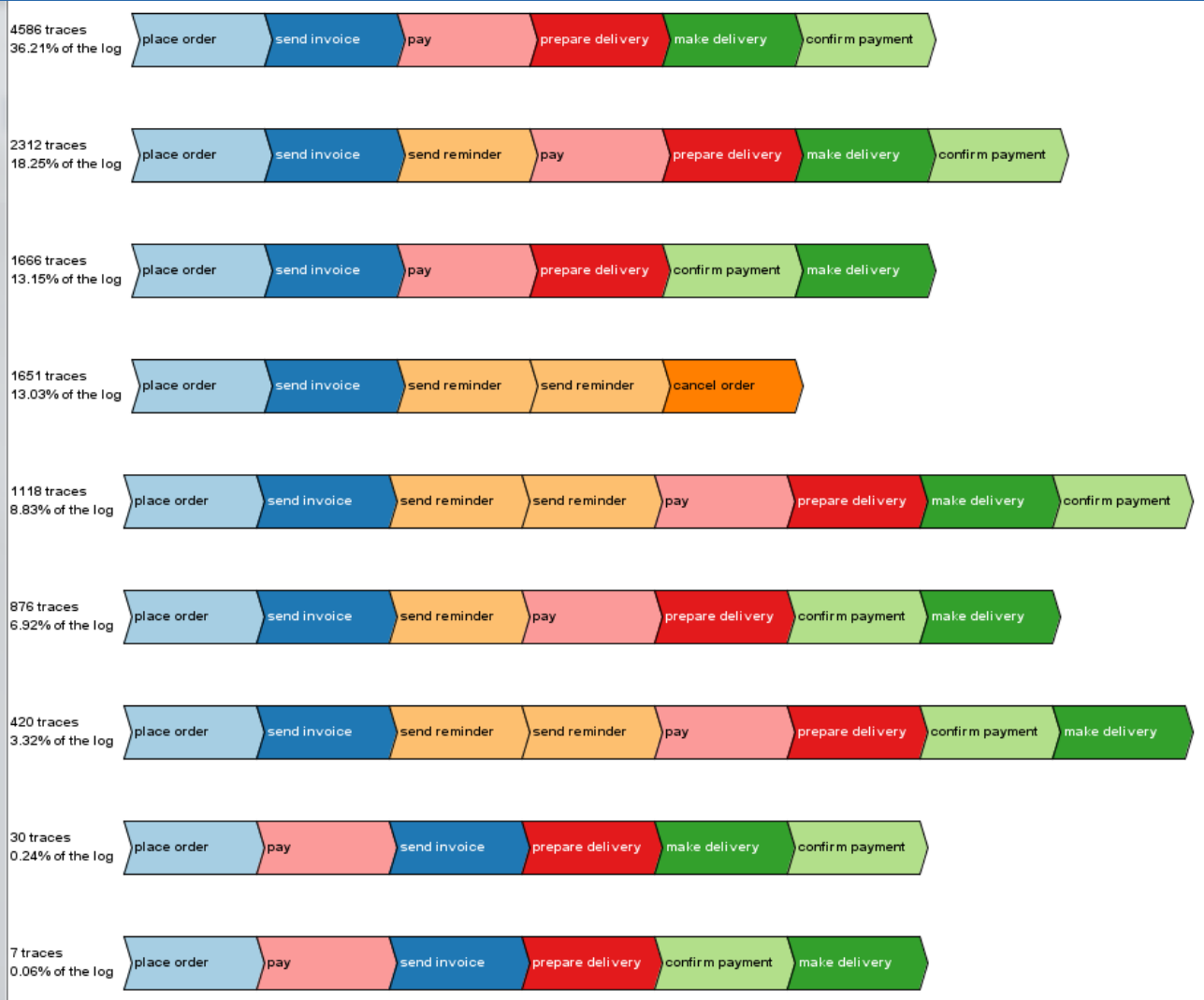
W.M.P. van der Aalst, A. Adriansyah, and B. van Dongen. Replaying History on Process Models for Conformance Checking and Performance Analysis. WIREs Data Mining and Knowledge Discovery, 2(2):182-192, 2012.
A. Adriansyah, J. Munoz-Gama, J. Carmona, B.F. van Dongen, and W.M.P. van der Aalst. Measuring Precision of Modeled Behavior. Information Systems and e-Business Management, 13(1):37-67, 2015.

Red arcs indicate that while replaying the model could do more than what was observed in the event log. It is possible to qualify precision by taking into account how often a node is visited and what the fraction of escaping edges is.

Tooling

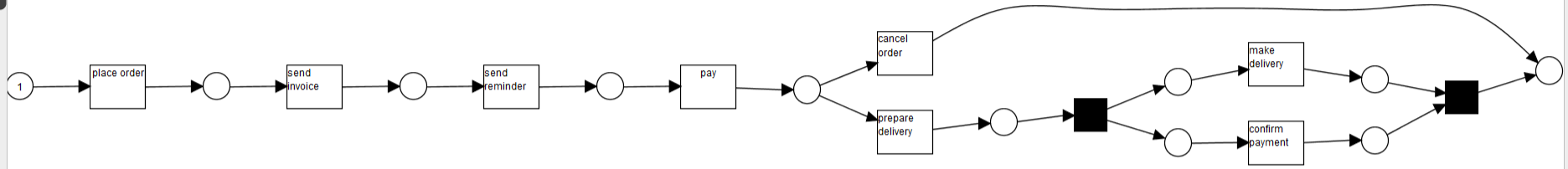
ProM: Load Event Log

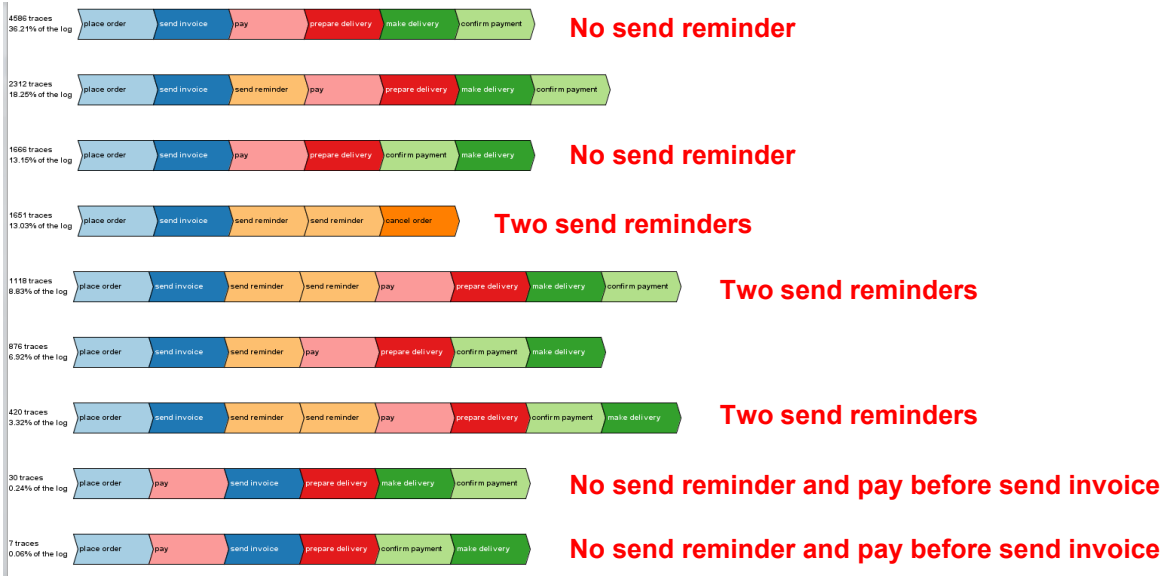
Event log



- 12,666 cases
- 80,609 events
- 8 unique activities

Chair of Process and Data Science


1
place order
send invoice
send reminder
pay
cancel order
prepare delivery
make delivery
confirm payment


4586 traces 36.21% of the log
place order
send invoice
pay
prepare delivery
make delivery
confirm payment
No send reminder
2312 traces 18.25% of the log
place order
send invoice
send reminder
pay
prepare delivery
make delivery
confirm payment
1666 traces 13.15% of the log
place order
send invoice
pay
prepare delivery
confirm payment
make delivery
No send reminder
1651 traces 13.03% of the log
place order
send invoice
send reminder
send reminder
cancel order
Two send reminders
1118 traces 8.83% of the log
place order
send invoice
send reminder
send reminder
pay
prepare delivery
make delivery
confirm payment
Two send reminders
876 traces 6.92% of the log
place order
send invoice
send reminder
pay
prepare delivery
confirm payment
make delivery
420 traces 3.32% of the log
place order
send invoice
send reminder
send reminder
pay
prepare delivery
confirm payment
make delivery
Two send reminders
30 traces 0.24% of the log
place order
pay
send invoice
prepare delivery
make delivery
confirm payment
No send reminder and pay before send invoice
7 traces 0.06% of the log
place order
pay
send invoice
prepare delivery
confirm payment
make delivery
No send reminder and pay before send invoice

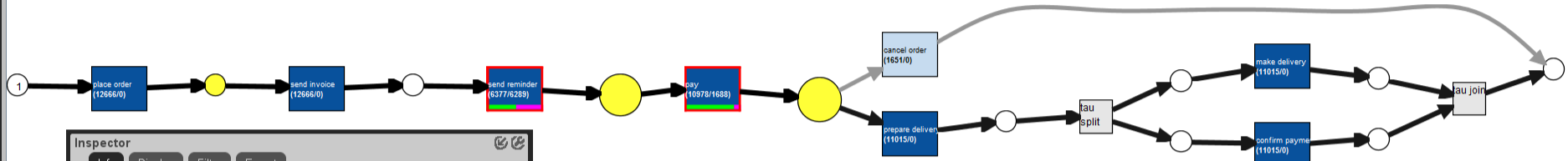
ProM UITopia

ProM 6

designed by fluxicon

Replay result - log XES Event Log on 17b4f6d0-518b-460c-91bc-a26759af0d7f using A* Cost-based Fitness Express with ILP (swap+replacement aware), a

Select visualisation ...



Inspector

Info | Display | Filter | Export

Legend

View

Elements Statistics

Selected elements* All elements, aggregated

Property	Value
#Move log+model	77383
#Move model only	30007
#Move log only	3226
#Traces	12666

* Click a place/transition on the projected model to see its stats, or use combobox

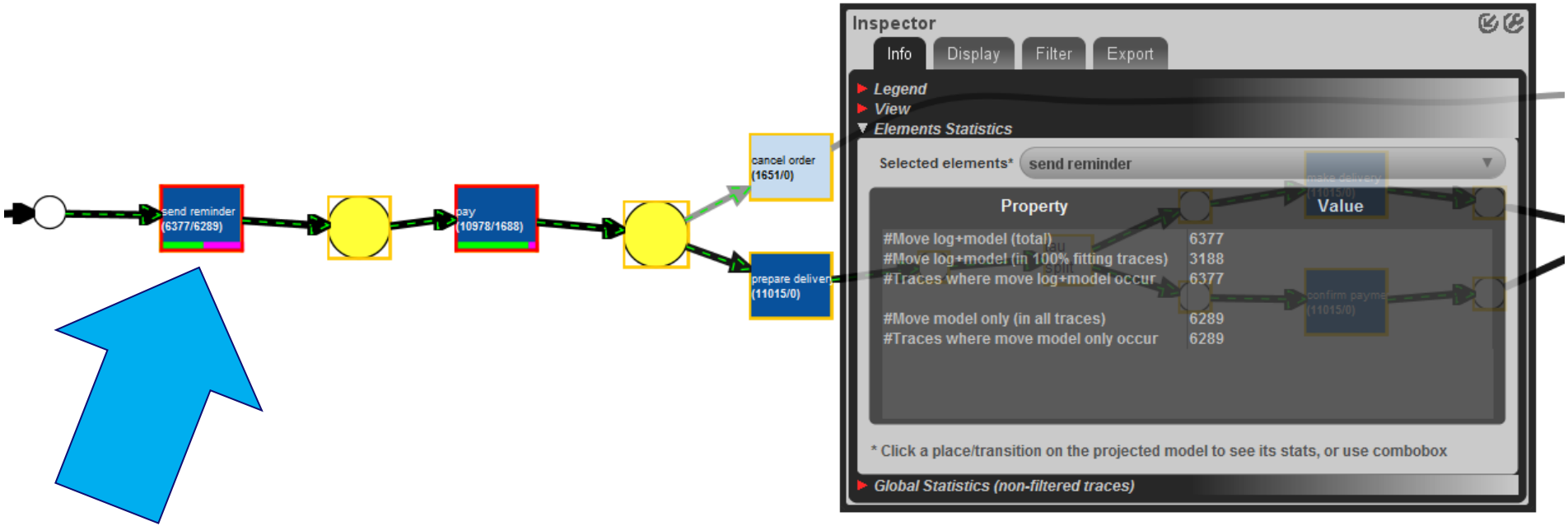
Global Statistics (non-filtered traces)

Property	Value
Raw Fitness Cost	0.8844939207326743
Queued States	25.729827885678265
Num. States	10.130033159639995
Calculation Time (ms)	1.4122848570977433
Move-Log Fitness	0.9582649086794042
Trace Fitness	0.9189197664801649
Trace Length	6.364203379125239
Move-Model Fitness	0.9025805869481839

3226 move on log only (yellow)
30007 move on model only (purple)

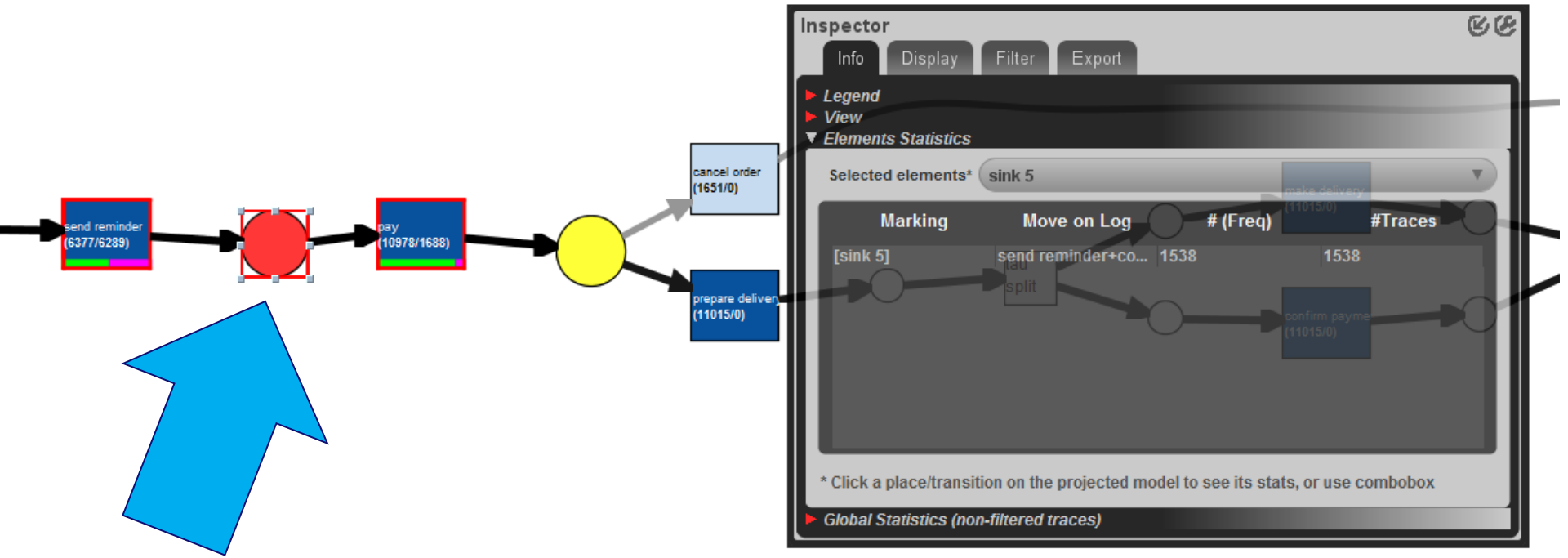
30007=6289+1688+11015+11015 move on model only

Send reminder is often skipped (6289 times)

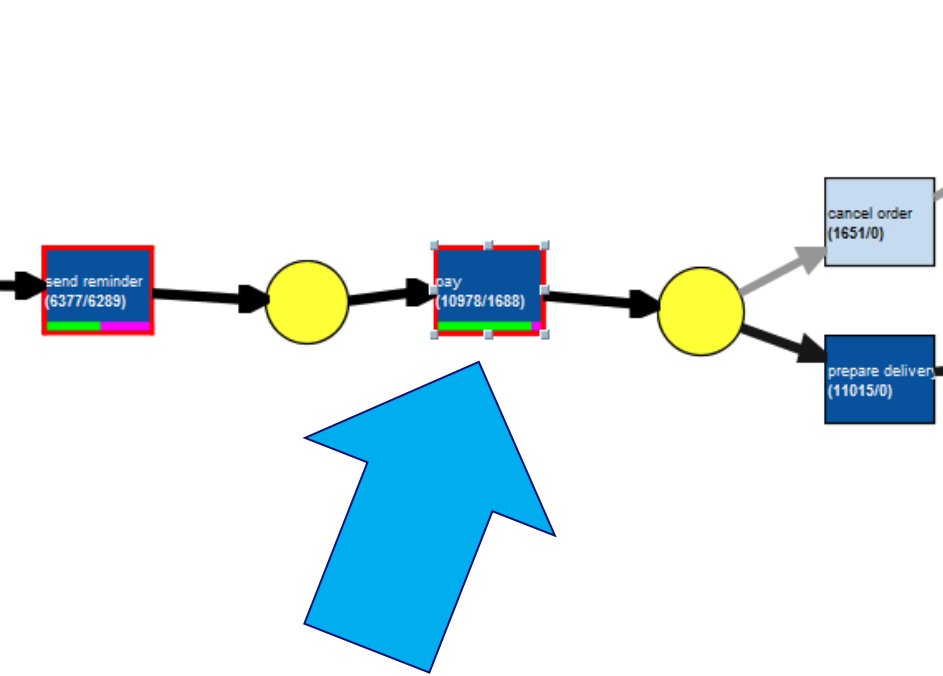


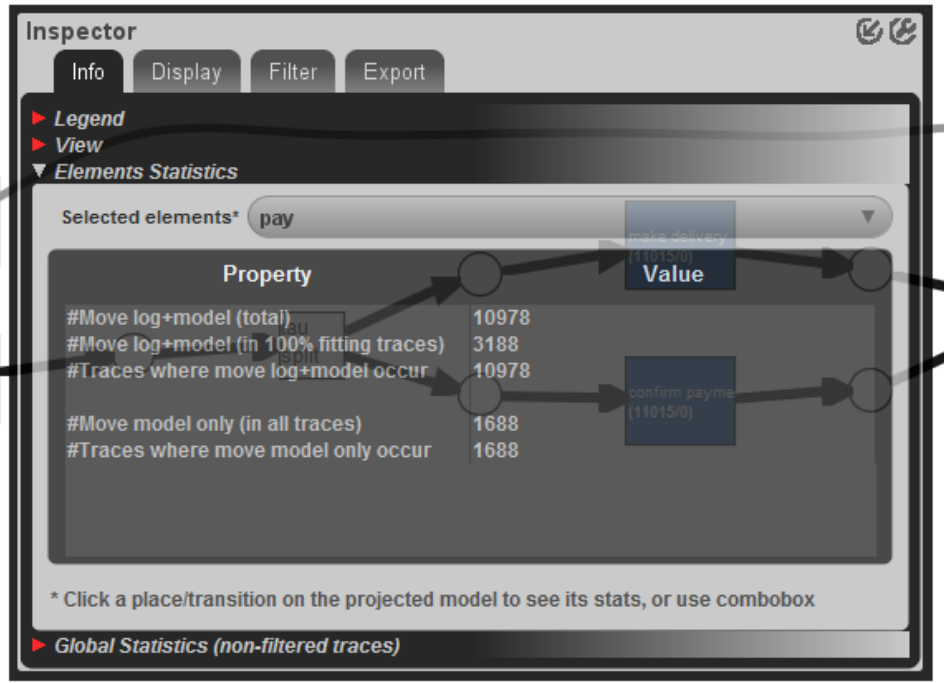
Chair of Process and Data Science

Additional send reminder (1538 times in this place)



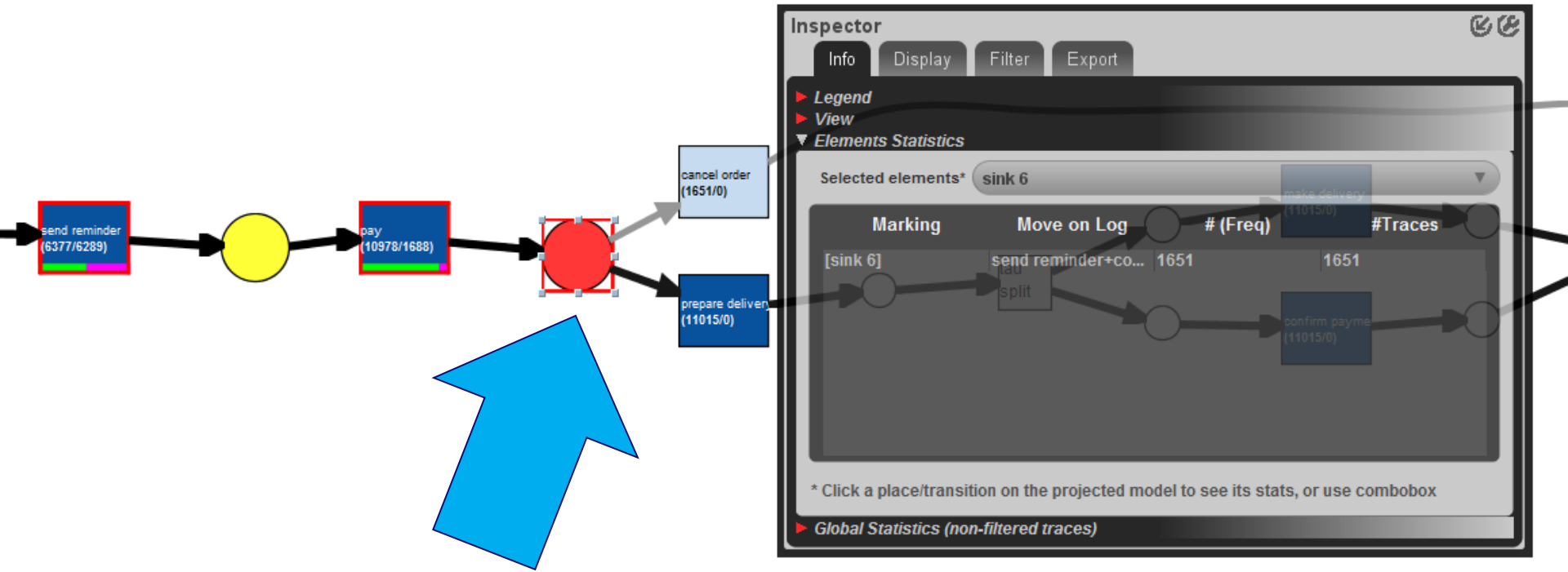
Pay is sometimes skipped (1688 times)





Chair of Process and Data Science

Additional send reminder (1651 times in this place)



Chair of Process and Data Science

Log view

synchronous move

move on model
(required activity was skipped in event log)

move on model
(silent transition in model was fired)

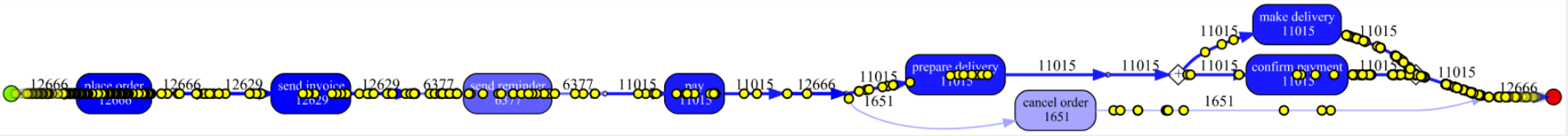
move on log
(activity in log was not possible in model)

Chair of Process and Data Science

Many more plug-ins exploring conformance (based on alignments or not)

Chair of Process and Data Science

Inductive miner



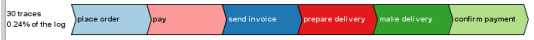
No send reminder

No send reminder

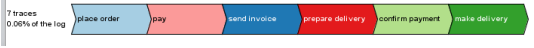
Two send reminders

Two send reminders

Two send reminders



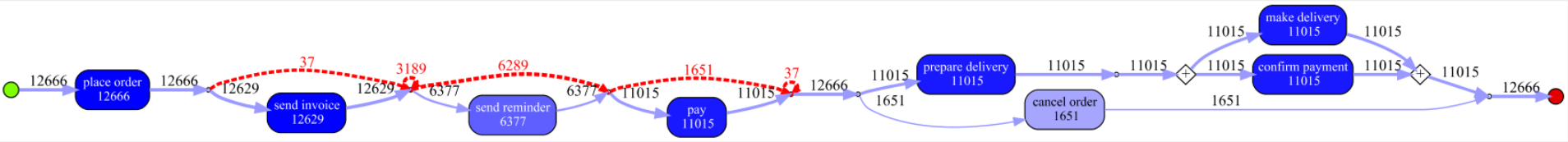
No send reminder and pay before send invoice

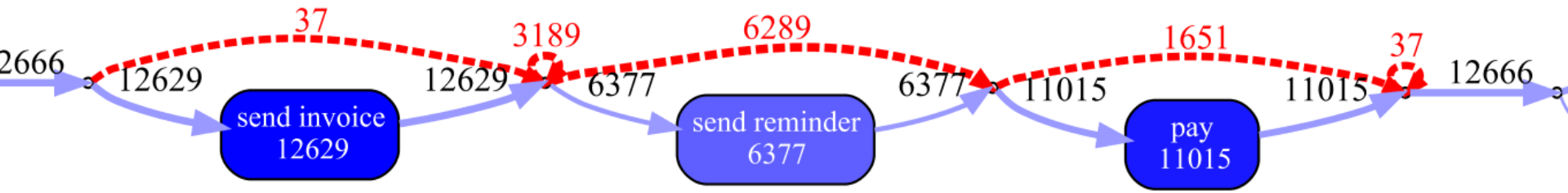


No send reminder and pay before send invoice

Chair of Process and Data Science

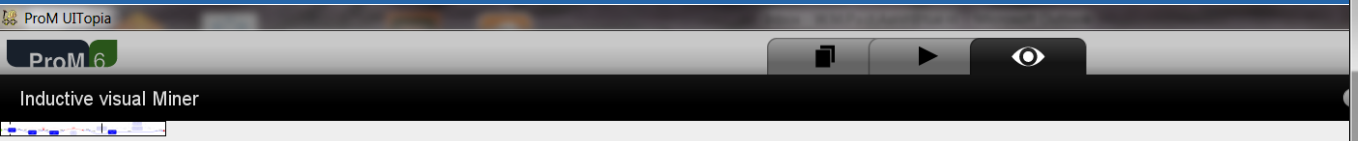
Deviations in inductive miner





Note that some numbers are different than before because of different alignments.

Drill down on deviations

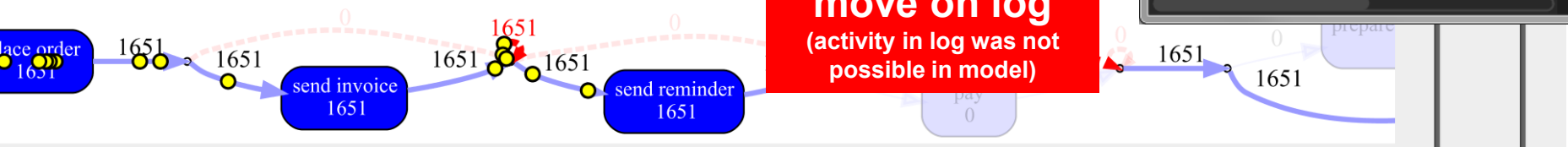


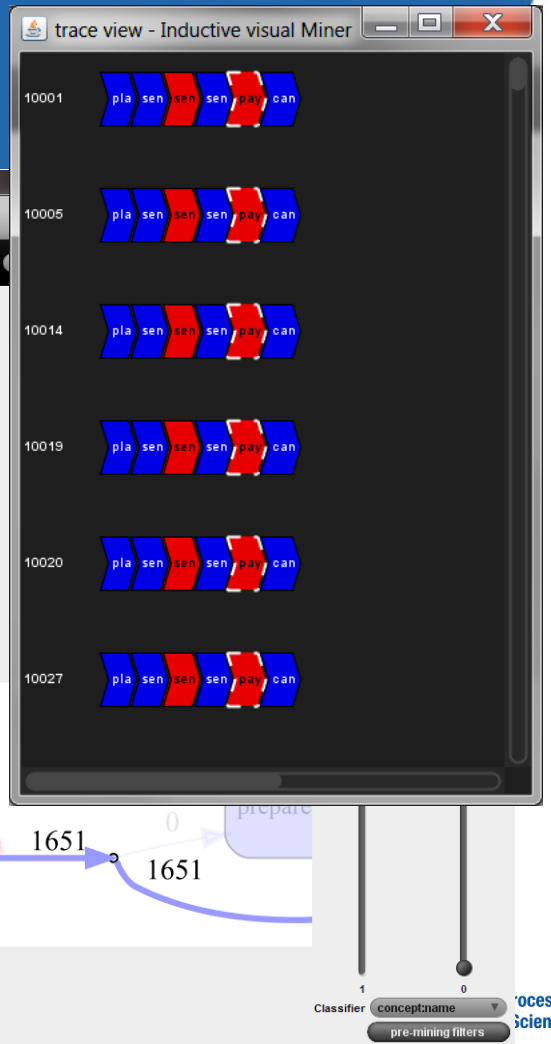
Pay is skipped 1651 times and all of these cases had an extra send reminder (these are the cases that were cancelled).

synchronous move

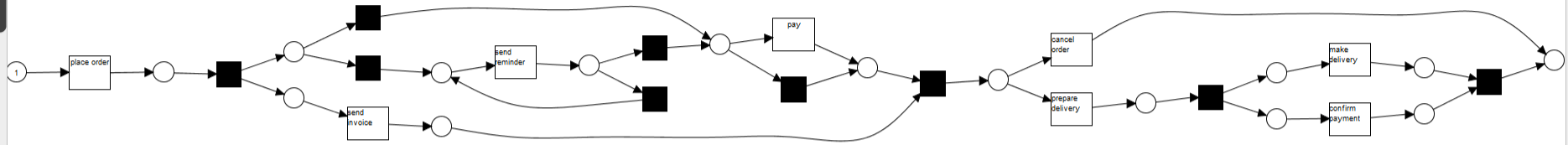
move on model
(required activity was skipped in event log)

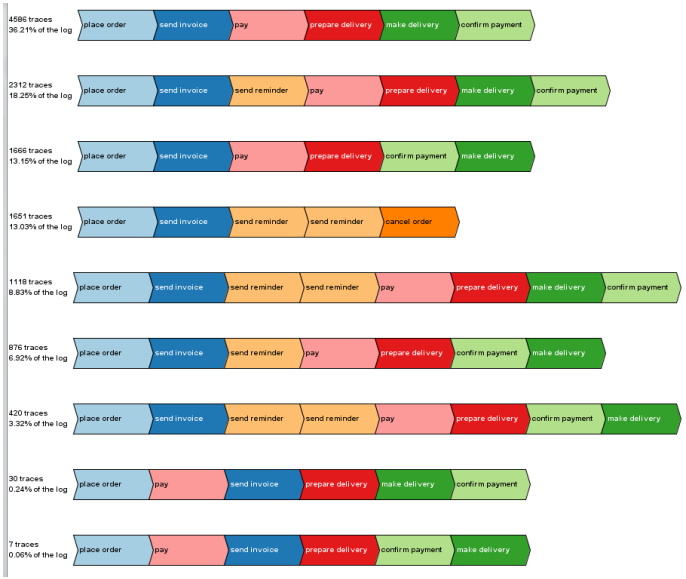
move on log
(activity in log was not possible in model)



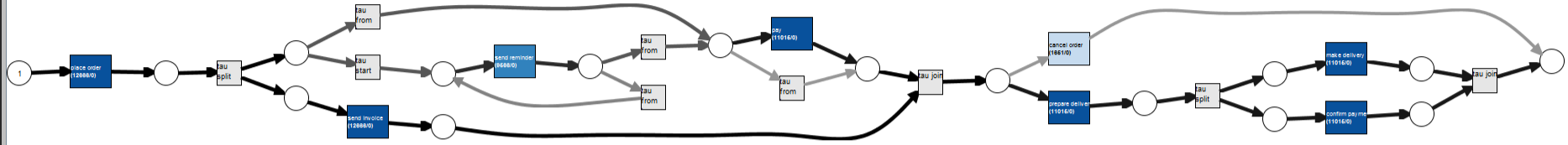


Same event log and different (underfitting) model





Perfect replay fitness (can reproduce all traces), but underfitting (i.e., not precise enough).



Inspector

Info | Display | Filter | Export

- Legend
- View
- Elements Statistics

Selected elements* All elements, aggregated

Property	Value
#Move log+model	80609
#Move model only	71245
#Move log only	0
#Traces	12666

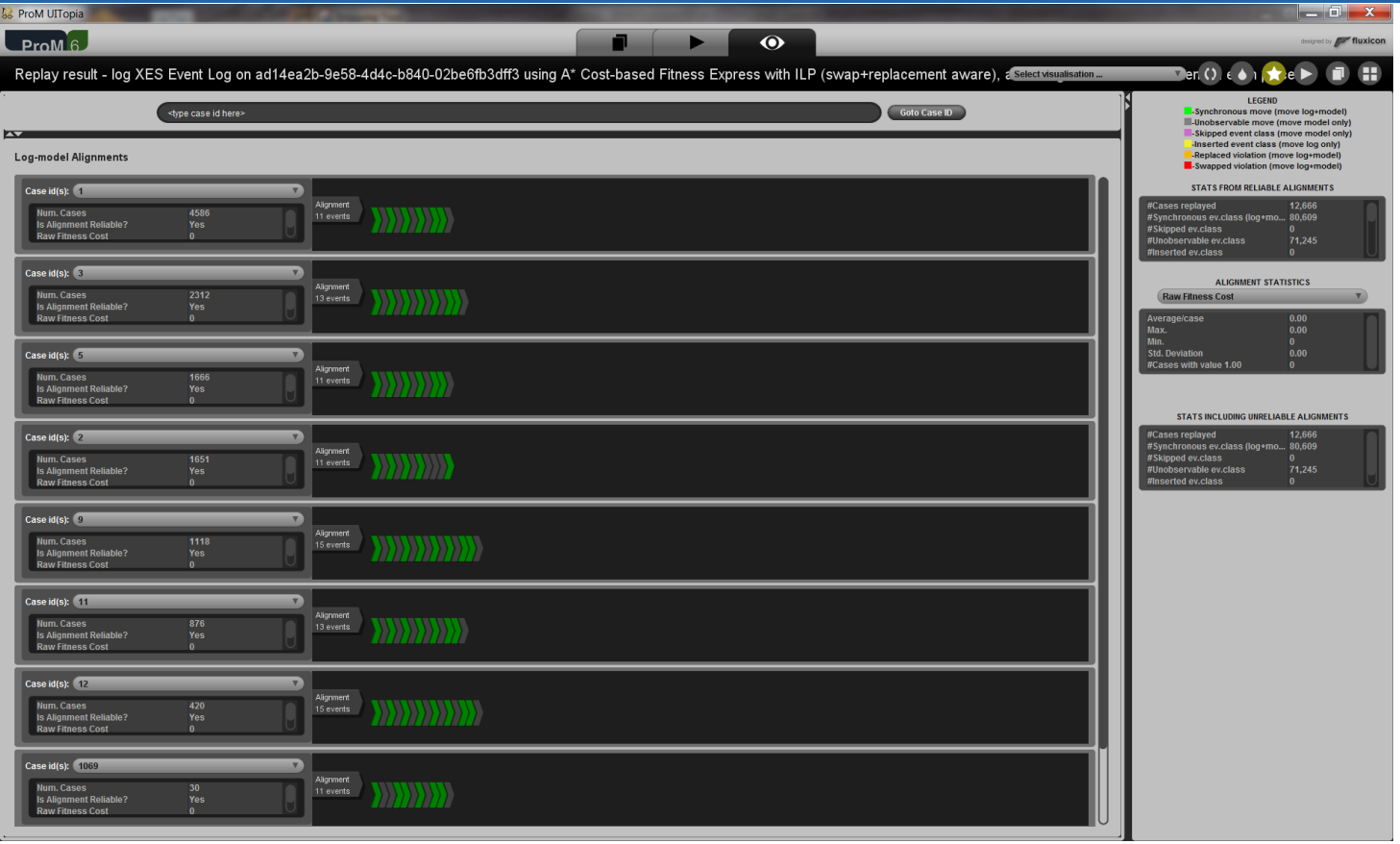
* Click a place/transition on the projected model to see its stats, or use combobox

- Global Statistics (non-filtered traces)

Property	Value
Raw Fitness Cost	0.0
Queued States	45.91386388757287
Num. States	15.896652455392376
Calculation Time (ms)	8.713721774830239
Move-Log Fitness	1.0
Trace Fitness	1.0
Trace Length	6.364203379125239
Move-Model Fitness	1.0

No problems, only moves on model for silent steps.

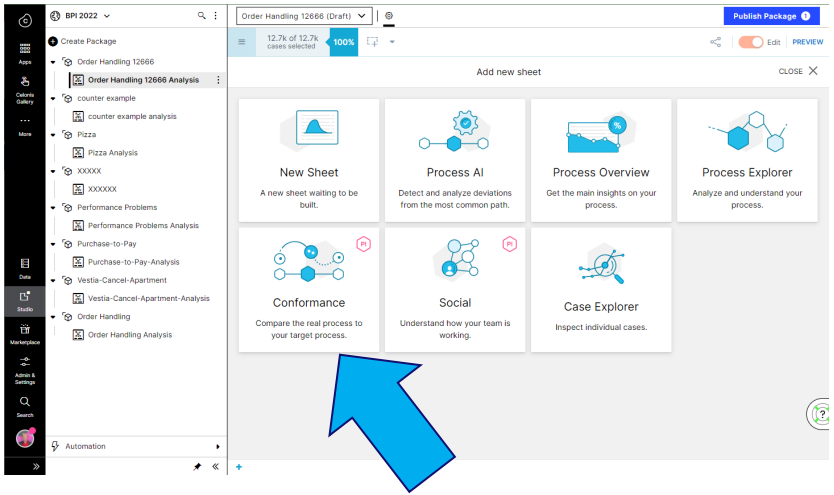
Log view

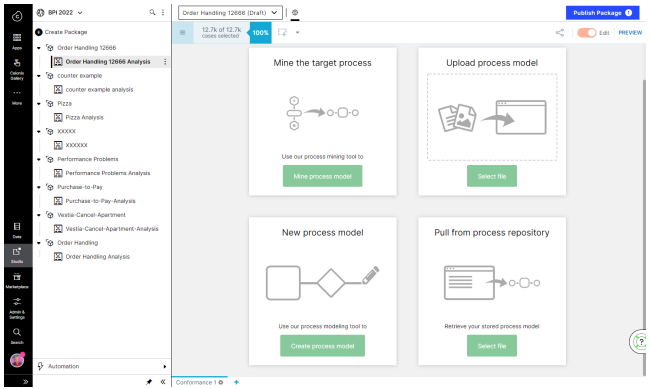


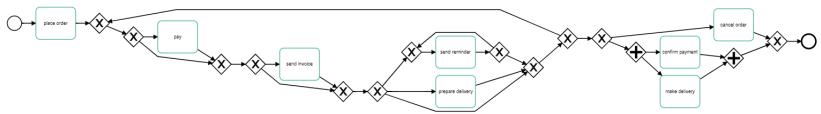
Chair of Process and Data Science

Conformance checking in Celonis

- Input = BPMN model
- Output = List of violations
- Internally: A variant of token-based replay

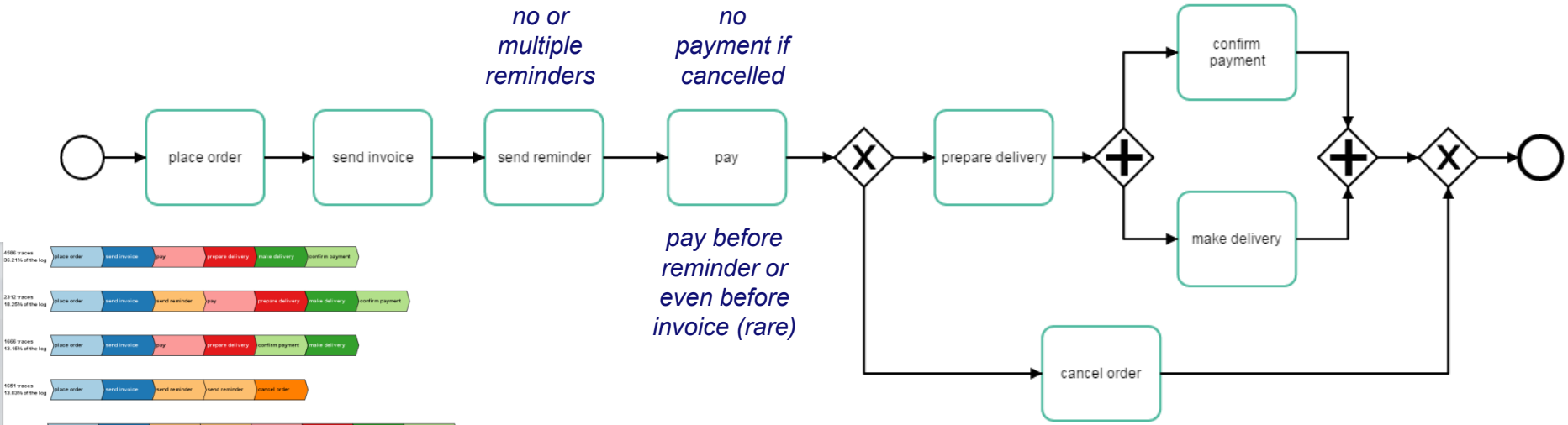


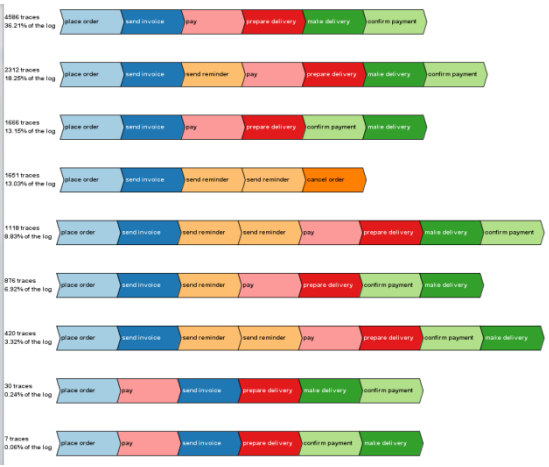




Chair of Process and Data Science

BPMN model (hand-made)





Chair of Process and Data Science

Conformance
Overview
Whitelist
Edit process model
KPIs

Conformance overview

Timeframe: All time
From: 2015-01-04
To: 2021-04-27

Statistics about conformance

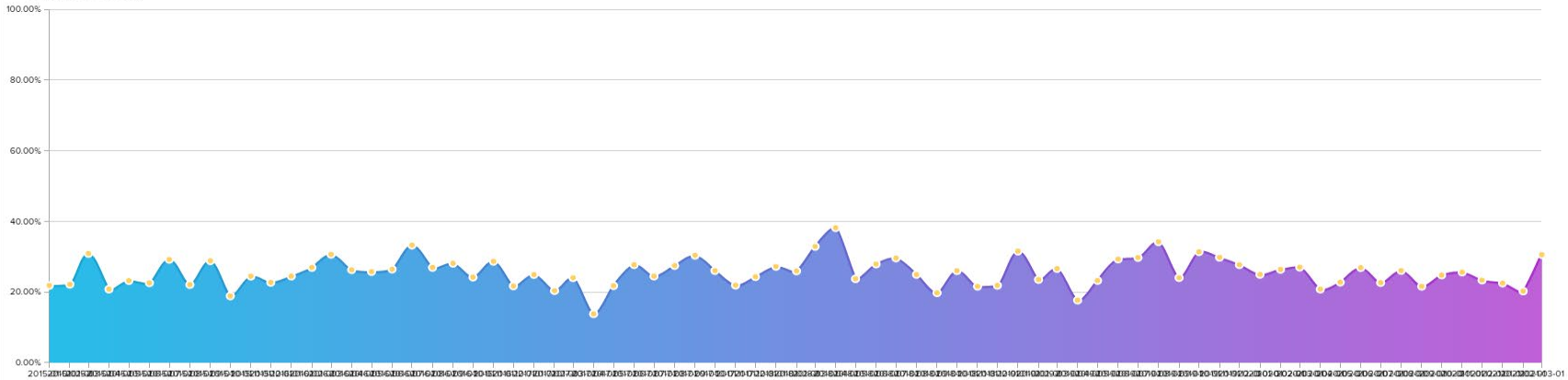
Conforming cases (%)
25%
Cases conforming

Conforming cases
3.19k vs 9.48k
Cases conforming

Violations
4
found in process model

Whitelisted violations
0
configured in whitelist

Conformance history



KPIs for violating vs. conforming cases

Violating cases
Conforming cases

Throughput time
22.3 vs 21.2 Days
Violations increased throughput time by 1.1 Days

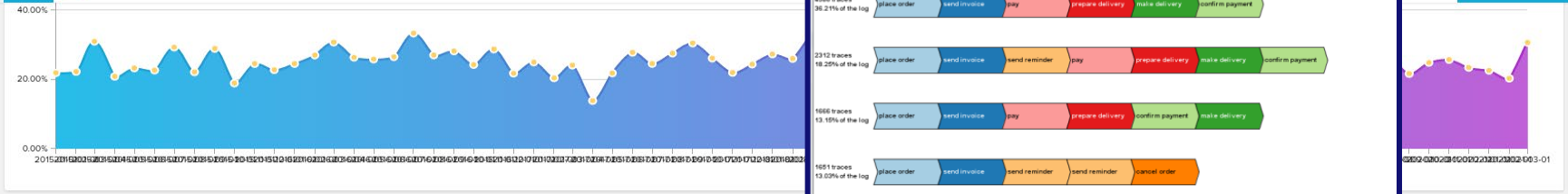
Steps per case
6.2 vs 7.0
Violations decreased steps per case by 0.8

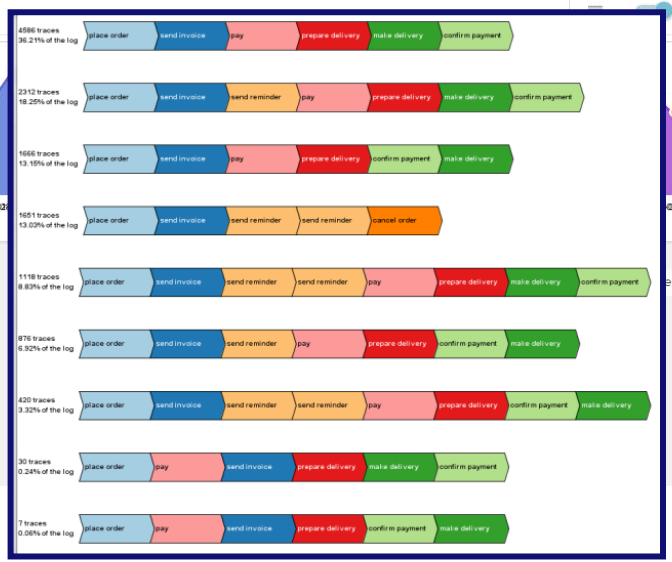
Violations

13k of 13k cases selected | 100% | EDIT | Reset selections

Conformance

- Overview
- Whitelist
- Edit process model
- KPIs





KPIs for violating vs. conforming cases

Conforming cases

Throughput time

22.3 vs 21.2 Days

Violations increased throughput time by 1.1 Days

Steps per case

6.2 vs 7.0

Violations decreased steps per case by 0.8

Violations

49% of cases

send invoice is followed by *pay*

Add to whitelist | View cases in ...

Effect on throughput time: 0 Days shorter

Effect on steps per case: - 1.0 Steps per case

25% of cases

send reminder is followed by *send reminder*

Add to whitelist | View cases in ...

Effect on throughput time: 4 Days longer

Effect on steps per case: - 0.6 Steps per case

13% of cases

send reminder is followed by *cancel order*

Add to whitelist | View cases in ...

Effect on throughput time: 1 Days longer

Effect on steps per case: - 2.0 Steps per case

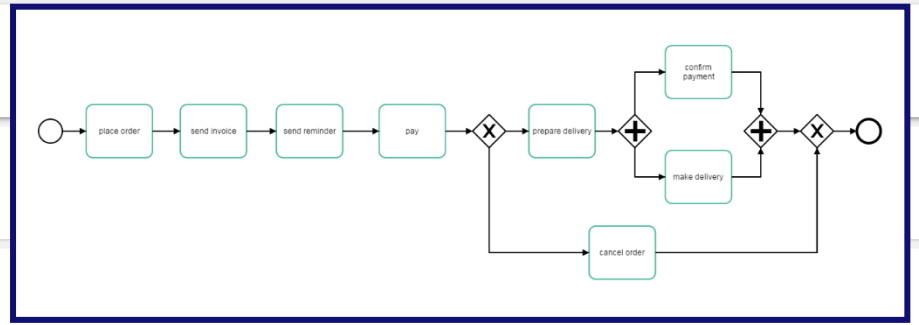
0% of cases

place order is followed by *pay*

Add to whitelist | View cases in ...

Effect on throughput time: 1 Days shorter

Effect on steps per case: - 1.0 Steps per case



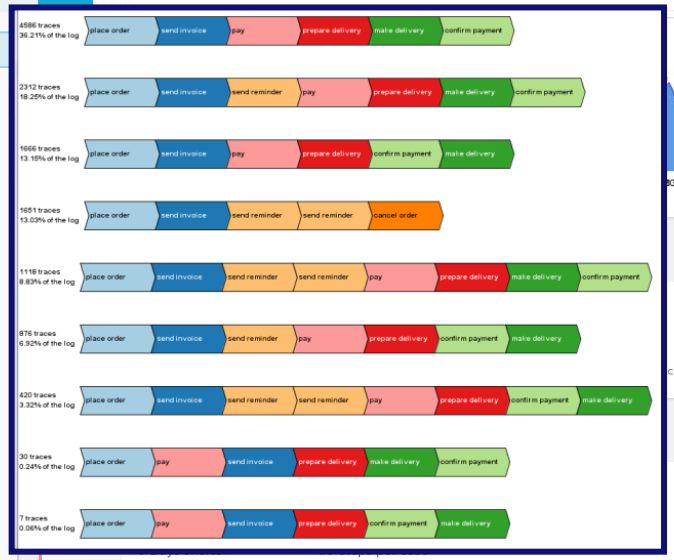
Process overview | Process explorer | Conformance (problems) | Conformance (underfitting) | PI Social 4 | New App 5 | Case Explorer 6 | +

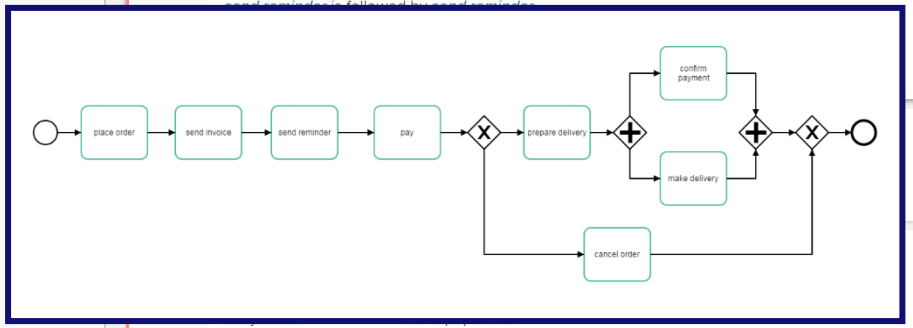
13k of 13k cases selected

100%

Conformance

- Overview
- Whitelist
- Edit process model
- KPIs





Process overview | Process explorer | Conformance (problems) | Conformance (underfitting) | PI Social 4 | New App 5 | Case Explorer 6 +

Violations

49% of cases

send invoice is followed by *pay*

Add to whitelist View cases in ...

Effect on throughput time	Effect on steps per case
0 Days shorter	- 1.0 Steps per case

25% of cases

send reminder is followed by *send reminder*

Add to whitelist View cases in ...

Effect on throughput time	Effect on steps per case
4 Days longer	- 0.6 Steps per case

13% of cases

send reminder is followed by *cancel order*

Add to whitelist View cases in ...

Effect on throughput time	Effect on steps per case
1 Days longer	- 2.0 Steps per case

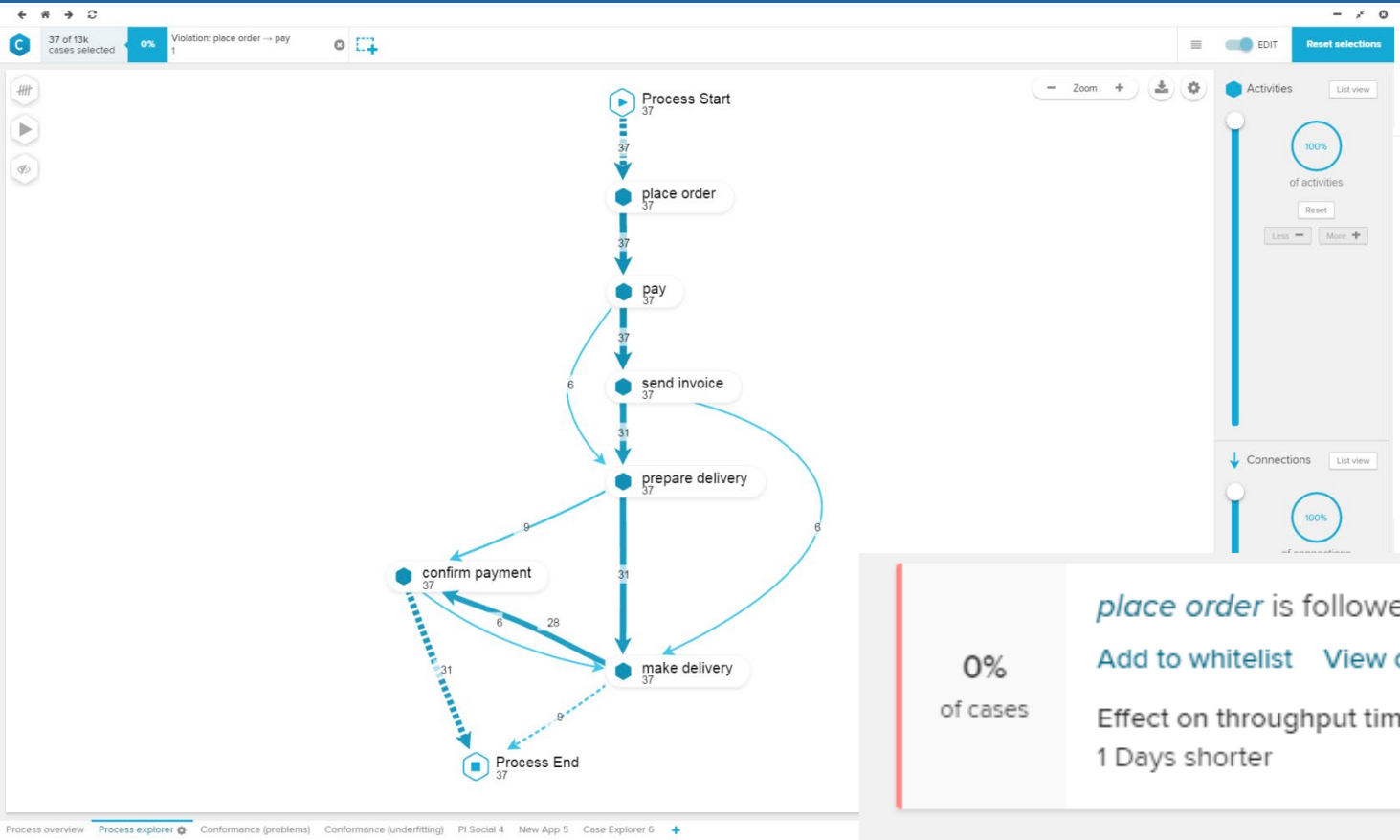
0% of cases

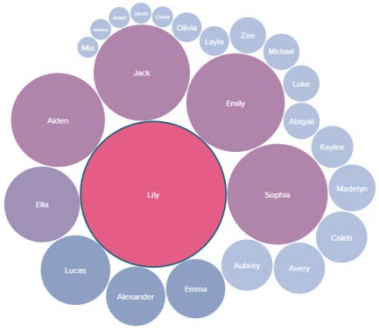
place order is followed by *pay*

Add to whitelist View cases in ...

Effect on throughput time	Effect on steps per case
1 Days shorter	- 1.0 Steps per case

Drill-down on most rare violation (37 cases)





0%
of cases

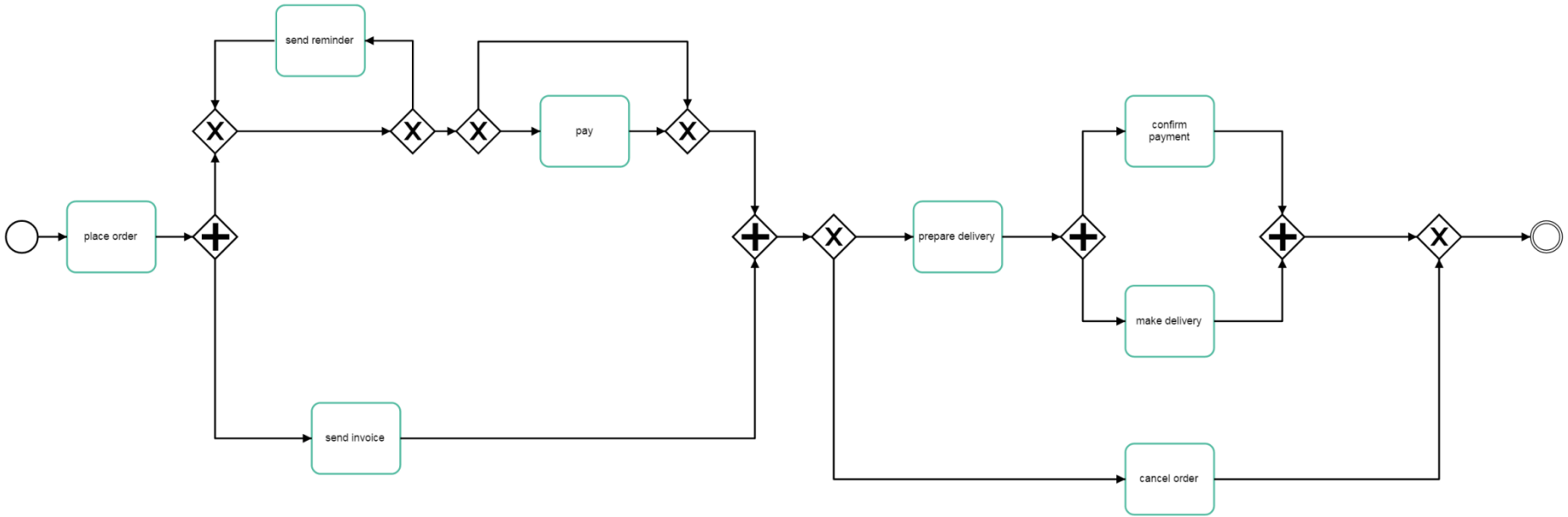
place order is followed by *pay*

Add to whitelist View cases in ...

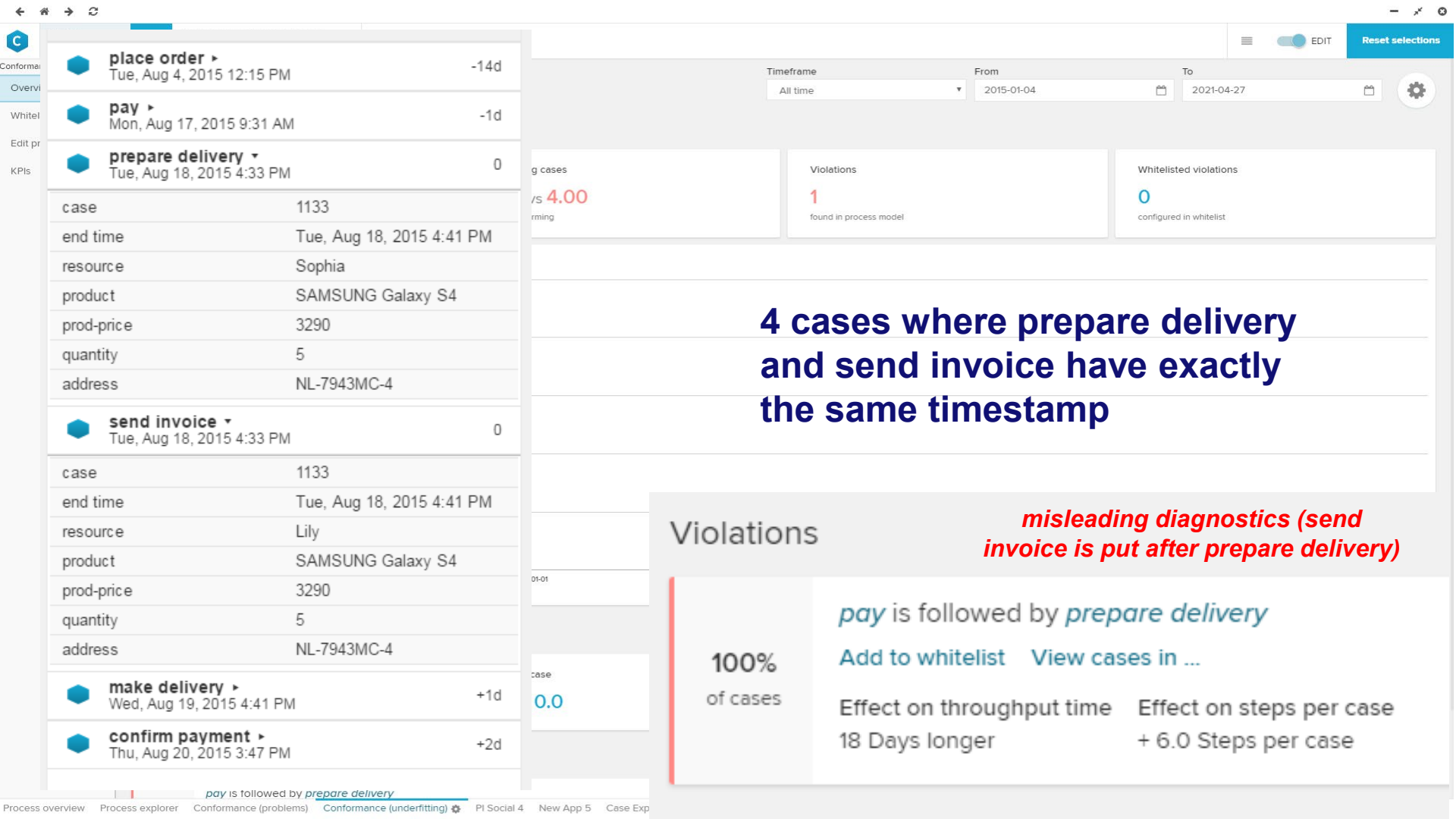
Effect on throughput time	Effect on steps per case
1 Days shorter	- 1.0 Steps per case

“Almost” perfectly fitting model

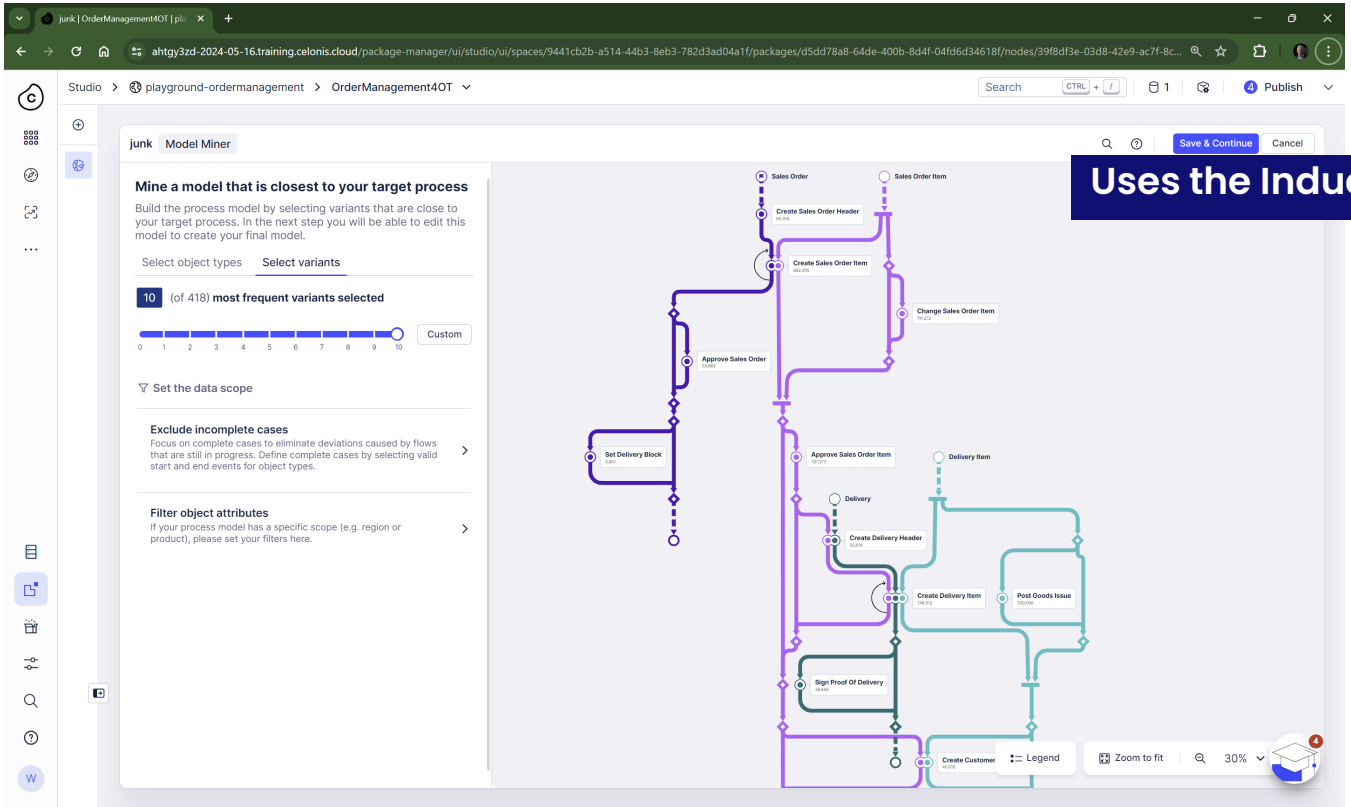
(not expecting any deviations)



Chair of Process and Data Science


place order
Tue, Aug 4, 2015 12:15 PM
-14d
pay
Mon, Aug 17, 2015 9:31 AM
-1d
prepare delivery
Tue, Aug 18, 2015 4:33 PM
0
case 1133
end time Tue, Aug 18, 2015 4:41 PM
resource Sophia
product SAMSUNG Galaxy S4
prod-price 3290
quantity 5
address NL-7943MC-4
send invoice
Tue, Aug 18, 2015 4:33 PM
0
case 1133
end time Tue, Aug 18, 2015 4:41 PM
resource Lily
product SAMSUNG Galaxy S4
prod-price 3290
quantity 5
address NL-7943MC-4
make delivery
Wed, Aug 19, 2015 4:41 PM
+1d
confirm payment
Thu, Aug 20, 2015 3:47 PM
+2d
EDIT
Reset selections
Timeframe
All time
From
2015-01-04
To
2021-04-27
Violations
1
found in process model
Whitelisted violations
0
configured in whitelist
4 cases where prepare delivery and send invoice have exactly the same timestamp
Violations
misleading diagnostics (send invoice is put after prepare delivery)
pay is followed by prepare delivery
100%
of cases
Add to whitelist
View cases in ...
Effect on throughput time
18 Days longer
Effect on steps per case
+ 6.0 Steps per case
pay is followed by prepare delivery
Process overview
Process explorer
Conformance (problems)
Conformance (underfitting)
PI Social 4
New App 5
Case Exp

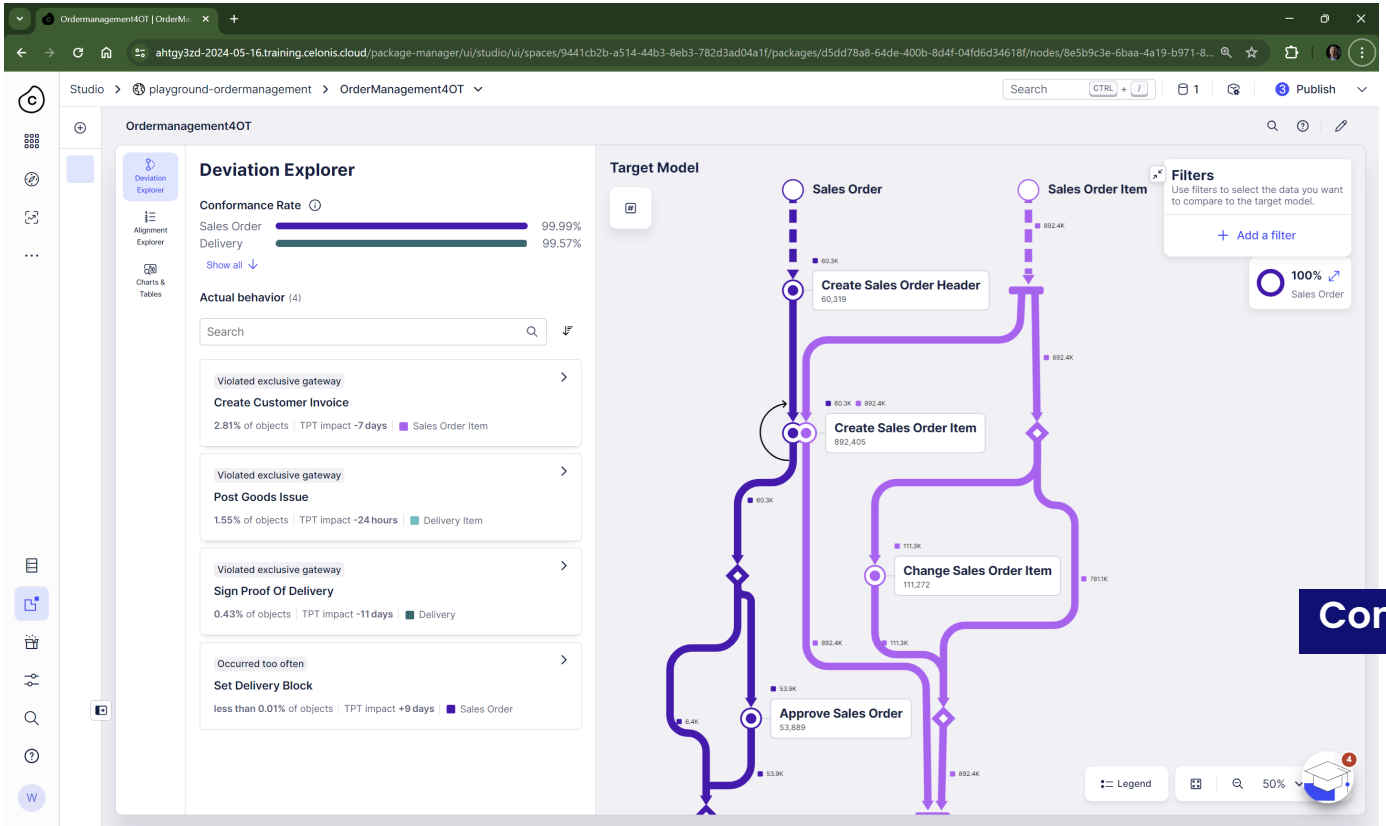
Process Adherence Management (PAM): Discover Object-Centric BPMN



Uses the Inductive Mining algorithm

Chair of Process and Data Science

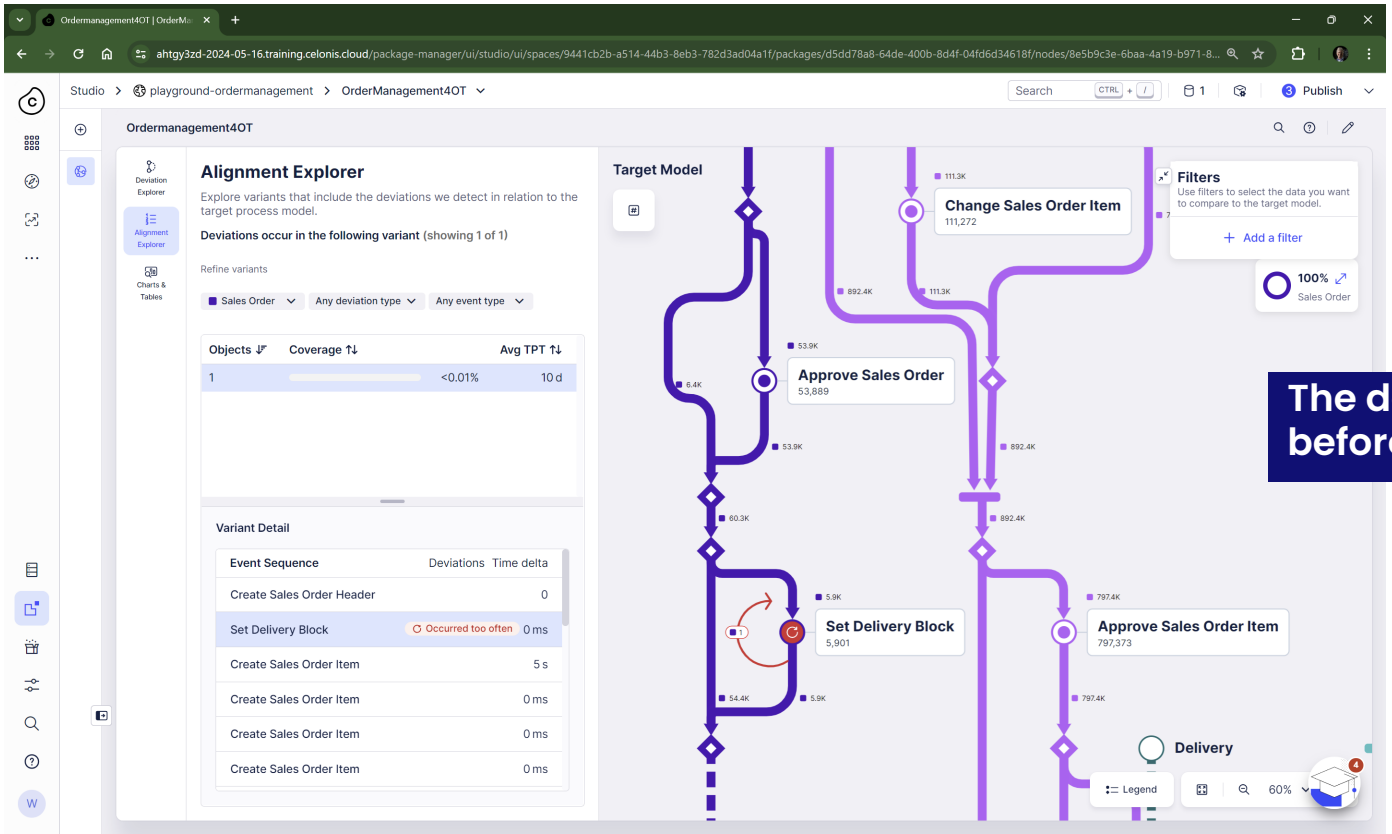
Process Adherence Management (PAM): Check Conformance



Computes alignments

Chair of Process and Data Science

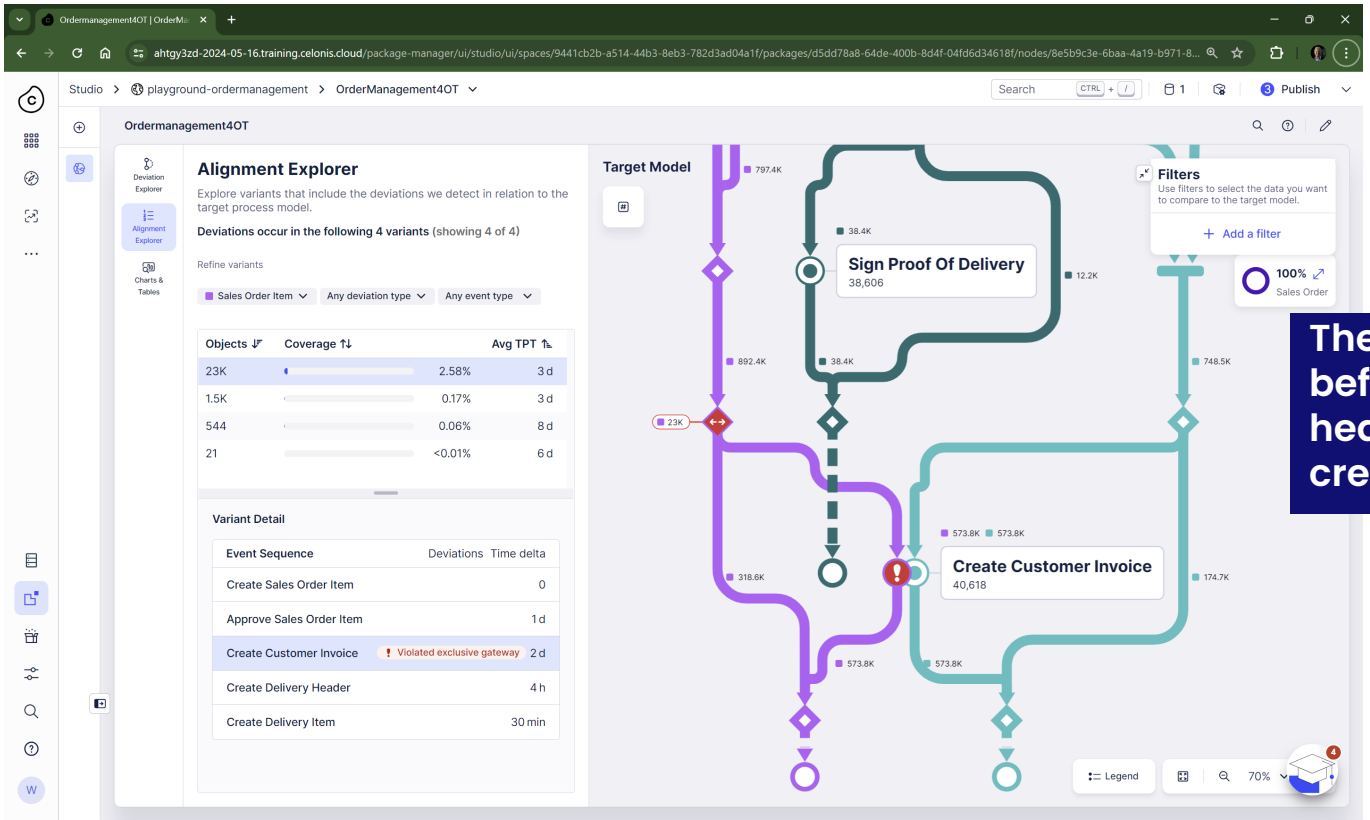
Process Adherence Management (PAM): Show Alignments



The delivery block is set before items are created

Chair of Process and Data Science

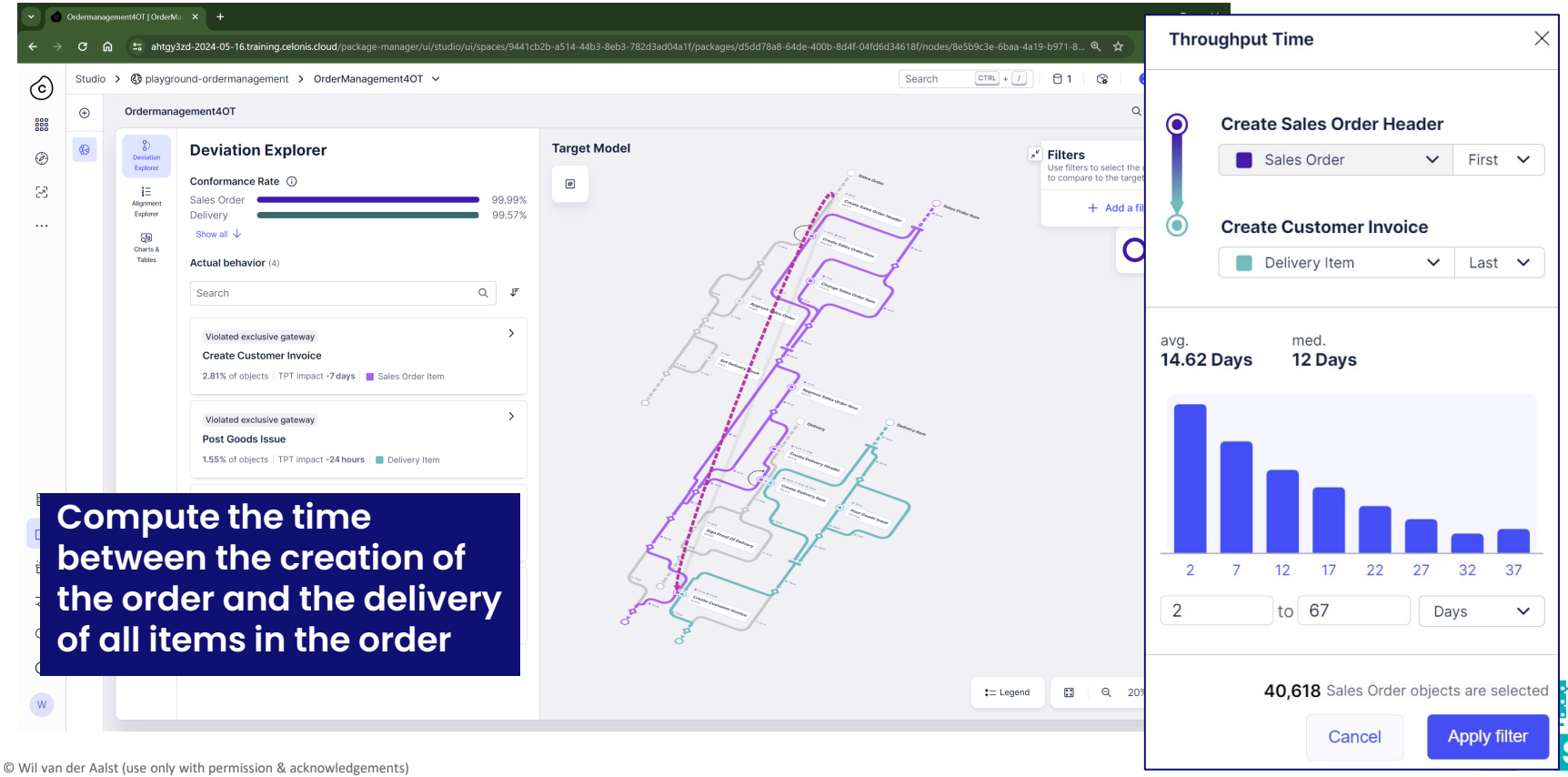
Process Adherence Management (PAM): Show Alignments



The invoice is created before the delivery header and item are created.

Chair of Process and Data Science

Process Adherence Management (PAM): Analyze Performance



Compute the time between the creation of the order and the delivery of all items in the order


Demo

Applications of Process Mining

Process Mining Is Used Everywhere

Technology

Financial Services & Insurance

Life Sciences & Chemicals

Consumer & Retail

Manufacturing

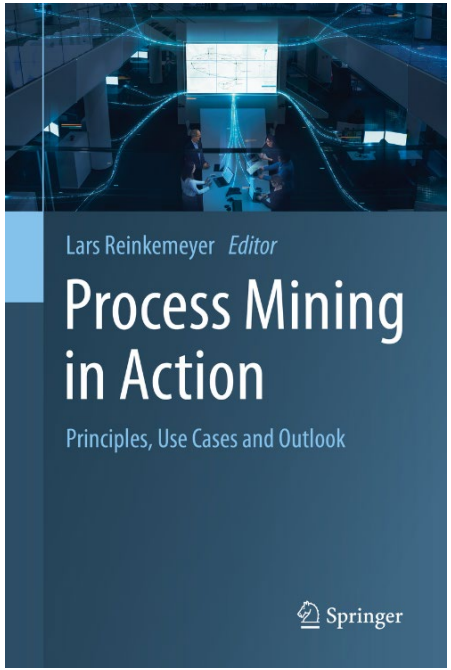
Telecommunications & Media

Energy & Utilities

Oil & Gas

Chair of Process and Data Science

Some Case Studies



Part II Best Practice Use Cases

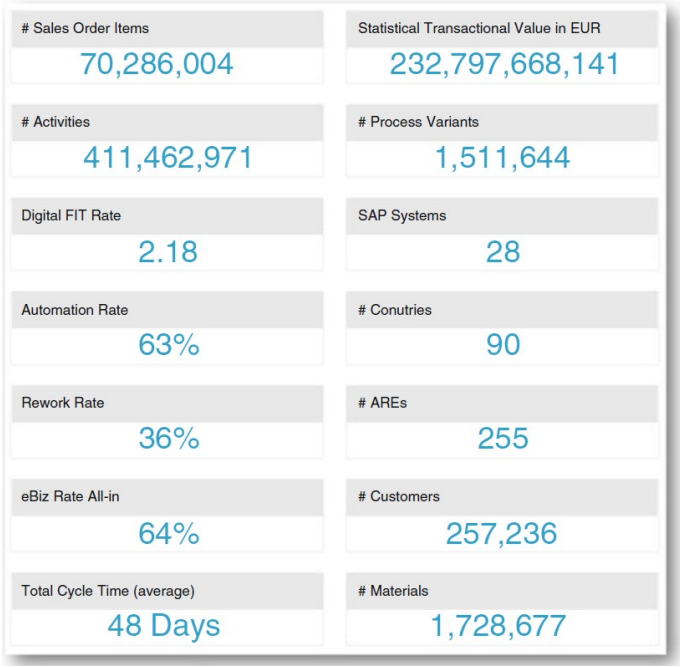
Example: Siemens Order-to-Cash (O2C)

Siemens: Driving Global Change with the Digital Fit Rate in Order2Cash

Gia-Thi Nguyen

“Using out-of-the-box reports from the Process Mining tool, the results included an increase in automation by 24% and a reduction in manual rework by 11%.”

“Over 70 million sales order items across data from 90 countries with over 1.5 million process variants.”

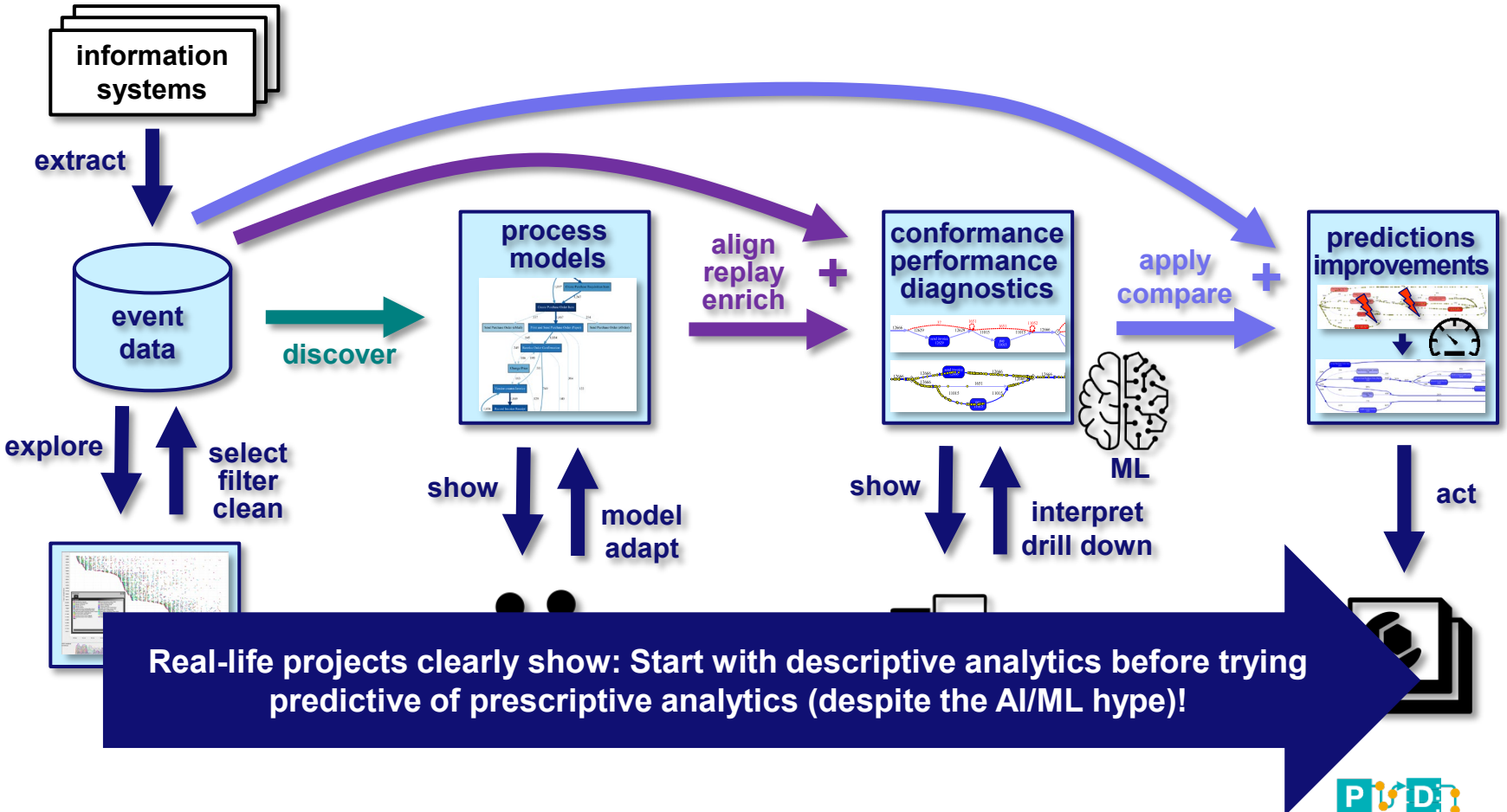


Chair of Process and Data Science

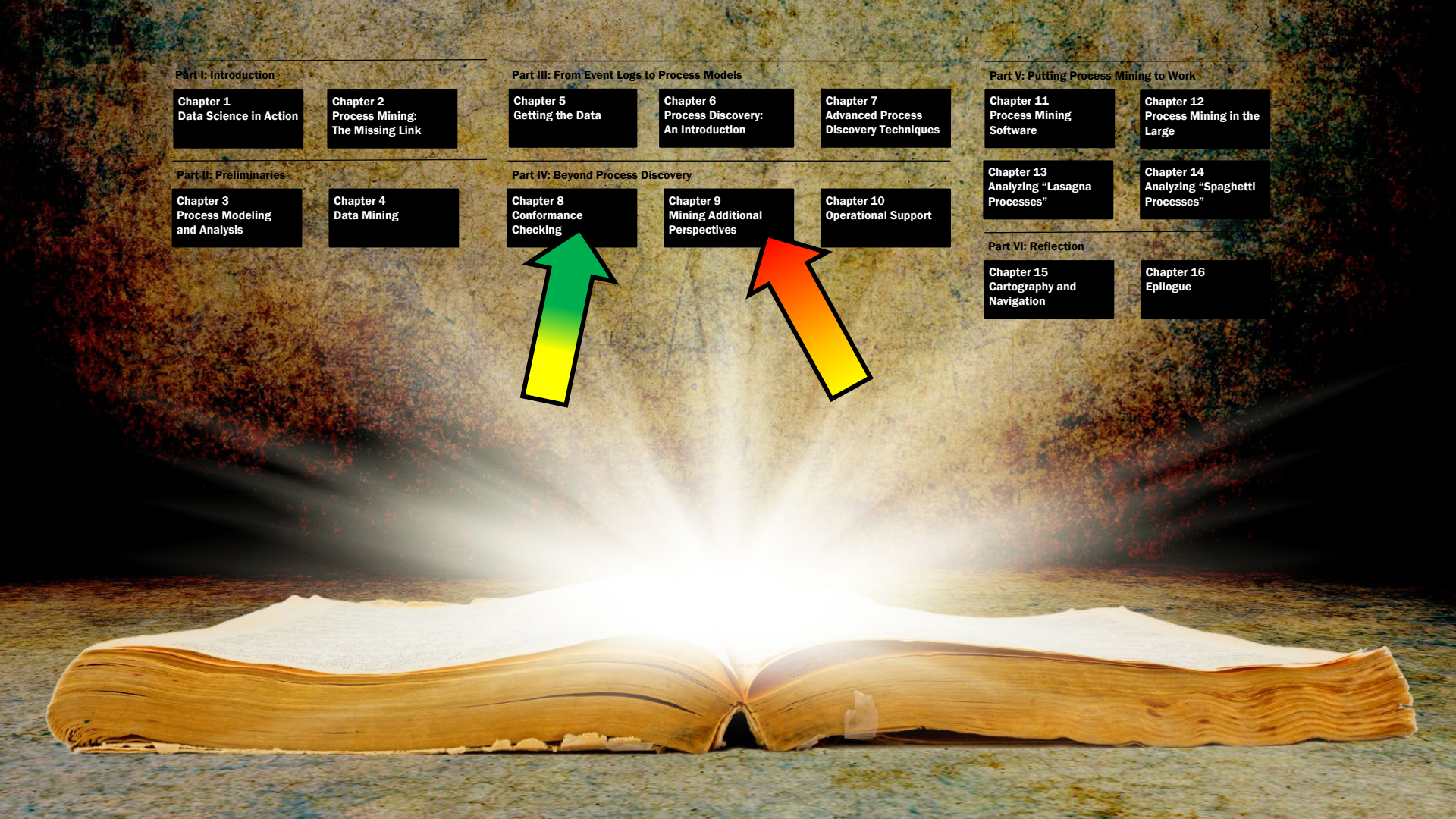
Example: BMW

More than **850 registered users** on BMW Celonis Process Mining Infrastructure
49 of BMW's 50 data models are in use
More than **10 terrabyte of raw relational data** is handled on the Process Mining Databases
More than **6.4 million process variants** are currently being analyzed within the BMW Process Mining Infrastructure
More than **500 million events** are analyzed on the BMW Celonis Process Mining Infrastructure
More than **30 million cases** are analyzed on the BMW Celonis Process Mining Infrastructure
Between **400 – 600 analytical views** are provided **each day** for the Process Mining Users of BMW
600 unique analyses are currently available for the 49 data models

Example: Lufthansa



Chair of Process and Data Science


Part I: Introduction
Chapter 1
Data Science in Action
Chapter 2
Process Mining:
The Missing Link
Part II: Preliminaries
Chapter 3
Process Modeling
and Analysis
Chapter 4
Data Mining
Part III: From Event Logs to Process Models
Chapter 5
Getting the Data
Chapter 6
Process Discovery:
An Introduction
Chapter 7
Advanced Process
Discovery Techniques
Part IV: Beyond Process Discovery
Chapter 8
Conformance
Checking
Chapter 9
Mining Additional
Perspectives
Chapter 10
Operational Support
Part V: Putting Process Mining to Work
Chapter 11
Process Mining
Software
Chapter 12
Process Mining in the
Large
Chapter 13
Analyzing “Lasagna
Processes”
Chapter 14
Analyzing “Spaghetti
Processes”
Part VI: Reflection
Chapter 15
Cartography and
Navigation
Chapter 16
Epilogue

ID	Topic	Date	Date	Place
Lecture 1	Introduction to Process Mining	08.04.24	Monday	AH V
Lecture 2	Data Science: Supervised Learning	09.04.24	Tuesday	AH V
Exercise 1	*Tool Introduction*	09.04.24	Tuesday	AH III
Lecture 3	Data Science: Unsupervised Learning and Evaluation	15.04.24	Monday	AH V
Lecture 4	Introduction to Process Discovery	16.04.24	Tuesday	AH V
Exercise 2	*Data Mining*	16.04.24	Tuesday	AH III
Lecture 5	Alpha Algorithm 1	22.04.24	Monday	AH V
Lecture 6	Alpha Algorithm 2	23.04.24	Tuesday	AH V
Exercise 3	*Petri Nets*	23.04.24	Tuesday	AH III
Lecture 7	Model Quality Representation	29.04.24	Monday	AH V
Lecture 8	Heuristic Mining	30.04.24	Tuesday	AH V
Exercise 4	*Alpha Miner*	30.04.24	Tuesday	AH III
Lecture 9	Region-Based Mining	06.05.24	Monday	AH V
Exercise 5	*Heuristic Mining and Region-Based Mining*	07.05.24	Tuesday	AH III
Lecture 10	Inductive Mining	13.05.24	Monday	AH V
Lecture 11	Event Data and Exploration	14.05.24	Tuesday	AH V
Exercise 6	*Inductive Mining*	14.05.24	Tuesday	AH III
Lecture 12	Conformance Checking 1	27.05.24	Monday	AH V
Lecture 13	Conformance Checking 2	28.05.24	Tuesday	AH V
	Q&A Session Assignment Part I	28.05.24	Tuesday	AH III
	Deadline Assignment Part I	02.06.24	Sunday	
Exercise 7	*Footprint and Token-Based Replay (Exercise)*	03.06.24	Monday	AH V
Exercise 8	*Alignments (Exercise)*	04.06.24	Tuesday	AH V
Lecture 14	Decision Mining	10.06.24	Monday	AH V
Lecture 15	*Celonis Guest Lecture*	11.06.24	Tuesday	AH V
Exercise 9	*Decision Mining*	11.06.24	Tuesday	AH III
Lecture 16	Performance Analysis and Organizational Mining	17.06.24	Monday	AH V
Exercise 10	*Performance Analysis (Exercise)*	18.06.24	Tuesday	AH V
Exercise 11	*Organizational Mining*	18.06.24	Tuesday	AH III
Exercise 12	*Celonis Case Study*	24.06.24	Monday	AH V
Lecture 17	Operational Support and Process Mining Applications	01.07.24	Monday	AH V
Lecture 18	Distributed, Streaming, and Comparative Process Mining	02.07.24	Tuesday	AH V
Exercise 13	*Operational Process Mining*	02.07.24	Tuesday	AH III
Lecture 19	Closing	08.07.24	Monday	AH V
	Q&A Session Assignment Part II	09.07.24	Tuesday	AH III
	Deadline Assignment Part II	14.07.24	Sunday	
	Q&A Session Exam	16.07.24	Tuesday	AH III

Chair of Process and Data Science