Passages in Big Data

Partitioning Event Logs and Process Models to Speed Up Process Mining Algorithms

Wil van der Aalst

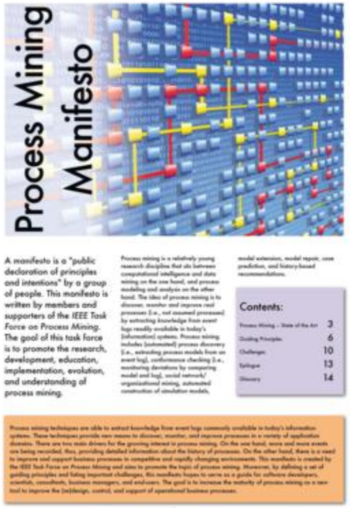
www.processmining.org

Technische Universiteit
Eindhoven
University of Technology

Where innovation starts

Advances in Process Mining

- Many process discovery and conformance checking algorithms and tools are available (cf. the various ProM packages).
- Also commercial software based on these ideas: Disco (Fluxicon), Reflect (Futura), BPMOne (Pallas Athena/Perceptive), ARIS Process Performance Manager (Software AG), Futura Reflect (Futura Technology), Interstage Automated Process Discovery (Fujitsu), QPR ProcessAnalyzer/Analysis (QPR Software), flow (fourspark), Discovery Analyst (StereoLOGIC), etc.
- We applied process mining in over 100 organizations.



Process Mining Manifesto

A manifesto is a "public declaration of principles and intentions" by a group of people. This manifesto is written by members and supporters of the IEEE Task Force on Process Mining. The goal of this task force is to promote the research, development, education, implementation, evolution, and understanding of process mining.

Process mining is a relatively young research discipline that sits between computational intelligence and data mining on the one hand, and process modeling and analysis on the other hand. The idea of process mining is to discover, monitor and improve real processes (i.e., not assumed processes) by extracting knowledge from event logs readily available in today's (information) systems. Process mining includes (automated) process discovery (i.e., extracting process models from an event log), conformance checking (i.e., monitoring deviations by comparing model and log), social network/ organizational mining, automated construction of simulation models, model extension, model repair, case prediction, and history-based recommendations.

Contents:

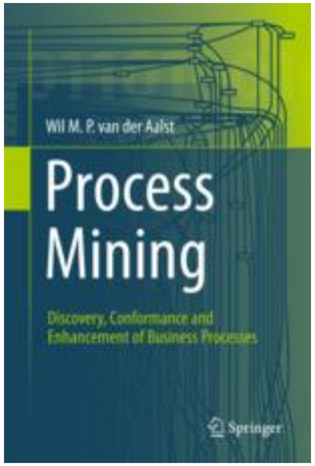
Process Mining – State of the Art	3
Guiding Principles	6
Challenges	10
Epilogue	13
Glossary	14

Process mining techniques are able to extract knowledge from event logs commonly available in today's information systems. These techniques provide new means to discover, monitor, and improve processes in a variety of application domains. There are two main drivers for the growing interest in process mining. On the one hand, more and more events are being recorded, thus, providing detailed information about the history of processes. On the other hand, there is a need to improve and support business processes in competitive and rapidly changing environments. This manifesto is created by the IEEE Task Force on Process Mining and aims to promote the topic of process mining. Moreover, by defining a set of guiding principles and listing important challenges, this manifesto hopes to serve as a guide for software developers, scientists, consultants, business managers, and end-users. The goal is to increase the maturity of process mining as a new tool to improve the (re)design, control, and support of operational business processes.

1

More than 75 people involving more than 50 organizations created the Process Mining Manifesto in the context of the IEEE Task Force on Process Mining.

Available in 13 languages


Wil M. P. van der Aalst
Process Mining
Discovery, Conformance and Enhancement of Business Processes
Springer

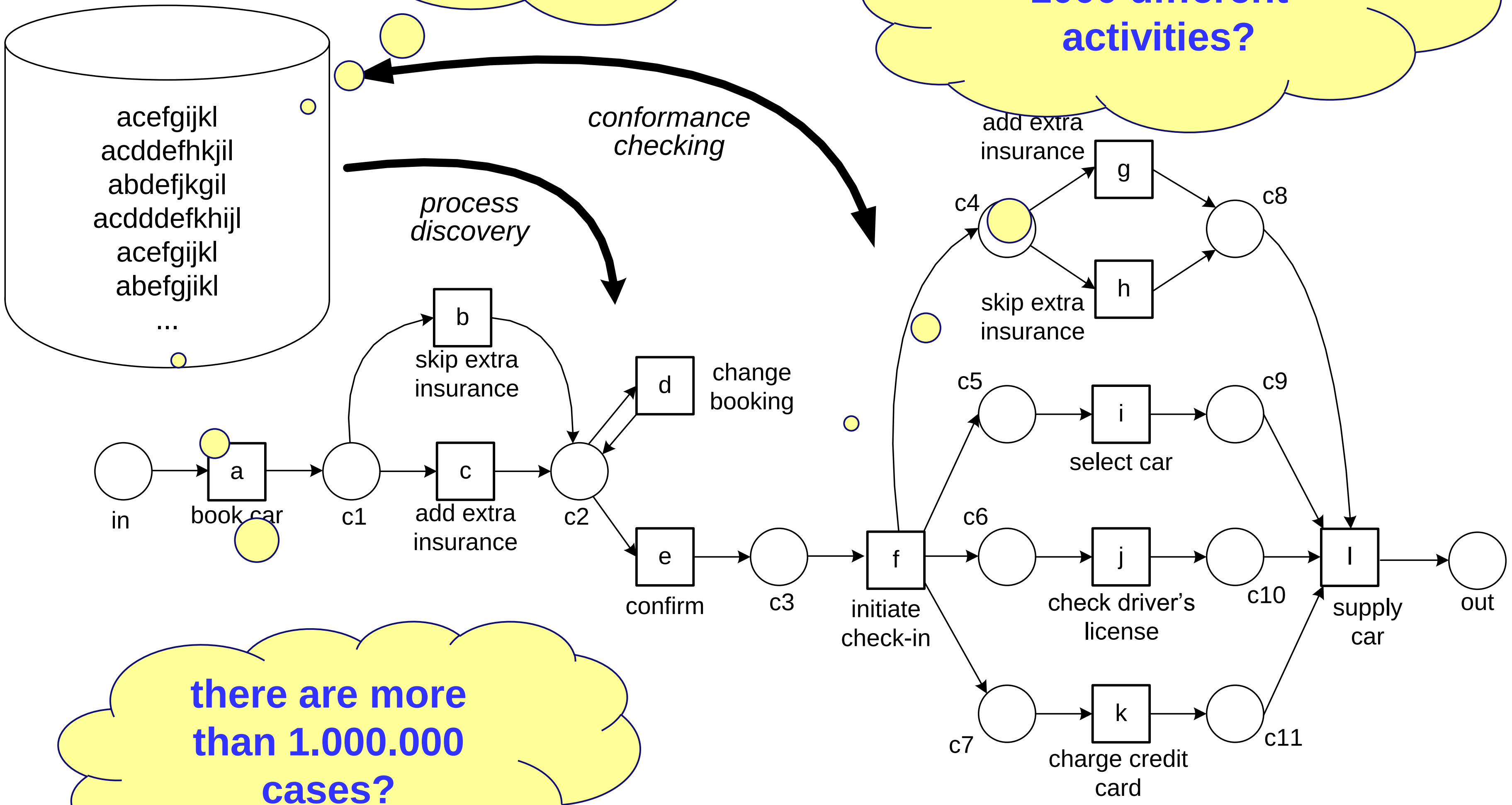
killer app for Petri nets!

Big Data: Opportunities and Challenges

What if?

there are more than 100.000.000 events?

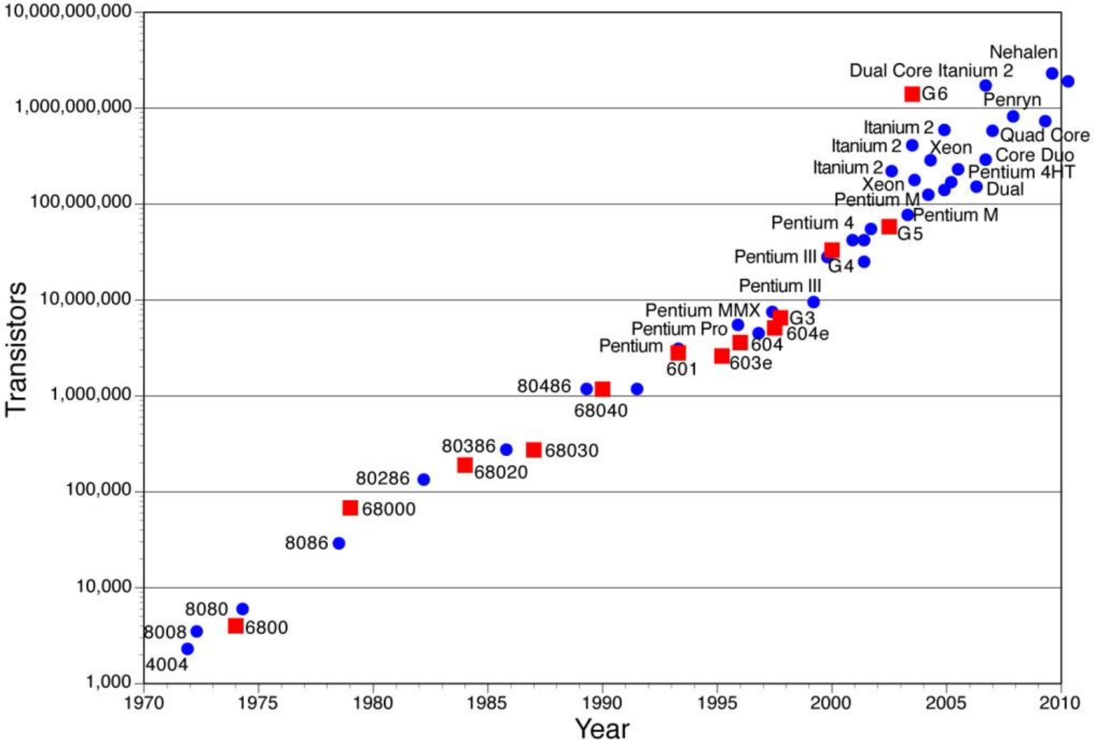
there are more than 1000 different activities?



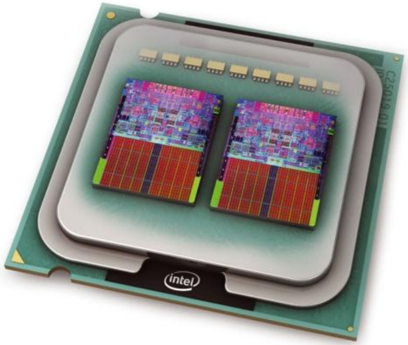
there are more than 1.000.000 cases?

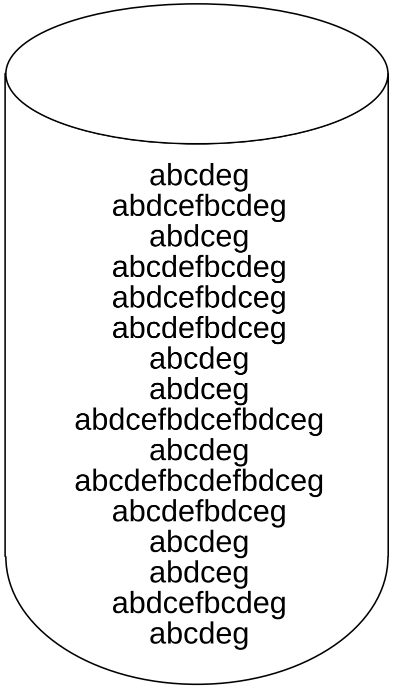
Distributed Computing

- multicore CPU
- manycore GPU
- cluster computing
- grid computing
- cloud computing
- …



How to distribute process discovery?

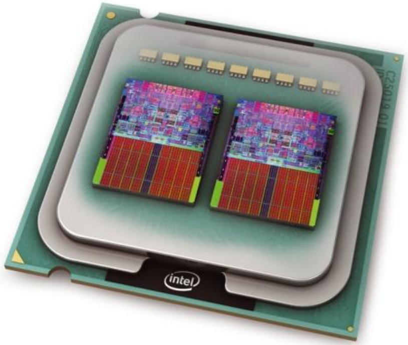


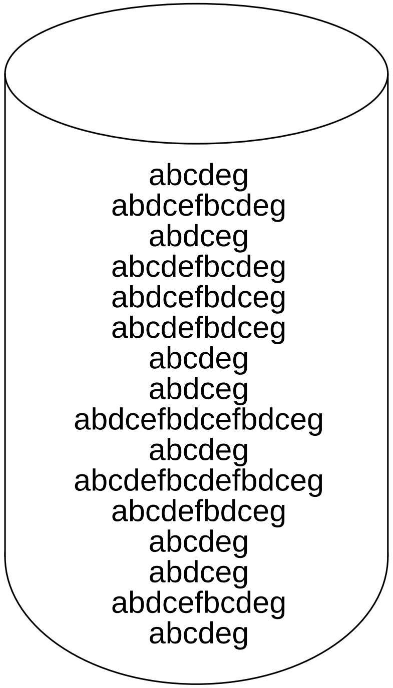


abcdeg
abdcefbcdeg
abdceg
abcdefbcdeg
abdcefbdceg
abcdefbdceg
abcdeg
abdceg
abdcefbdcefbdceg
abcdeg
abcdefbcdefbdceg
abcdefbdceg
abcdeg
abdceg
abdcefbcdeg
abcdeg

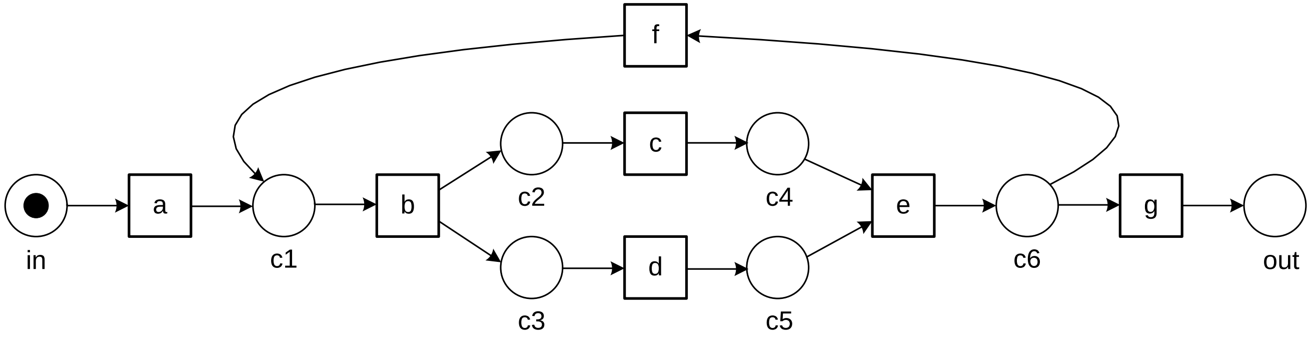
?

How to distribute process discovery?

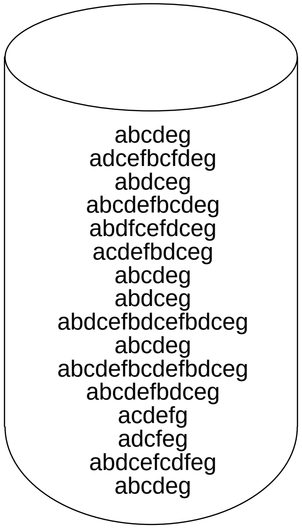




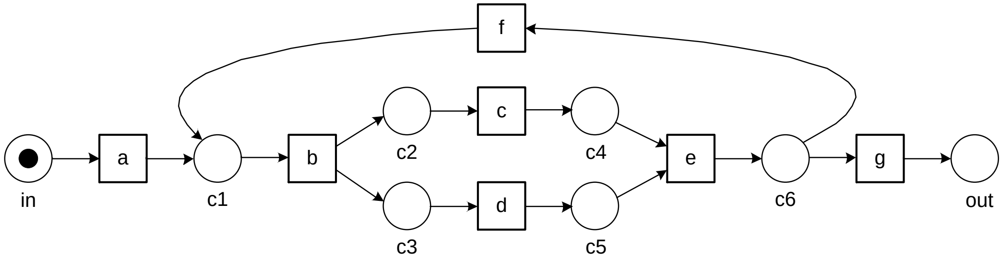
abcdeg
abdcefbcdeg
abdceg
abcdefbcdeg
abdcefbdceg
abcdefbdceg
abcdeg
abdceg
abdcefbdcefbdceg
abcdeg
abcdefbcdefbdceg
abcdefbdceg
abcdeg
abdceg
abdcefbcdeg
abcdeg

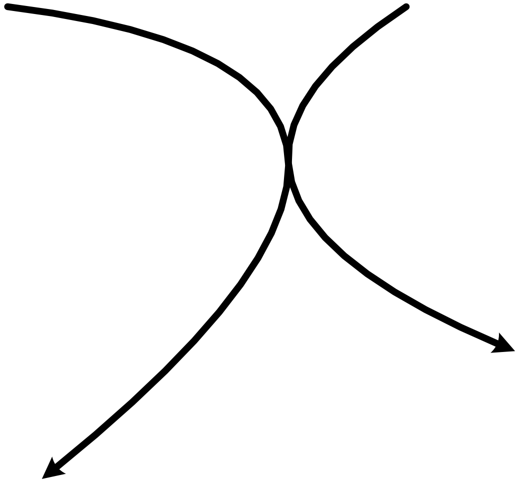


How to distribute conformance checking?



abcdeg
adcefbcfdeg
abdceg
abcdefbcdeg
abdfcefdceg
acdefbdceg
abcdeg
abdceg
abdcefbdcefbdceg
abcdeg
abcdefbcdefbdceg
abcdefbdceg
acdefg
adcfeg
abdcefcdfeg
abcdeg

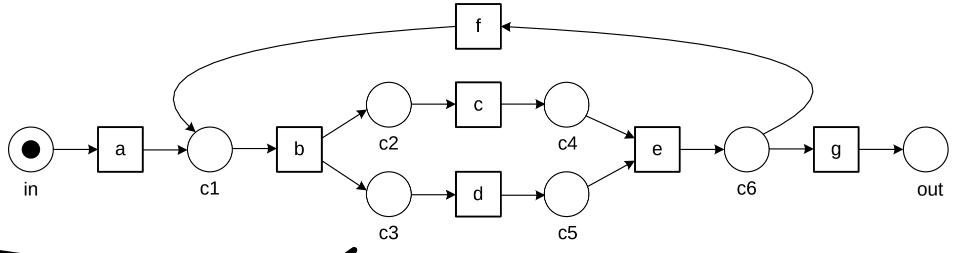


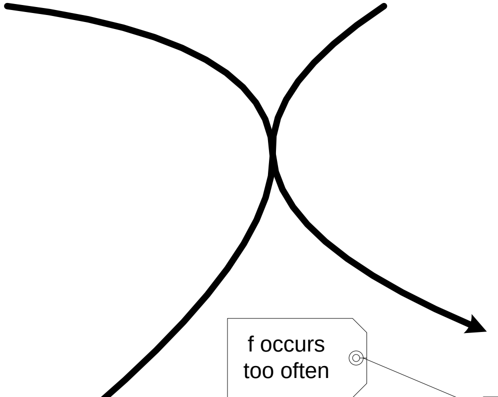


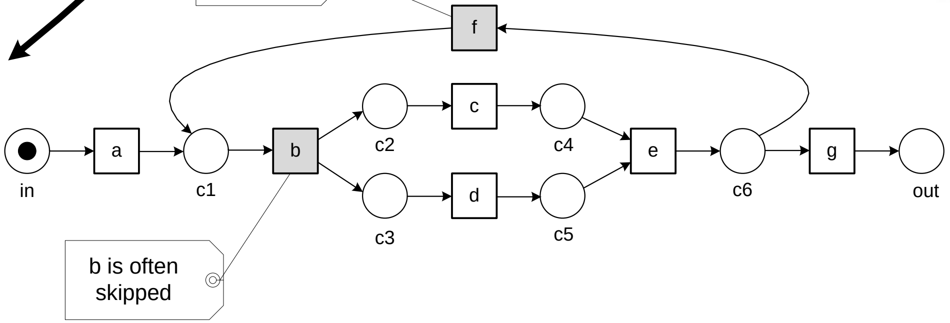
?

How to distribute conformance checking?

abcdeg
adcefbcfdeg
abdceg
abcdefbcdeg
abdfcefdceg
acdefbdceg
abcdeg
abdceg
abdcefbdcefbdceg
abcdeg
abcdefbcdefbdceg
abcdefbdceg
acdefg
adcfeg
abdcefcdfeg
abcdeg







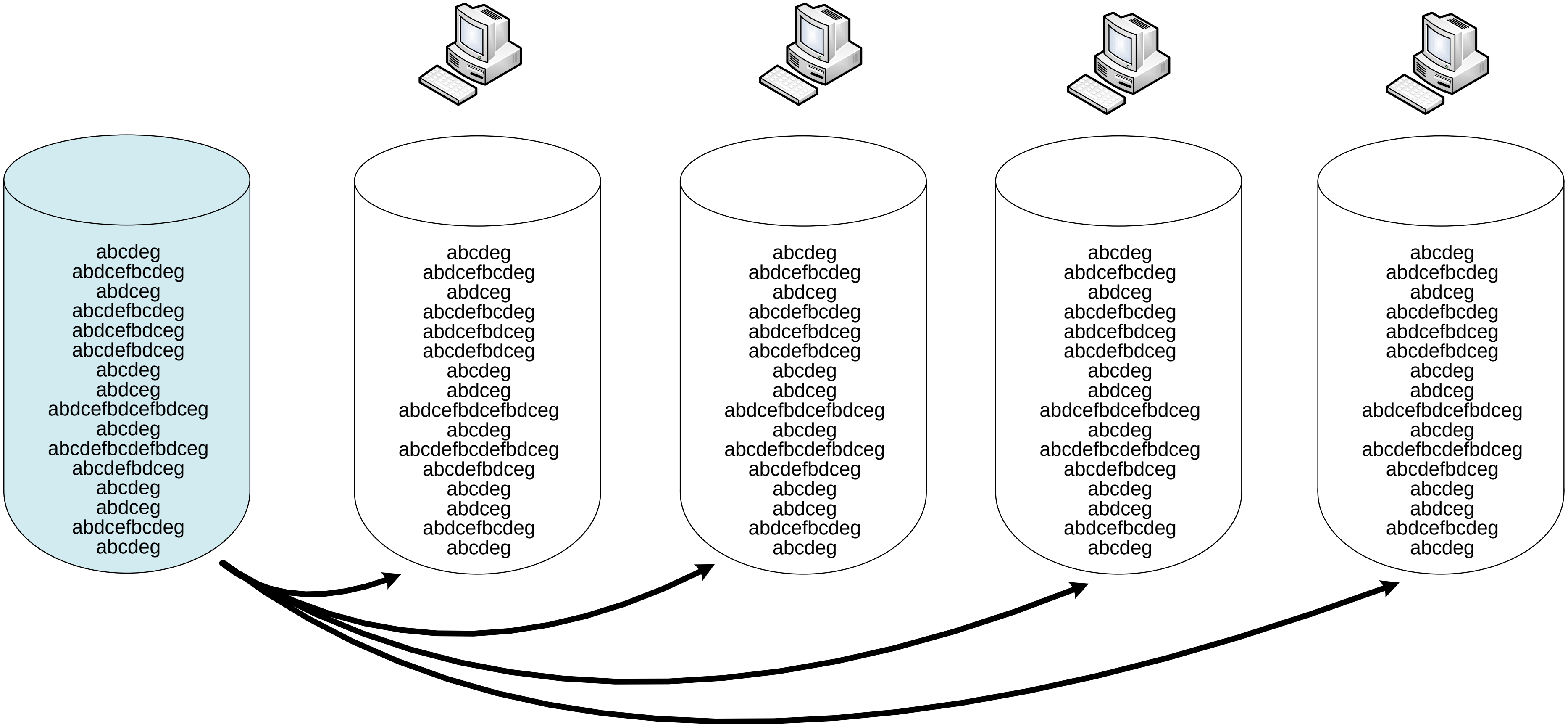
f occurs too often

b is often skipped

abcdeg
abdceg
abcdefbcdeg
abcdeg
abdceg
abdcefbdcefbdceg
abcdeg
abcdefbcdefbdceg
abcdefbdceg
abcdeg

~~adcefbcfdeg~~
~~abdfcefdceg~~
~~acdefbdceg~~
~~acdefg~~
~~adcfeg~~
~~abdcefcdfeg~~

Replication: Same event log on all computing nodes



abcdeg
abdcefbcdeg
abdceg
abcdefbcdeg
abdcefbdceg
abcdefbdceg
abcdeg
abdceg
abdcefbdcefbdceg
abcdeg
abcdefbcdefbdceg
abcdefbdceg
abcdeg
abdceg
abdcefbcdeg
abcdeg

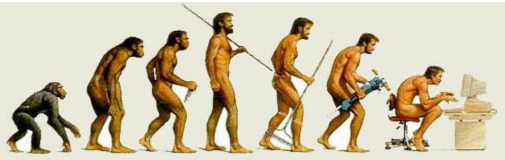
abcdeg
abdcefbcdeg
abdceg
abcdefbcdeg
abdcefbdceg
abcdefbdceg
abcdeg
abdceg
abdcefbdcefbdceg
abcdeg
abcdefbcdefbdceg
abcdefbdceg
abcdeg
abdceg
abdcefbcdeg
abcdeg

abcdeg
abdcefbcdeg
abdceg
abcdefbcdeg
abdcefbdceg
abcdefbdceg
abcdeg
abdceg
abdcefbdcefbdceg
abcdeg
abcdefbcdefbdceg
abcdefbdceg
abcdeg
abdceg
abdcefbcdeg
abcdeg

abcdeg
abdcefbcdeg
abdceg
abcdefbcdeg
abdcefbdceg
abcdefbdceg
abcdeg
abdceg
abdcefbdcefbdceg
abcdeg
abcdefbcdefbdceg
abcdefbdceg
abcdeg
abdceg
abdcefbcdeg
abcdeg

abcdeg
abdcefbcdeg
abdceg
abcdefbcdeg
abdcefbdceg
abcdefbdceg
abcdeg
abdceg
abdcefbdcefbdceg
abcdeg
abcdefbcdefbdceg
abcdefbdceg
abcdeg
abdceg
abdcefbcdeg
abcdeg

Only makes sense if random elements, e.g., genetic process mining.



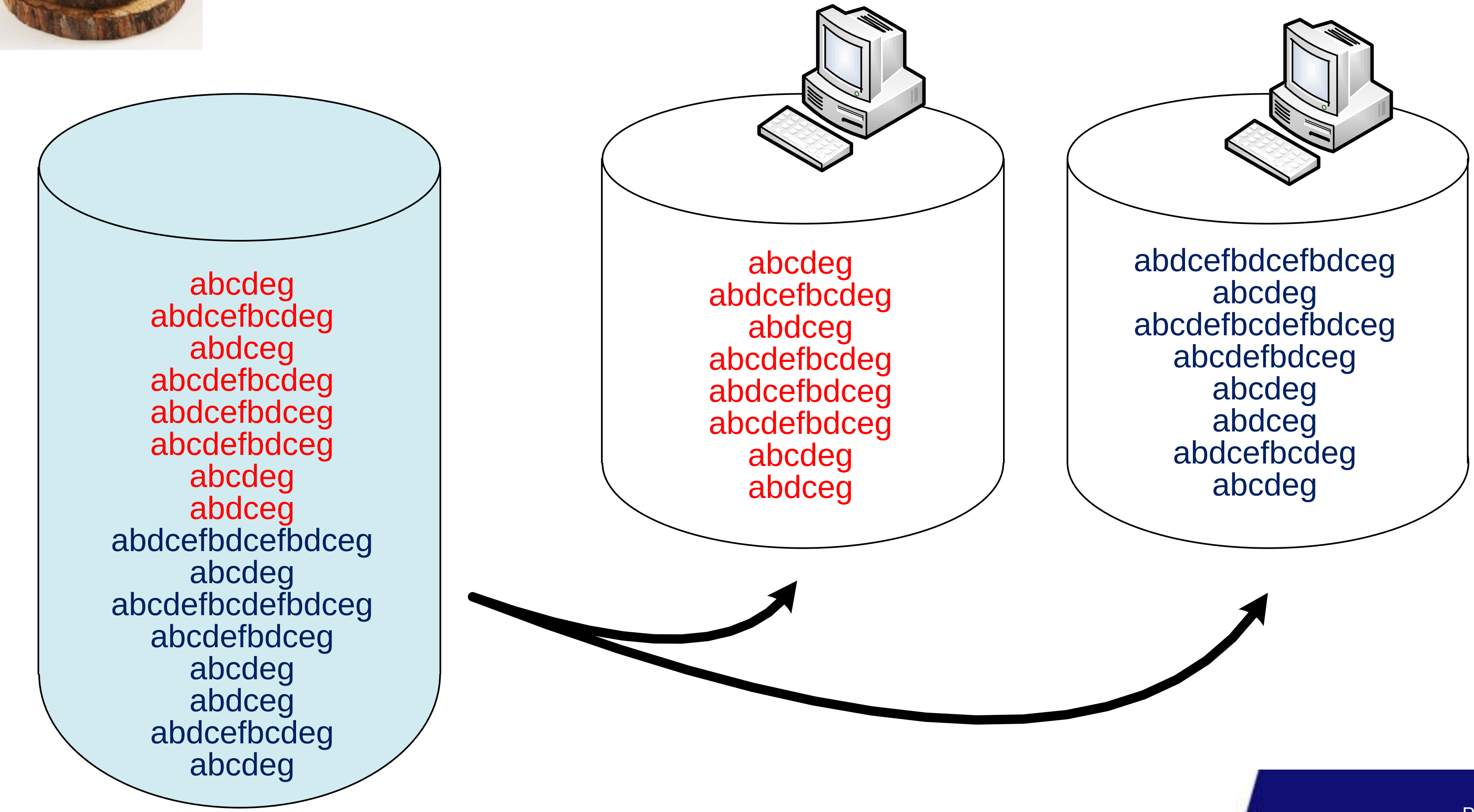
Classification based on partitioning of event log: vertical and horizontal

sets of cases

sets of activities

Vertical distribution I:
Split cases arbitrarily

sets of cases

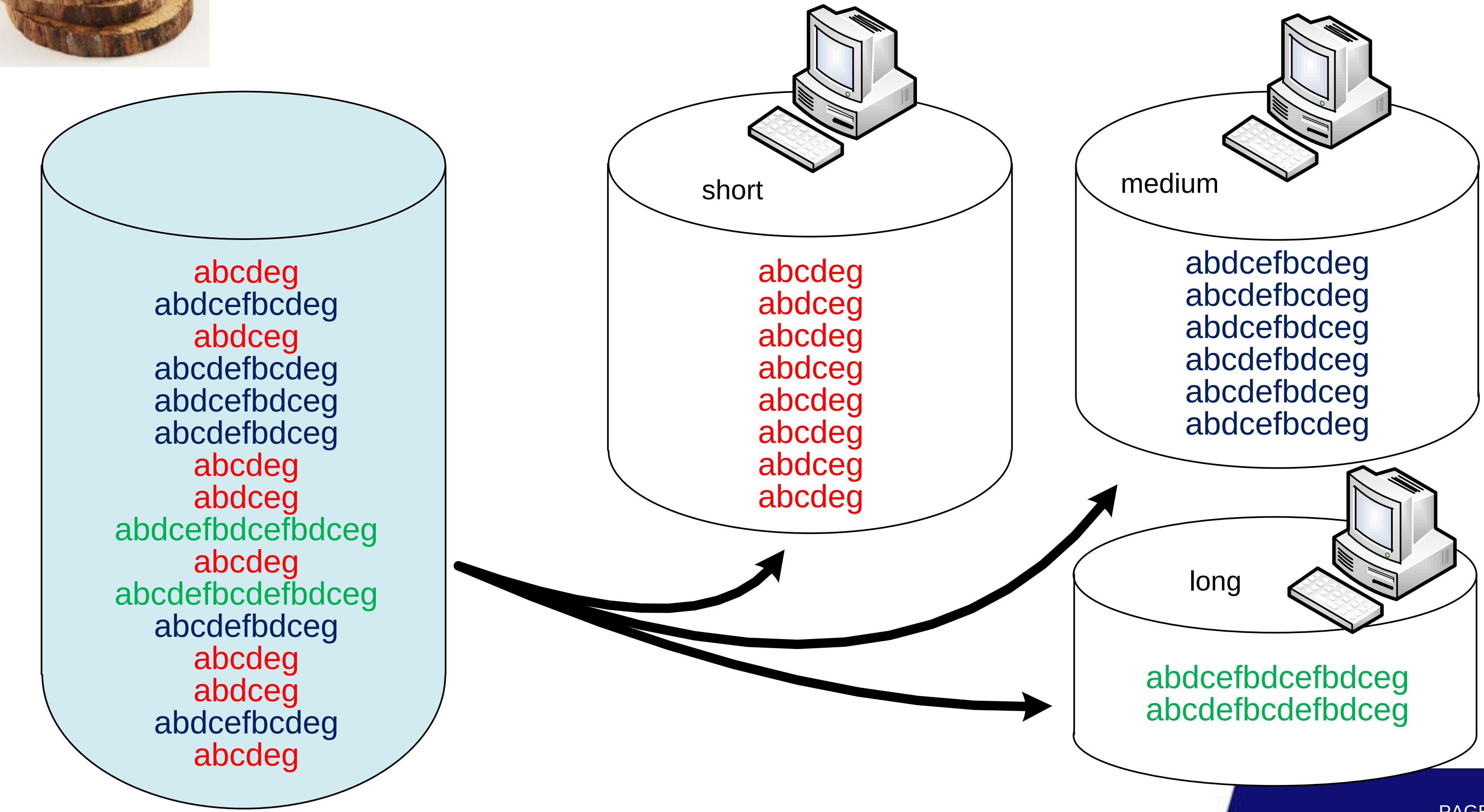


abcdeg
abdcefbcdeg
abdceg
abcdefbcdeg
abdcefbdceg
abcdefbdceg
abcdeg
abdceg
abdcefbdcefbdceg
abcdeg
abcdefbcdefbdceg
abcdefbdceg
abcdeg
abdceg
abdcefbcdeg
abcdeg

abcdeg
abdcefbcdeg
abdceg
abcdefbcdeg
abdcefbdceg
abcdefbdceg
abcdeg
abdceg

abdcefbdcefbdceg
abcdeg
abcdefbcdefbdceg
abcdefbdceg
abcdeg
abdceg
abdcefbcdeg
abcdeg

Vertical distribution II:
Split cases based on a specific feature



abcdeg
abdcefbcdeg
abdceg
abcdefbcdeg
abdcefbdceg
abcdefbdceg
abcdeg
abdceg
abdcefbdcefbdceg
abcdeg
abcdefbcdefbdceg
abcdefbdceg
abcdeg
abdceg
abdcefbcdeg
abcdeg

short

abcdeg
abdceg
abcdeg
abdceg
abcdeg
abcdeg
abdceg
abcdeg

medium

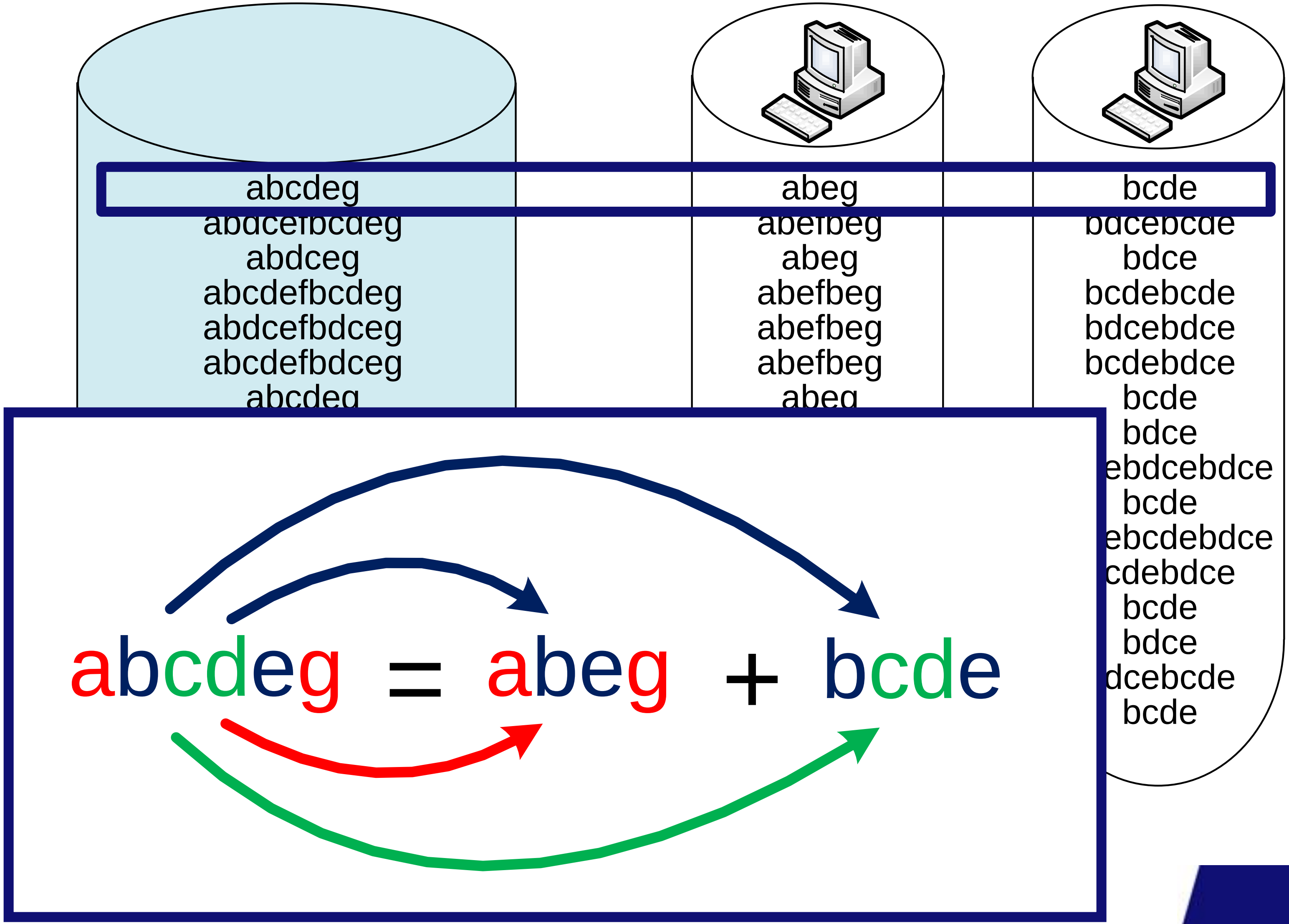
abdcefbcdeg
abcdefbcdeg
abdcefbdceg
abcdefbdceg
abcdefbdceg
abdcefbcdeg

long

abdcefbdcefbdceg
abcdefbcdefbdceg

Horizontal distribution

sets of activities



abcdeg	abeg	bcde
abdcefbcdeg	abefbeg	bdcebcde
abdceg	abeg	bdce
abcdefbcdeg	abefbeg	bcdebcde
abdcefbdceg	abefbeg	bdcebdce
abcdefbdceg	abefbeg	bcdebdce
abcdeg	abeg	bcde
		bdce
		ebdcebdce
		bcde
		ebcdebdce
		cdebdce
		bcde
		bdce
		dcebcde
		bcde

abcdeg = abeg + bcde

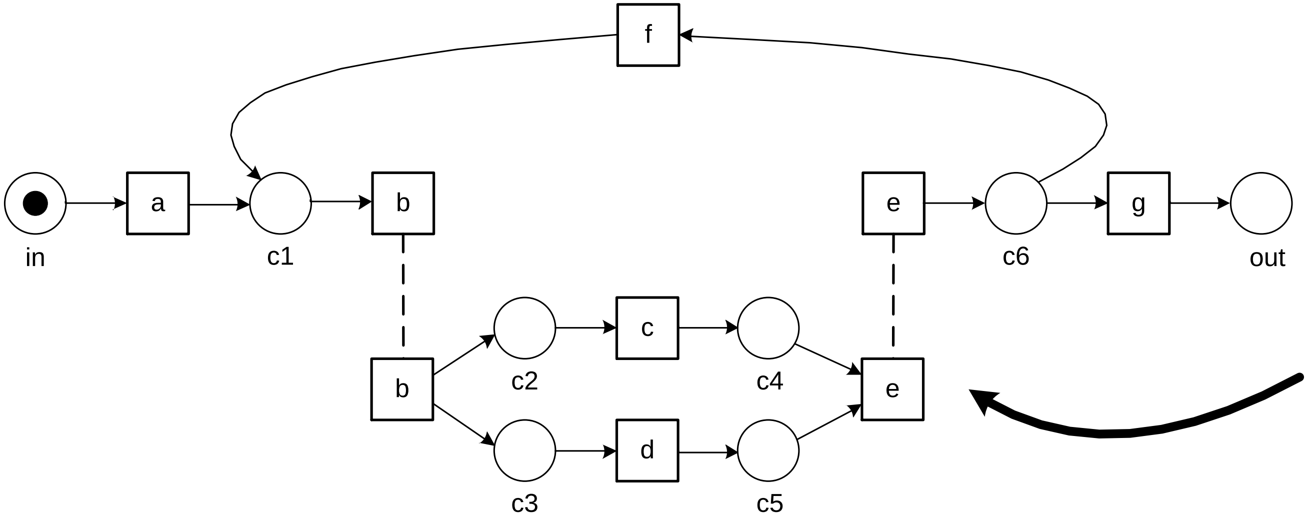
Horizontal distribution: The key idea

projected on a,b,e,f,g

abeg
abefbeg
abeg
abefbeg
abefbeg
abefbeg
abeg
abeg
abefbefbeg
abeg
abefbefbeg
abefbeg
abeg
abeg
abefbeg
abeg

projected on b,c,d,e

bcde
bdcebcde
bdce
bcdebcde
bdcebdce
bcdebdce
bcde
bdce
bdcebdcebdce
bcde
bcdebcdebdce
bcdebdce
bcde
bdce
bdcebcde
bcde



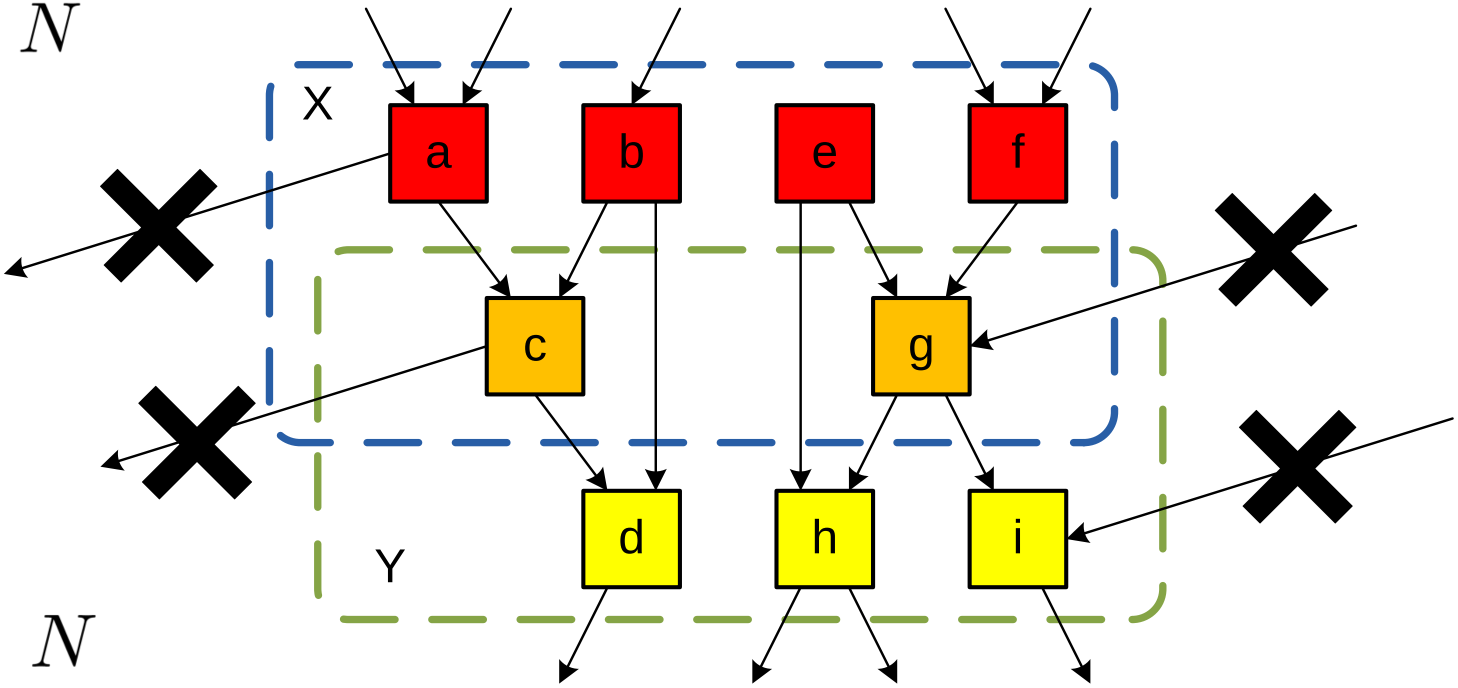
Passages

Moscow GUM

Passage *P=(X,Y)*

causal dependency:
may trigger or enable

$$\emptyset \neq X \subseteq N$$

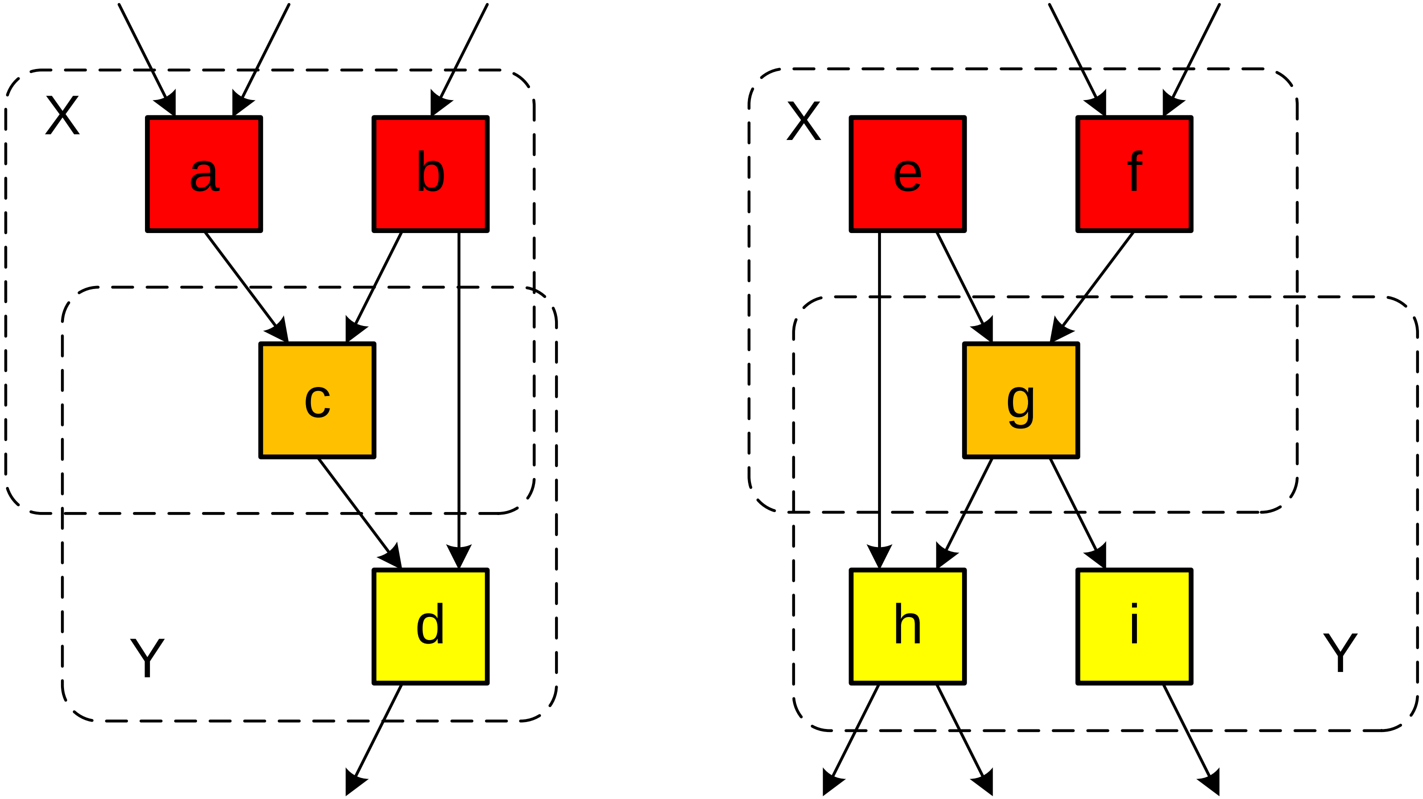


$$\emptyset \neq Y \subseteq N$$

$$\boxed{X \overset{G}{\bullet}= Y}$$

$$\boxed{X =\overset{G}{\bullet} Y}$$

Minimal passages

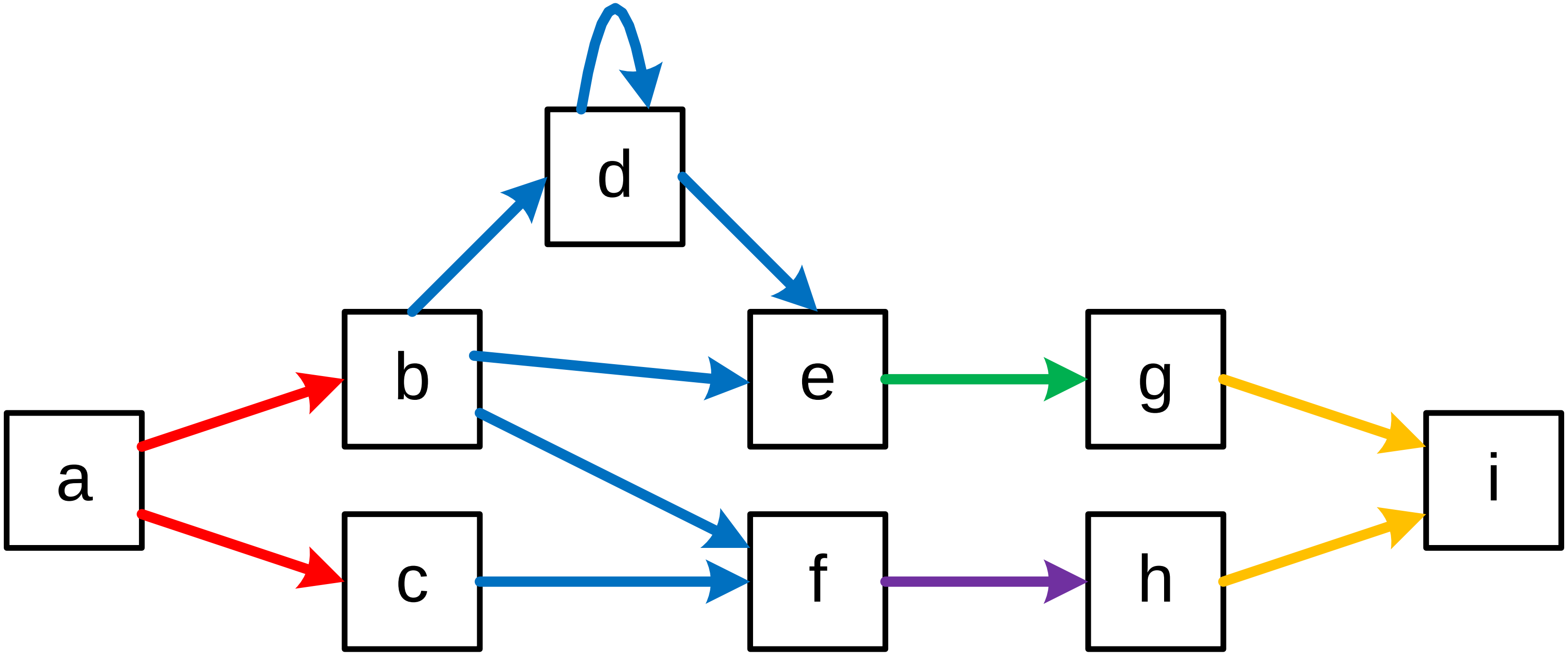


$$X \overset{G}{\bullet}= Y$$

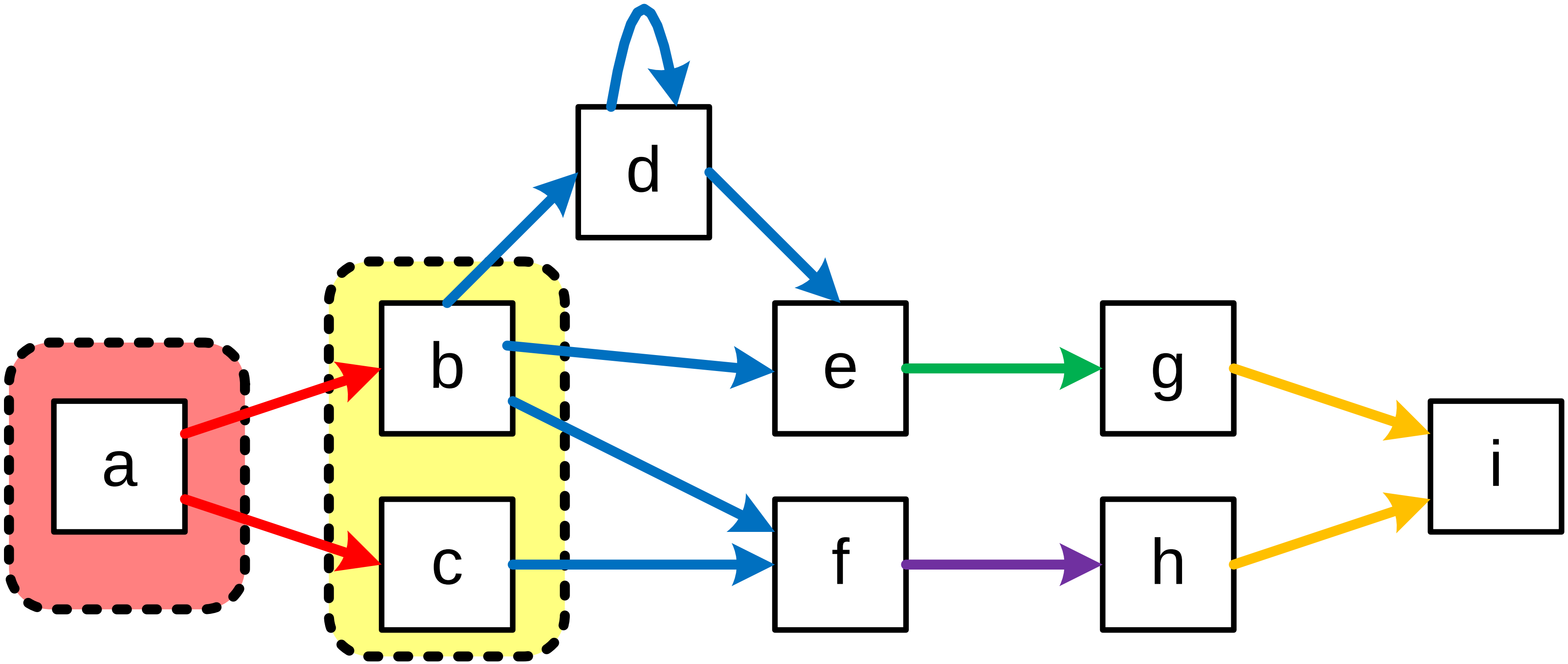
$$X =\overset{G}{\bullet} Y$$

a passage is minimal if it does not contain smaller passages

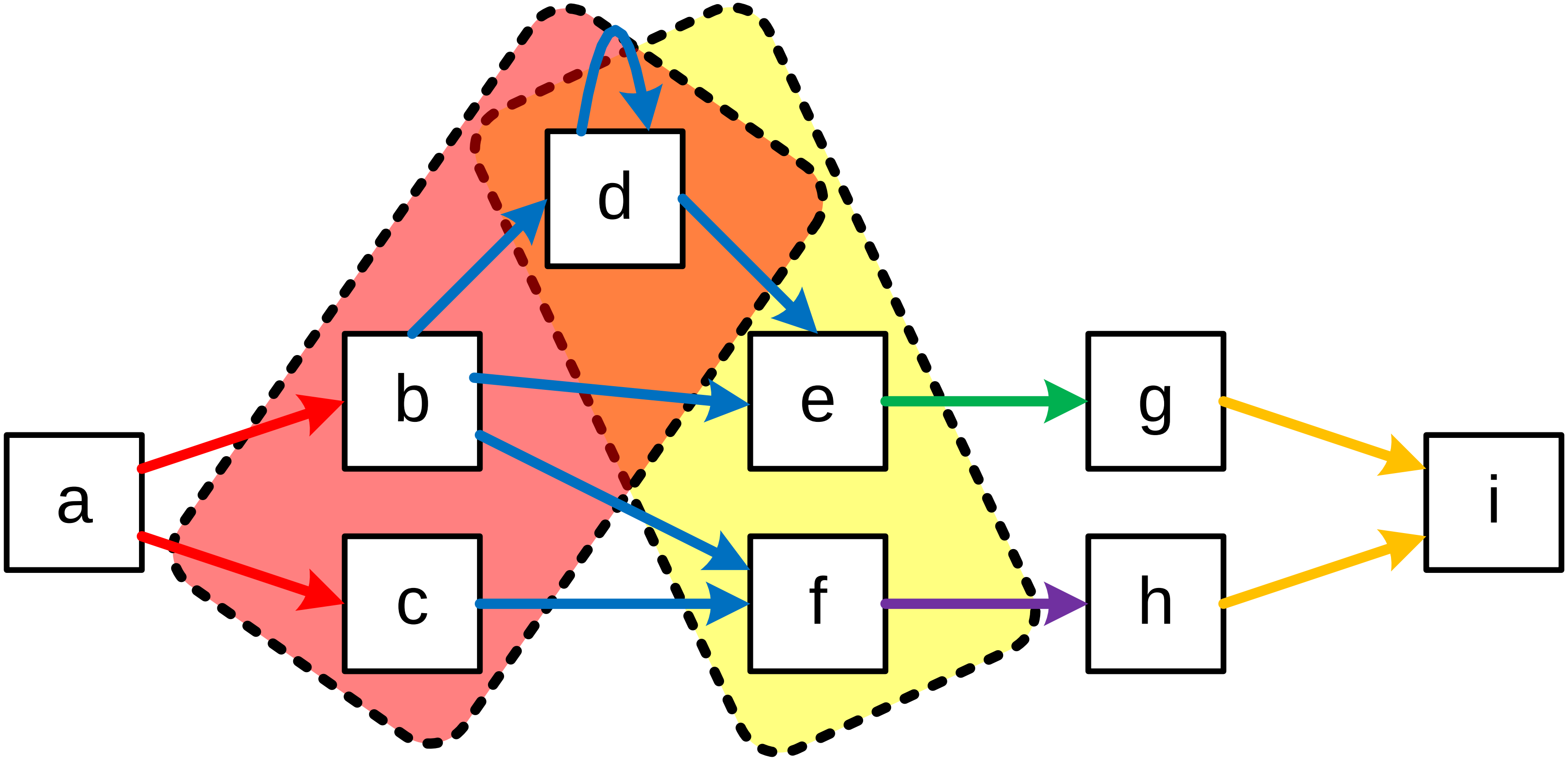
Passages define an equivalence relation on the edges in the graph



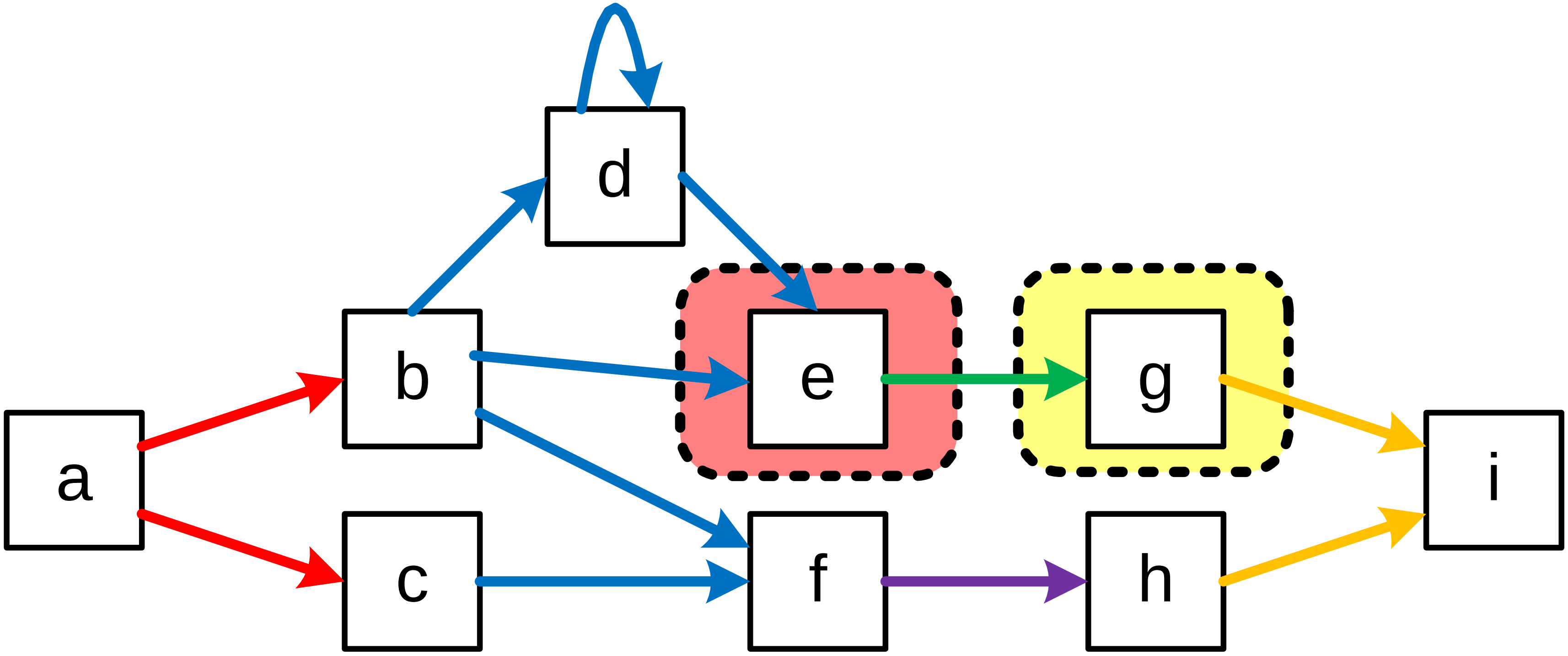
Minimal passage 1: ({a},{b,c})



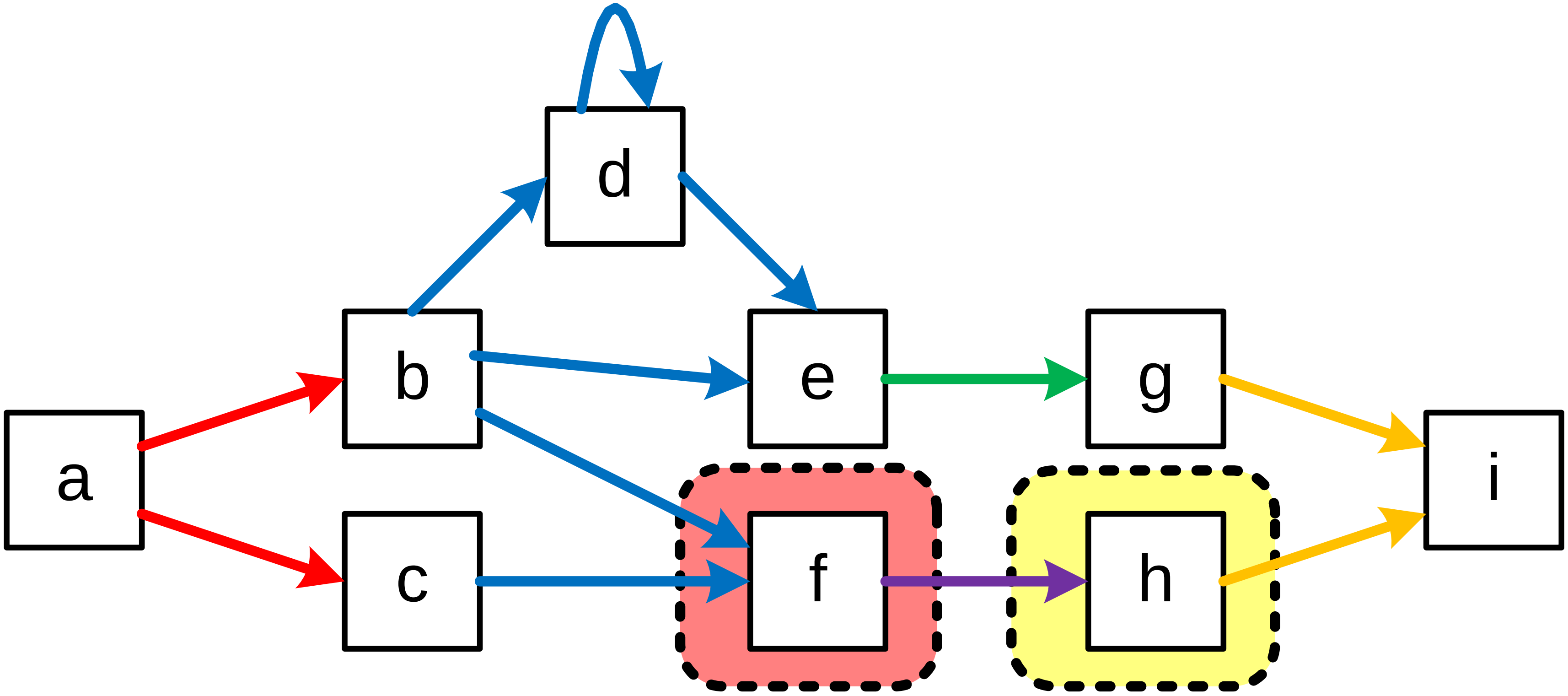
Minimal passage 2: ({b,c,d},{d,e,f})



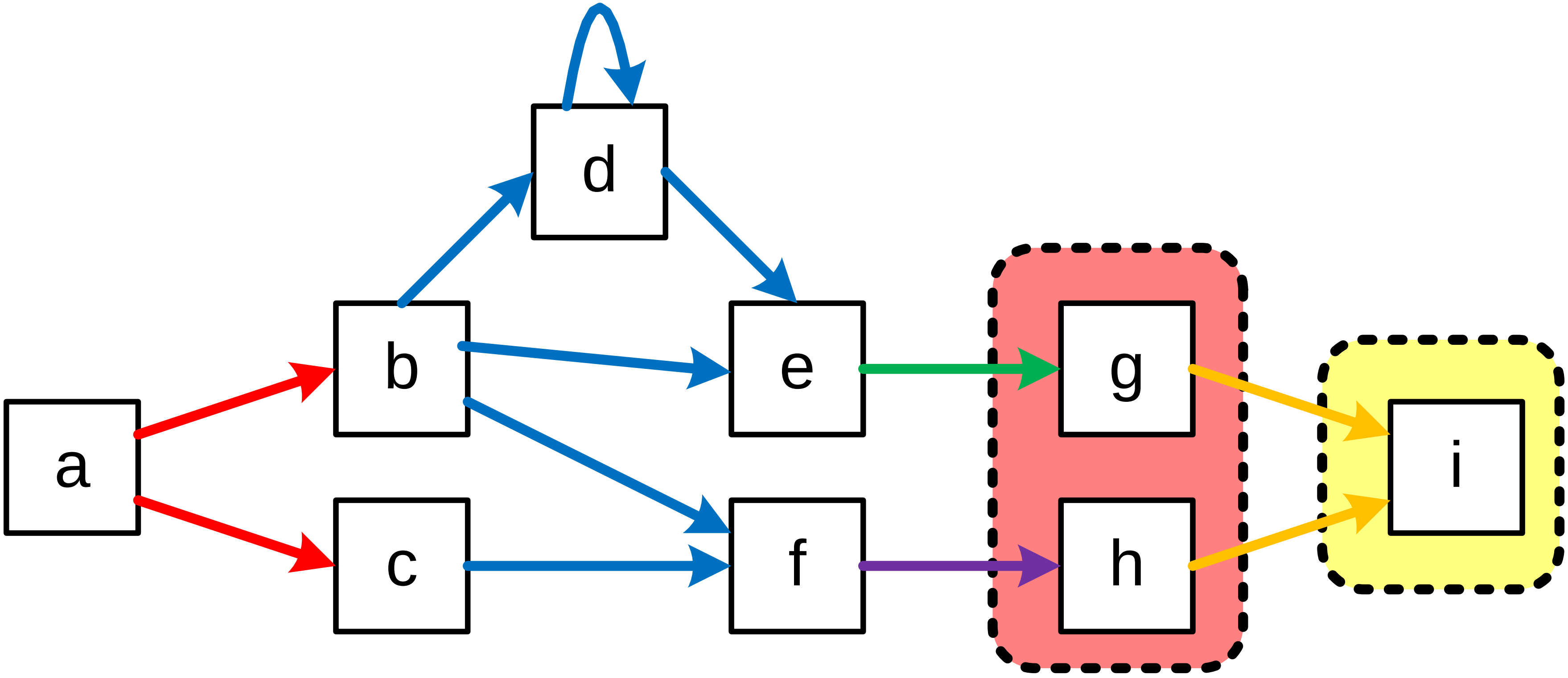
Minimal passage 3: ({e},{g})



Minimal passage 4: ({f},{h})

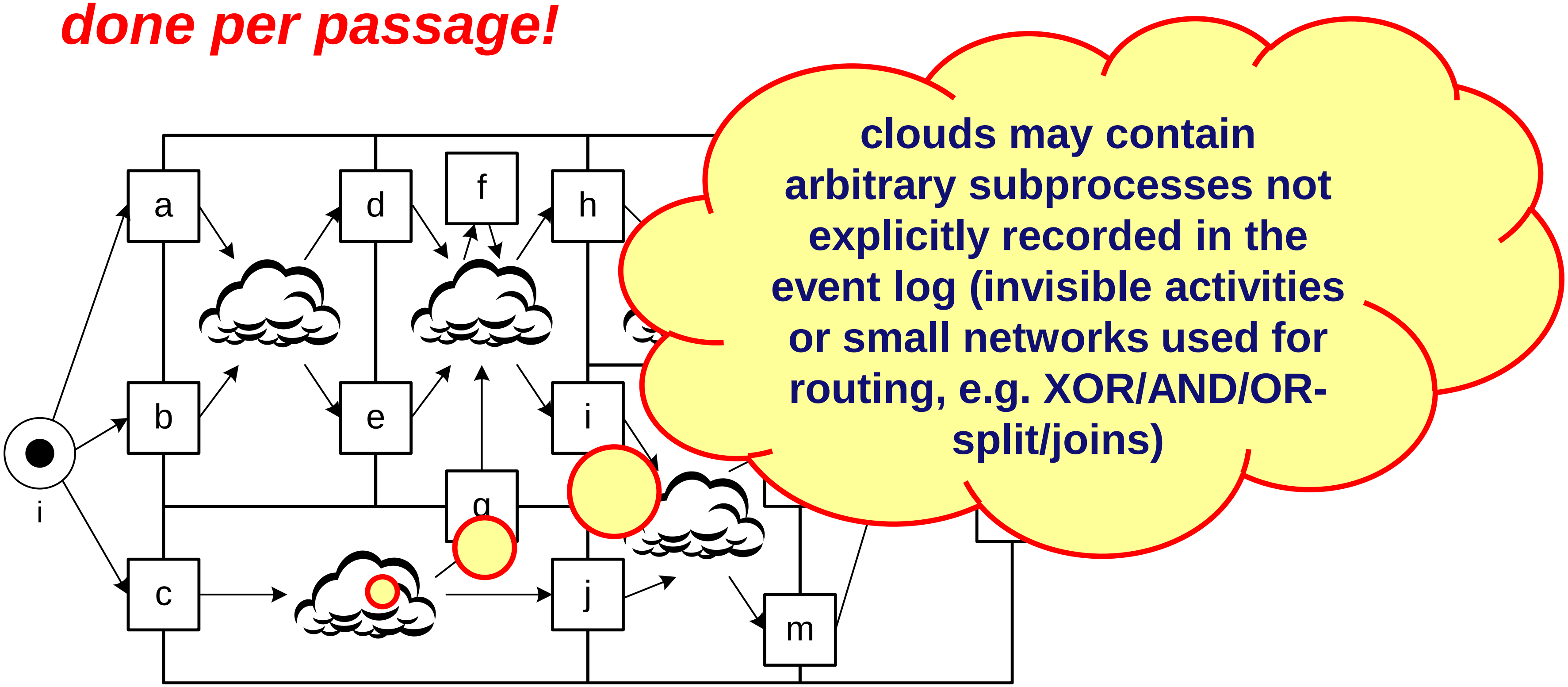


Minimal passage 5: ({g,h},{i})



So What?

- **Any process model can be partitioned in minimal passages.**
- **Claim: *Discovery and conformance checking can be done per passage!***



clouds may contain arbitrary subprocesses not explicitly recorded in the event log (invisible activities or small networks used for routing, e.g. XOR/AND/OR-split/joins)

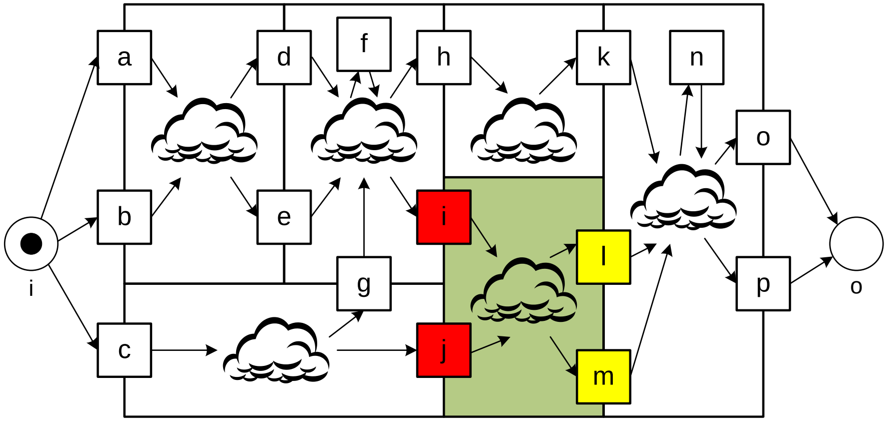
Example result for Petri nets

Theorem 1 (Main Theorem). *Let $L \in \mathcal{B}(A^*)$ be an event log and let $WF = (PN, in, T_i, out, T_o)$ be a WF-net with $PN = (P, T, F, T_v)$.*

L is perfectly fitting system net $SN = (PN, [in], [out])$ if and only if

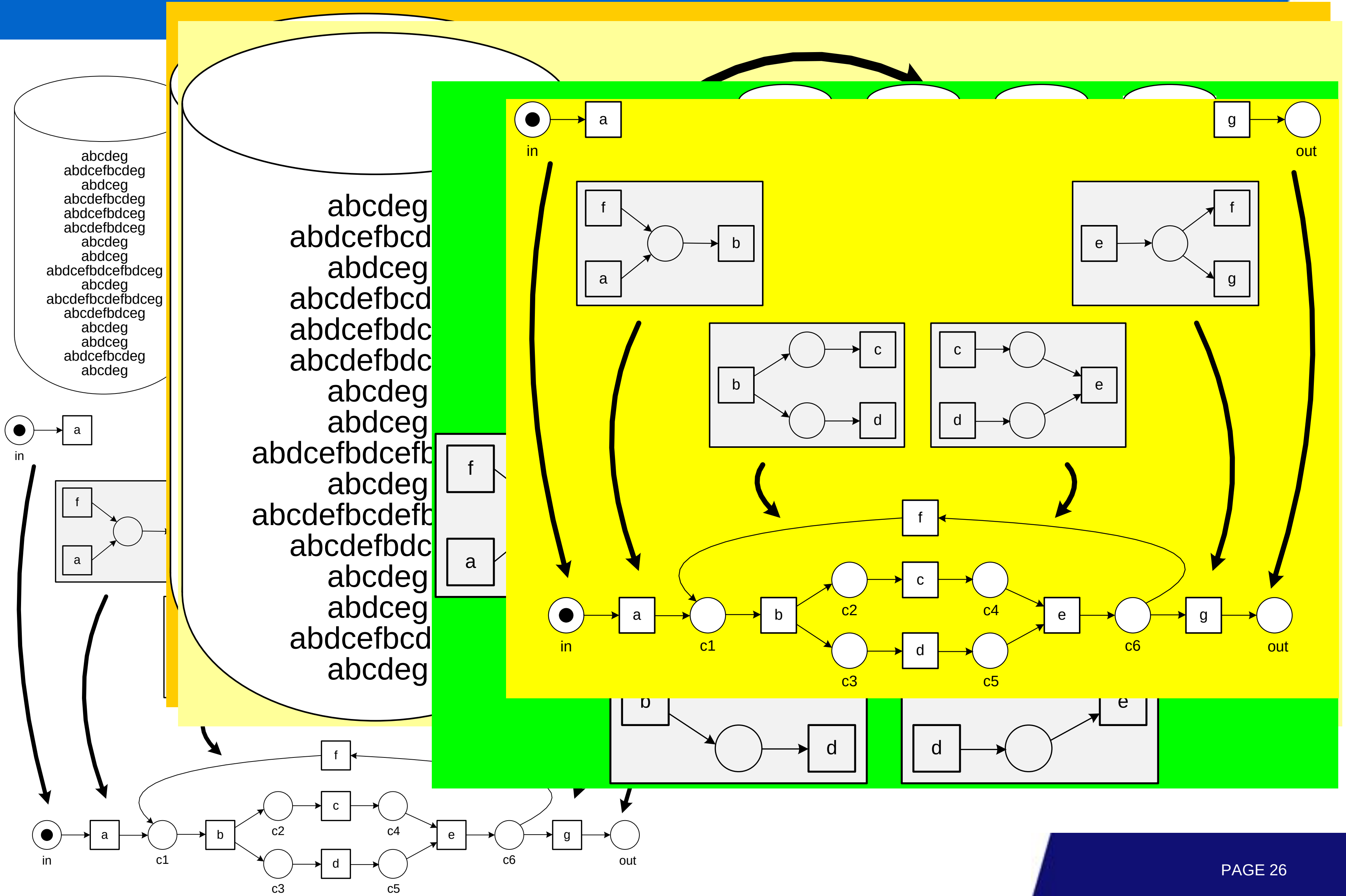
- *for any $\langle a_1, a_2, \ldots a_k \rangle \in L$: $a_1 \in T_i$ and $a_k \in T_o$, and*
- *for any $(X, Y) \in pas_{min}(skel(PN))$: $L \restriction_{X \cup Y}$ is perfectly fitting $SN^{(X,Y)} = (PN^{(X,Y)}, [\,], [\,])$.*

“The event log fits all passages if and only if the event log fits the whole model.”

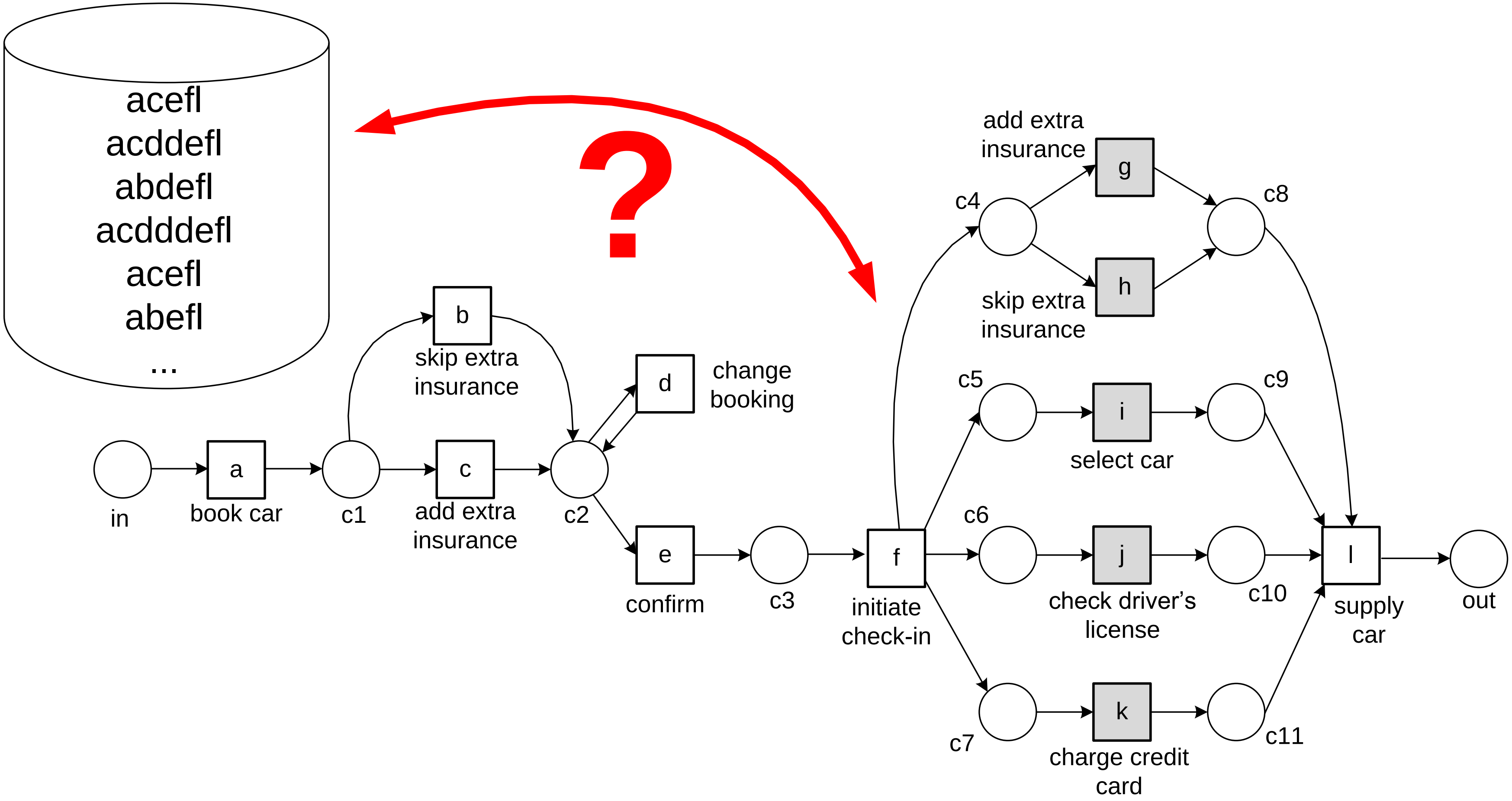


Key insight: interface transitions controlled by event log

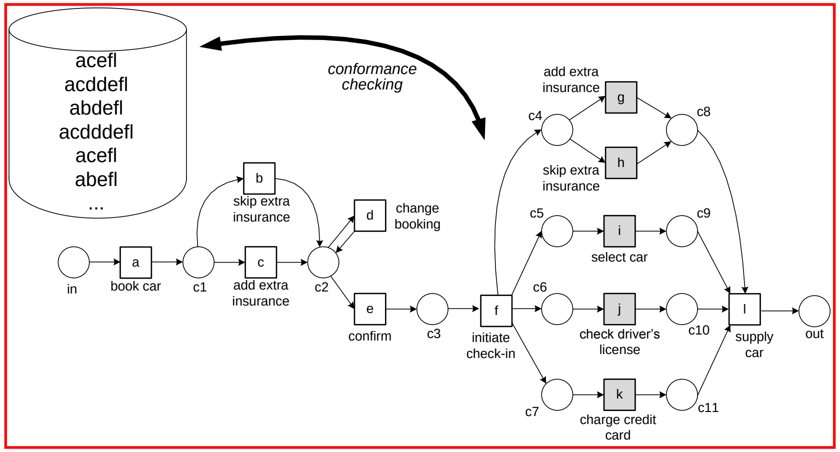
Discovery example

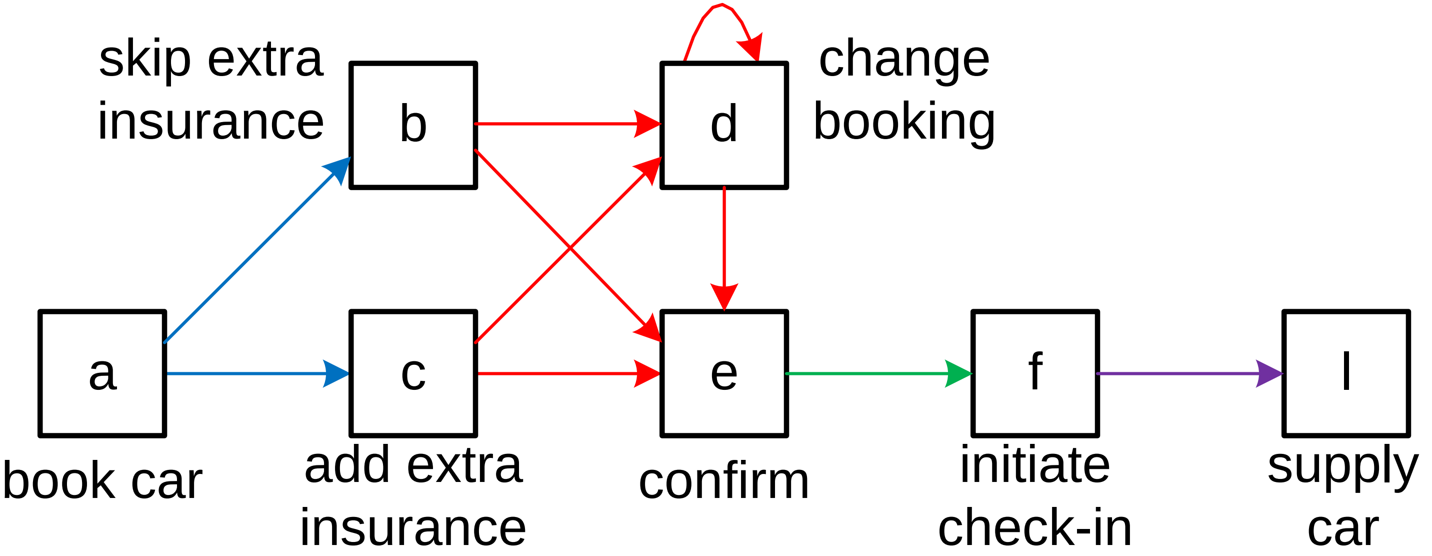


Conformance checking

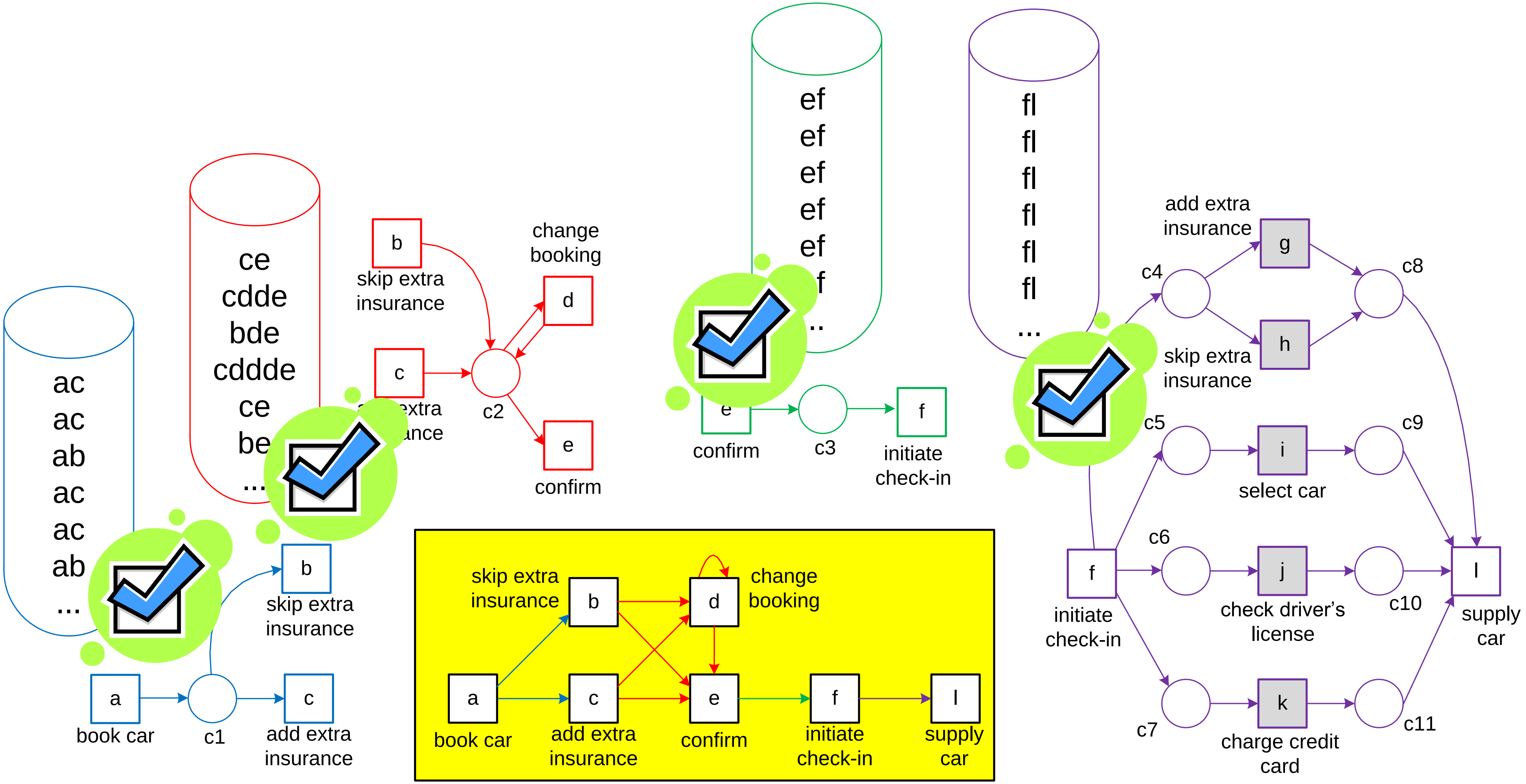


Create Skeleton





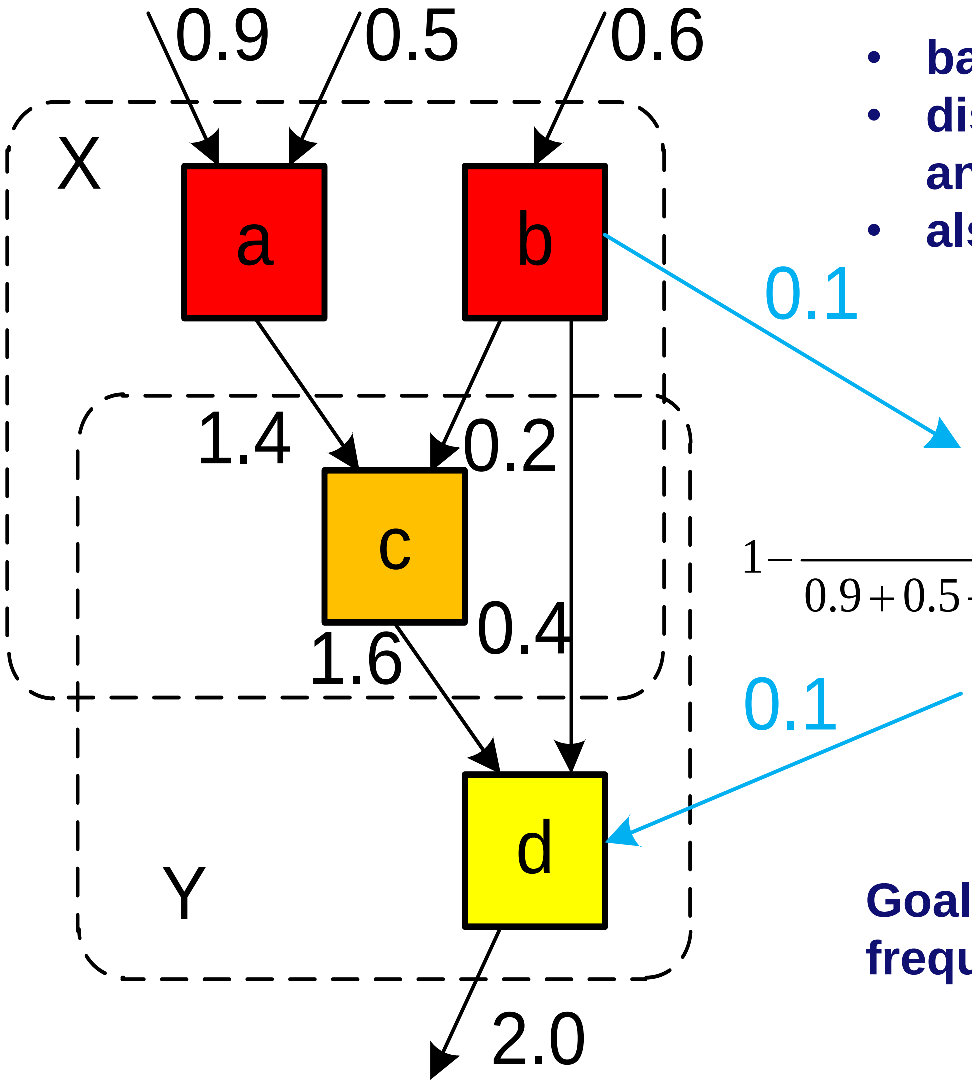
Net fragments per passage



Limitations

- Need to discover causal dependencies first (only issue for discovery, use fuzzy/heuristic rules).
- “Interface transitions” need to have a unique label.
- Minimal passages may be large in dense graphs.

“Almost passages”

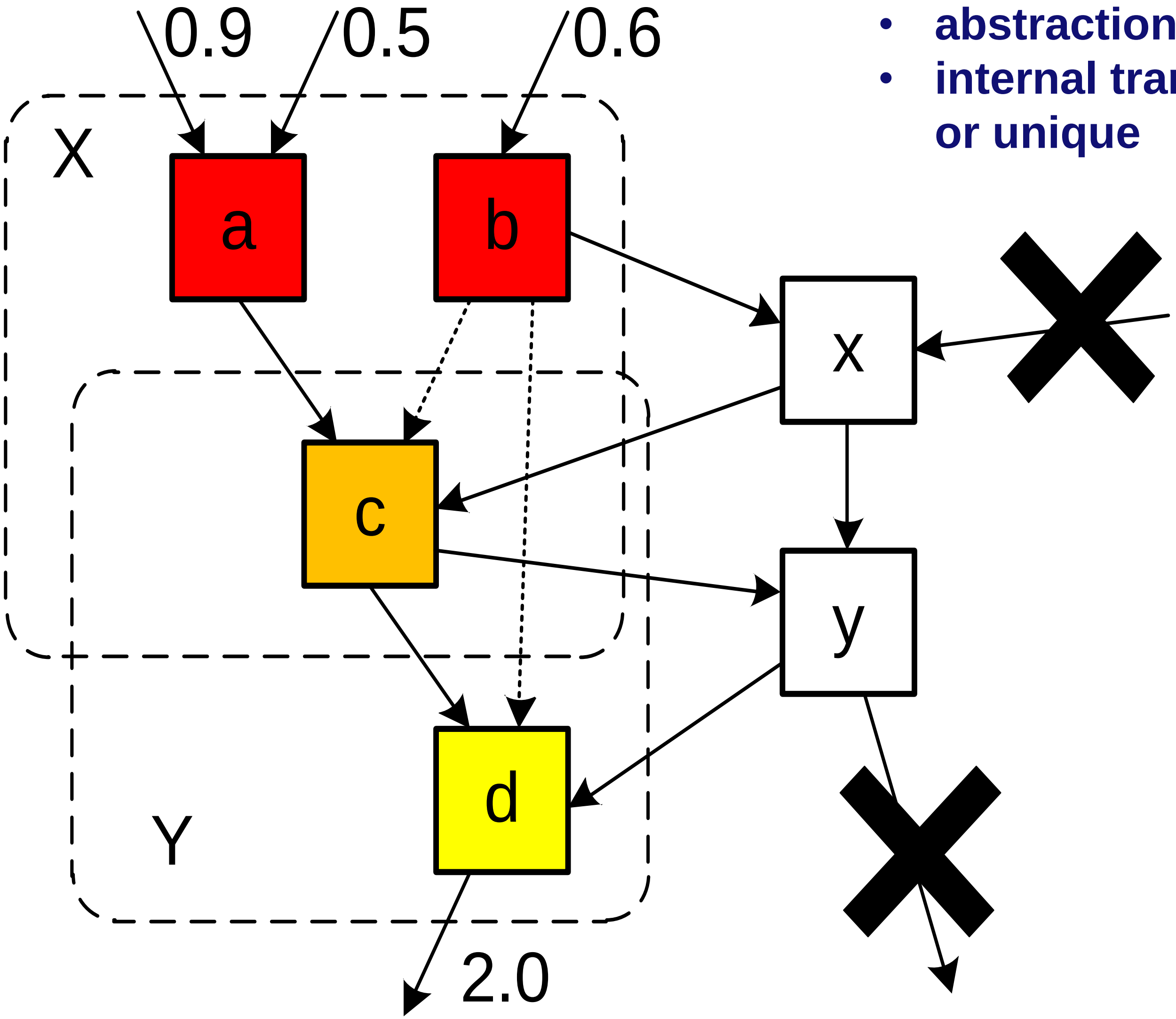


- **balance between size and quality**
- **discard same arcs in all passages (use an iterative procedure)**
- **also adding arcs? (don't think so)**

$$1-\frac{0.1+0.1}{0.9+0.5+0.6+0.1+1.4+0.2+1.6+0.4+0.1+2.0}$$

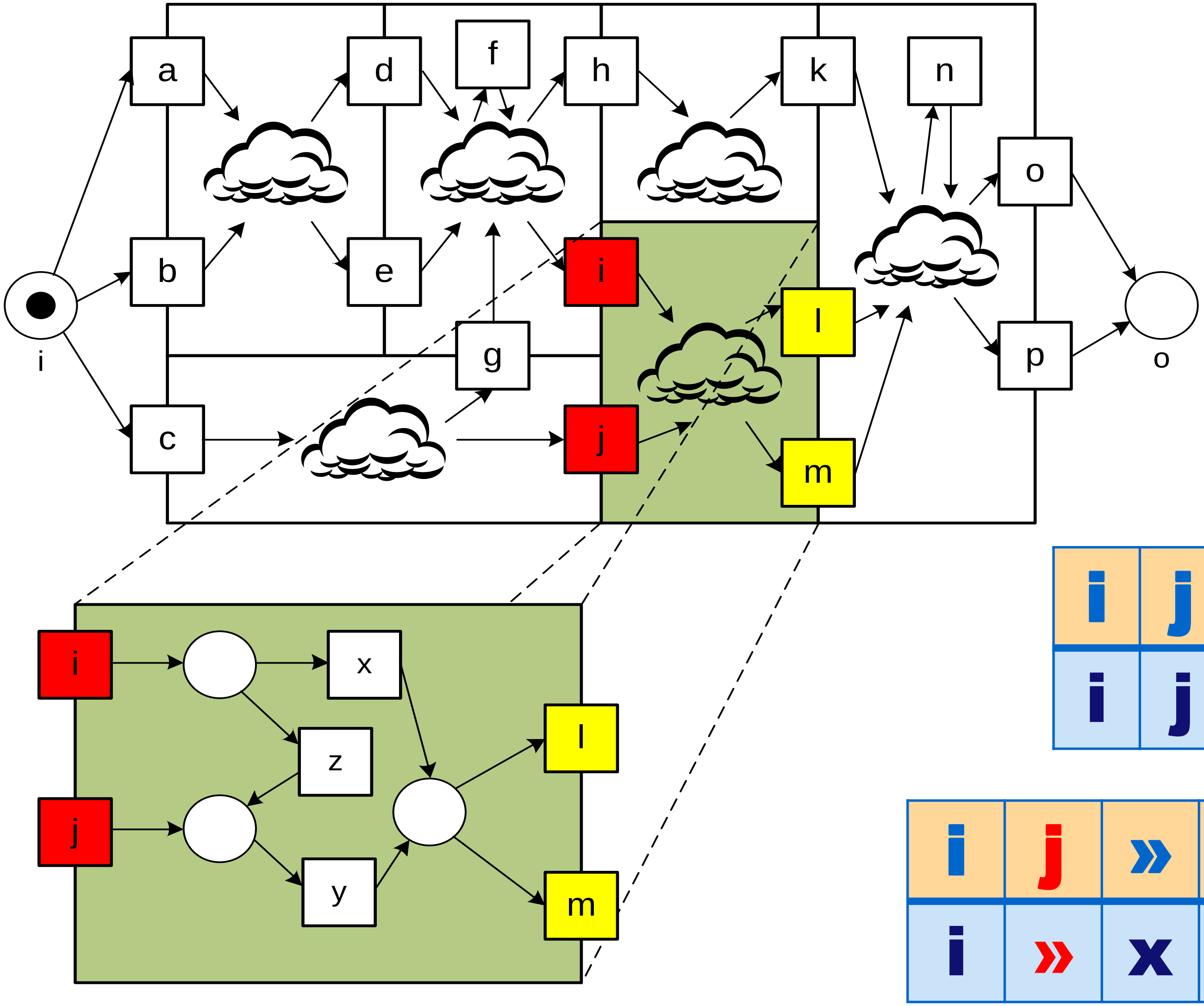
Goal: small passages with only low-frequent arcs violating the rule.

Extended passages



- abstraction yields passage
- internal transitions may be silent or unique

Conformance checking (with silent steps inside passages)

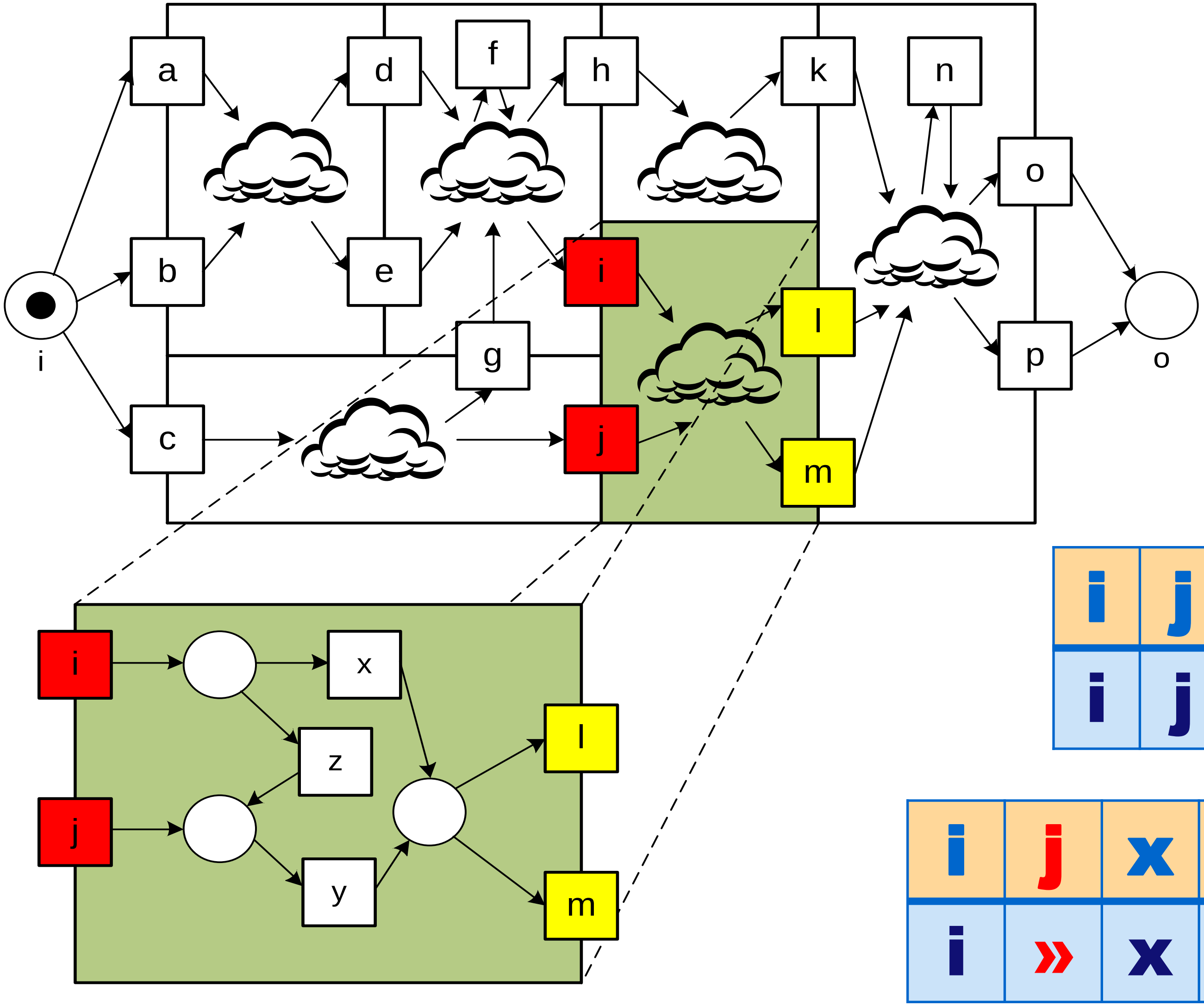


i	»	l
i	x	l

i	j	»	»	l	»
i	j	x	y	l	m

i	j	»	l
i	»	x	l

Internal steps may be visible as long as they correspond to only one passage

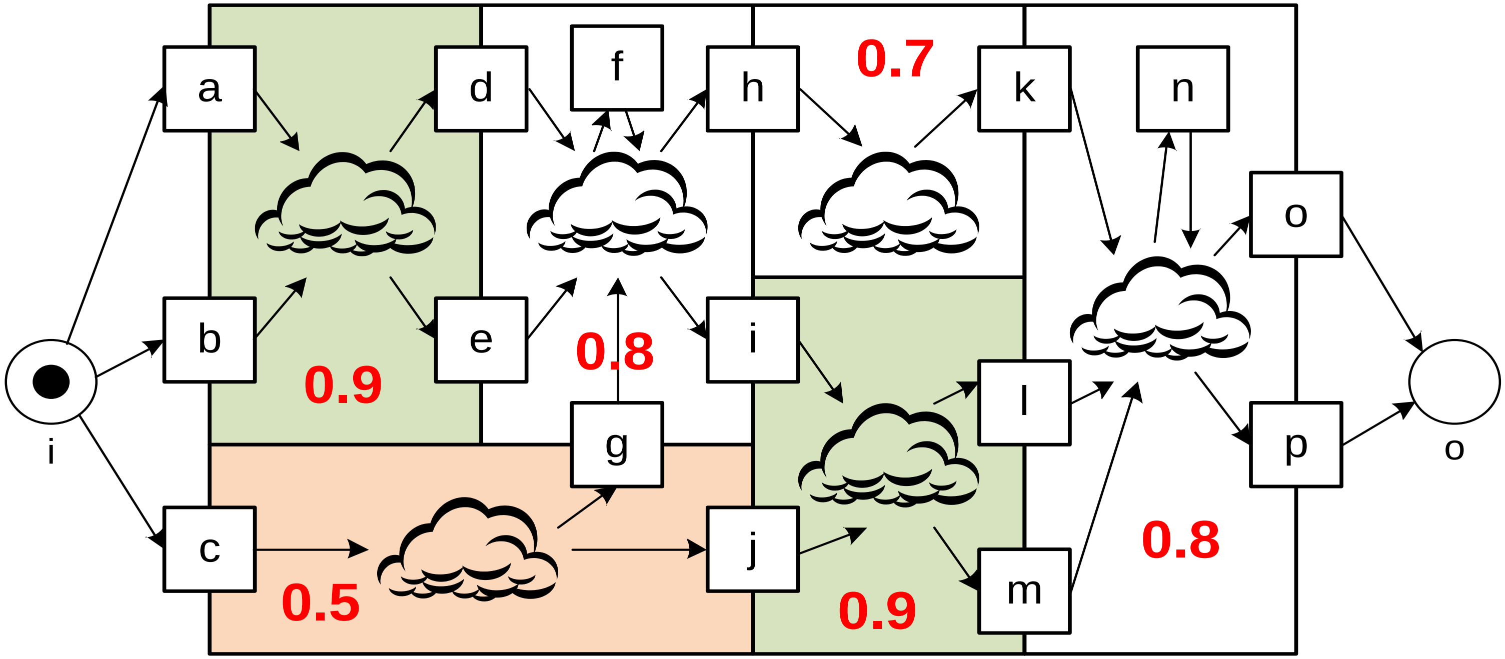


i	x	l
i	x	l

i	j	x	y	l	»
i	j	x	y	l	m

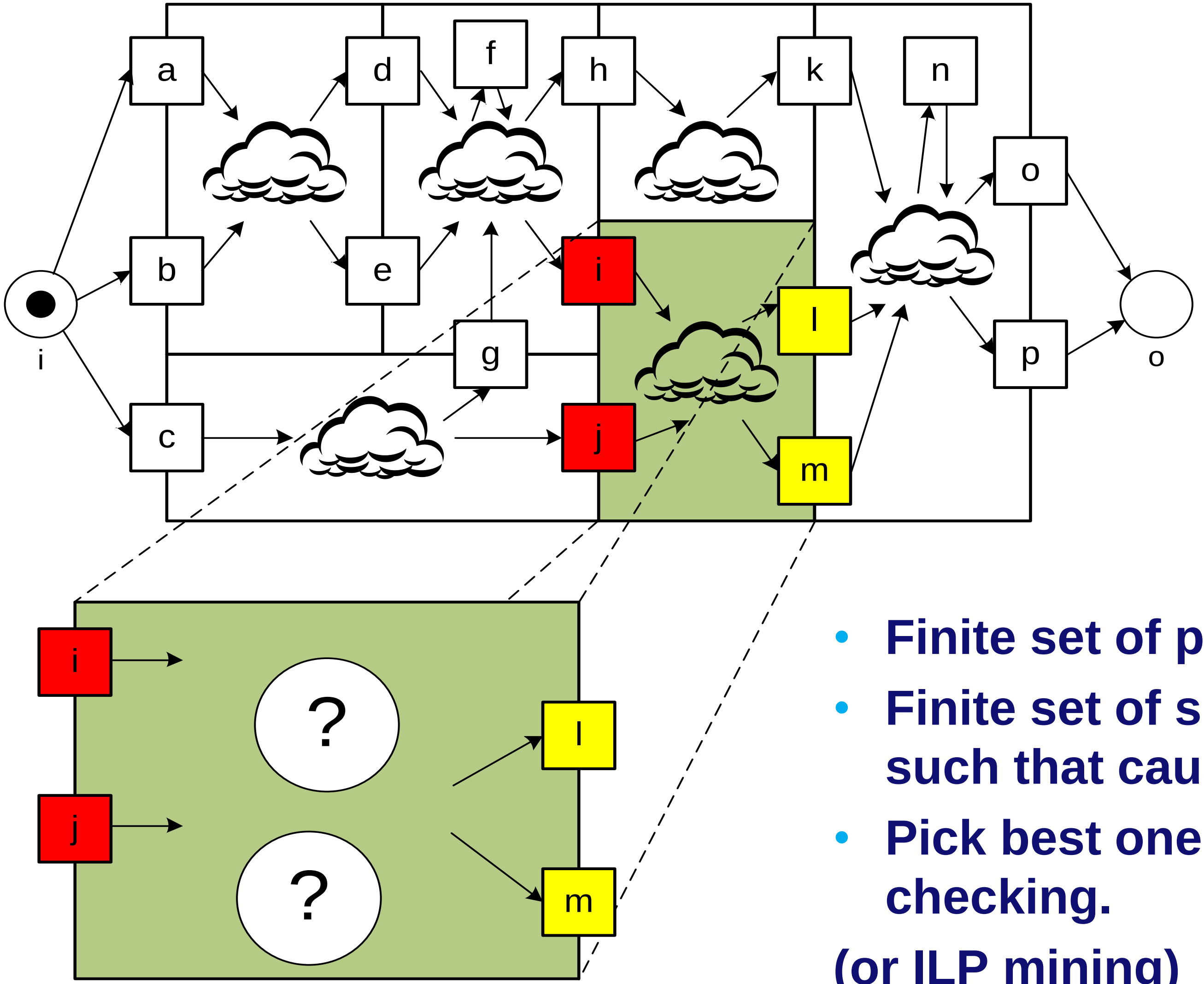
i	j	x	l
i	»	x	l

Aggregating conformance metrics



- **For all four conformance dimensions (fitness, simplicity, generalization, and precision)?**
- **Overall metrics: aggregated local values should be close to global values (e.g., computed fitness value is a lower bound).**
- **Local diagnostics (problem spots).**

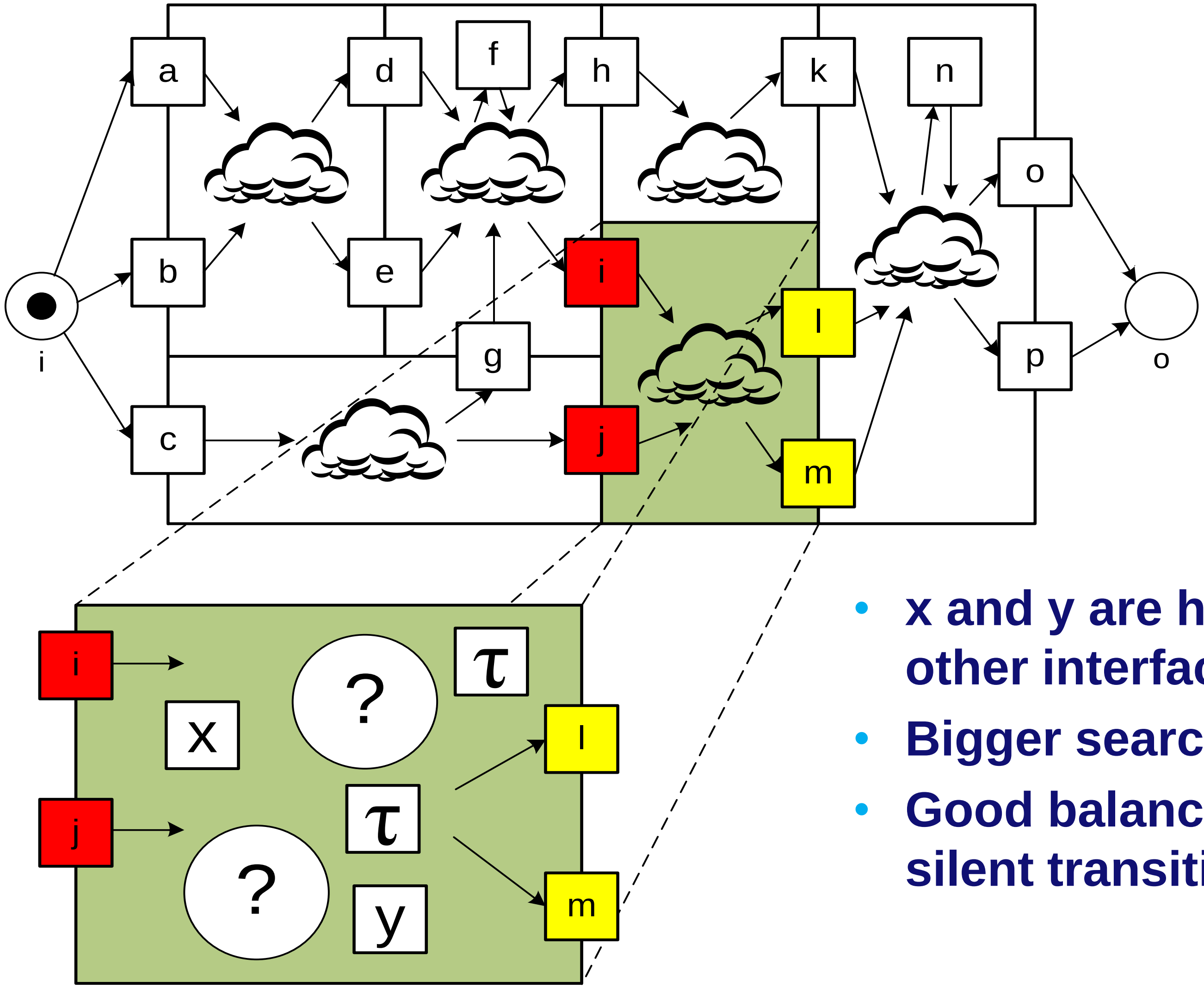
Discovery (no silent/internal transitions)



- Finite set of possible places.
- Finite set of subsets of places such that causalities hold.
- Pick best one using conformance checking.

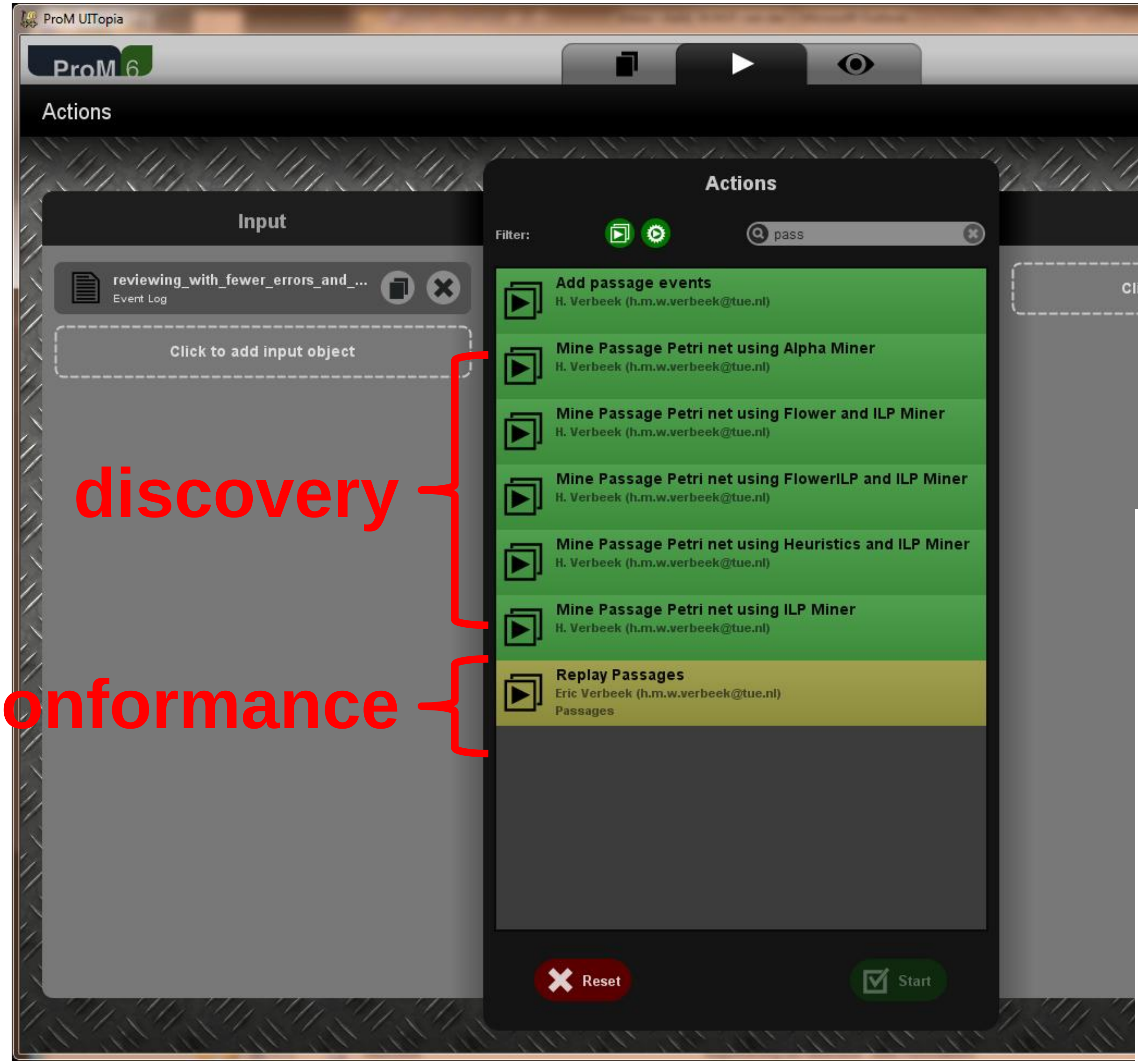
(or ILP mining)

Discovery (with silent or internal transitions)



- x and y are handled like the other interface transitions.
- Bigger search space.
- Good balance: one layer of silent transitions.

Tool Support in ProM
(implemented by Eric Verbeek)



ProM UITopia

ProM 6

Actions

Input

reviewing_with_fewer_errors_and_...
Event Log

Click to add input object

Actions

Filter: pass

- Add passage events
  H. Verbeek (h.m.w.verbeek@tue.nl)
- Mine Passage Petri net using Alpha Miner
  H. Verbeek (h.m.w.verbeek@tue.nl)
- Mine Passage Petri net using Flower and ILP Miner
  H. Verbeek (h.m.w.verbeek@tue.nl)
- Mine Passage Petri net using FlowerILP and ILP Miner
  H. Verbeek (h.m.w.verbeek@tue.nl)
- Mine Passage Petri net using Heuristics and ILP Miner
  H. Verbeek (h.m.w.verbeek@tue.nl)
- Mine Passage Petri net using ILP Miner
  H. Verbeek (h.m.w.verbeek@tue.nl)
- Replay Passages
  Eric Verbeek (h.m.w.verbeek@tue.nl)
  Passages

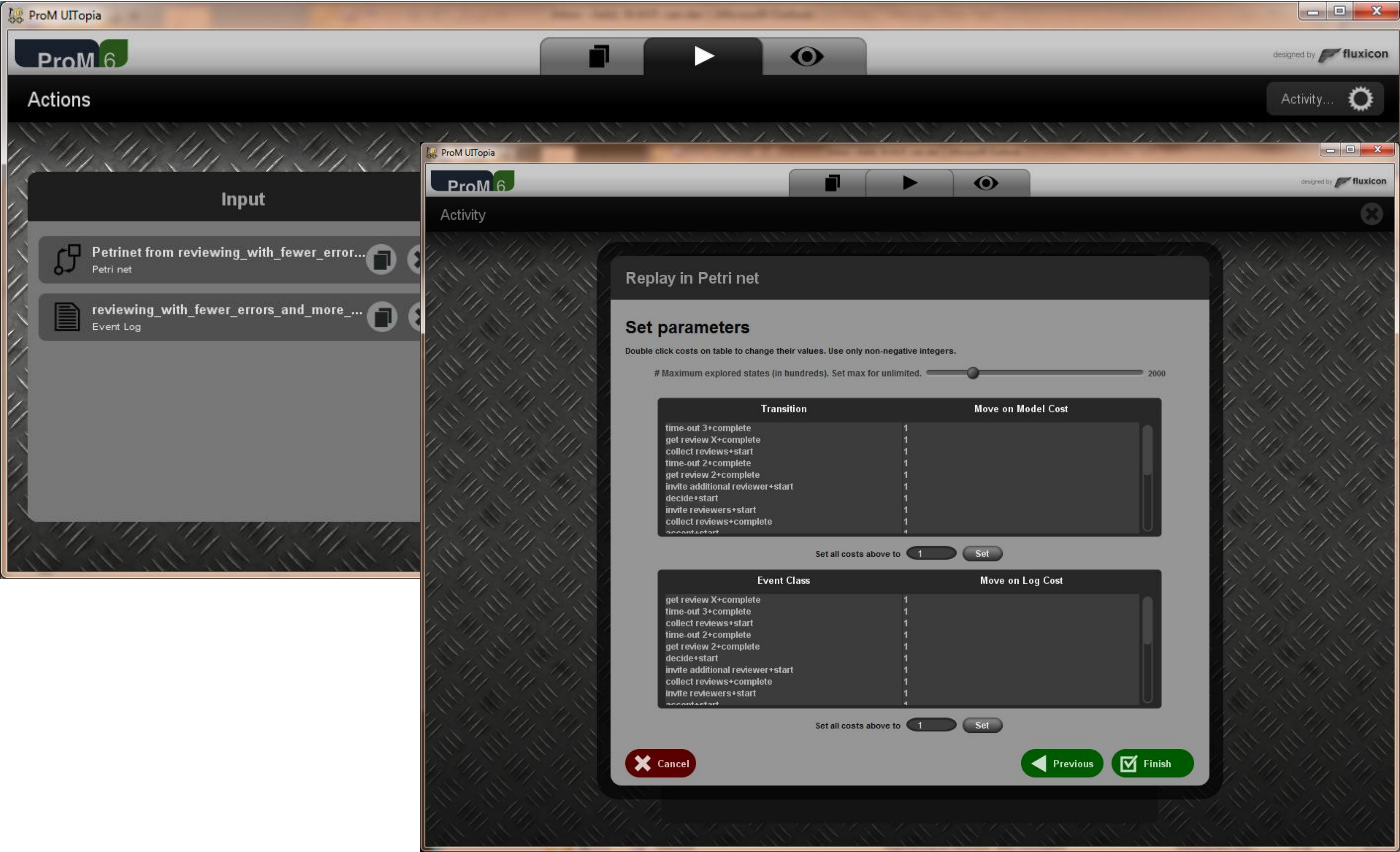
Output

Click to add output object

discovery

conformance

Passage-Based Conformance Checking



Petrinet from reviewing_with_fewer_error...
Petri net

reviewing_with_fewer_errors_and_more_...
Event Log

Replay in Petri net

Set parameters

Double click costs on table to change their values. Use only non-negative integers.

Maximum explored states (in hundreds). Set max for unlimited. 2000

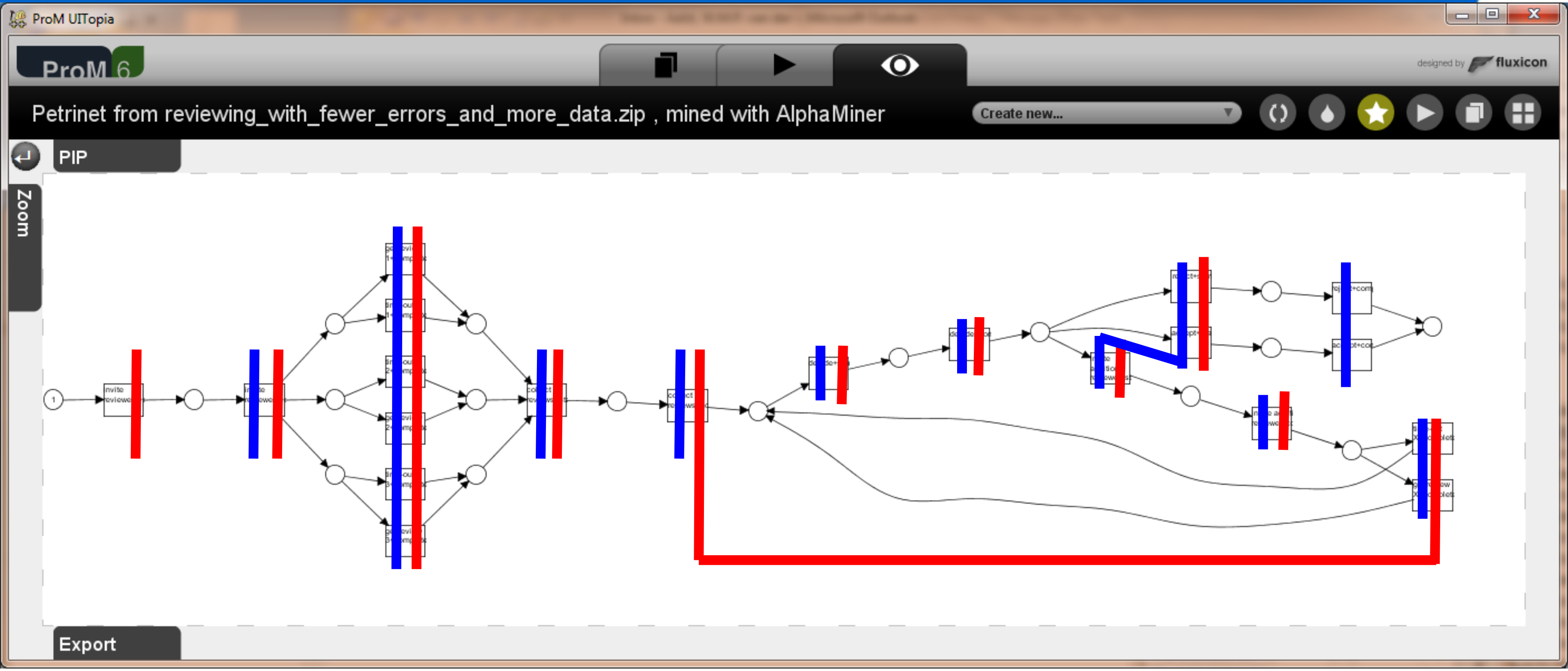
Transition	Move on Model Cost
time-out 3+complete	1
get review X+complete	1
collect reviews+start	1
time-out 2+complete	1
get review 2+complete	1
invite additional reviewer+start	1
decide+start	1
invite reviewers+start	1
collect reviews+complete	1
accept+start	1

Set all costs above to 1 Set

Event Class	Move on Log Cost
get review X+complete	1
time-out 3+complete	1
collect reviews+start	1
time-out 2+complete	1
get review 2+complete	1
decide+start	1
invite additional reviewer+start	1
collect reviews+complete	1
invite reviewers+start	1
accept+start	1

Set all costs above to 1 Set

Process Model with 11 Passages

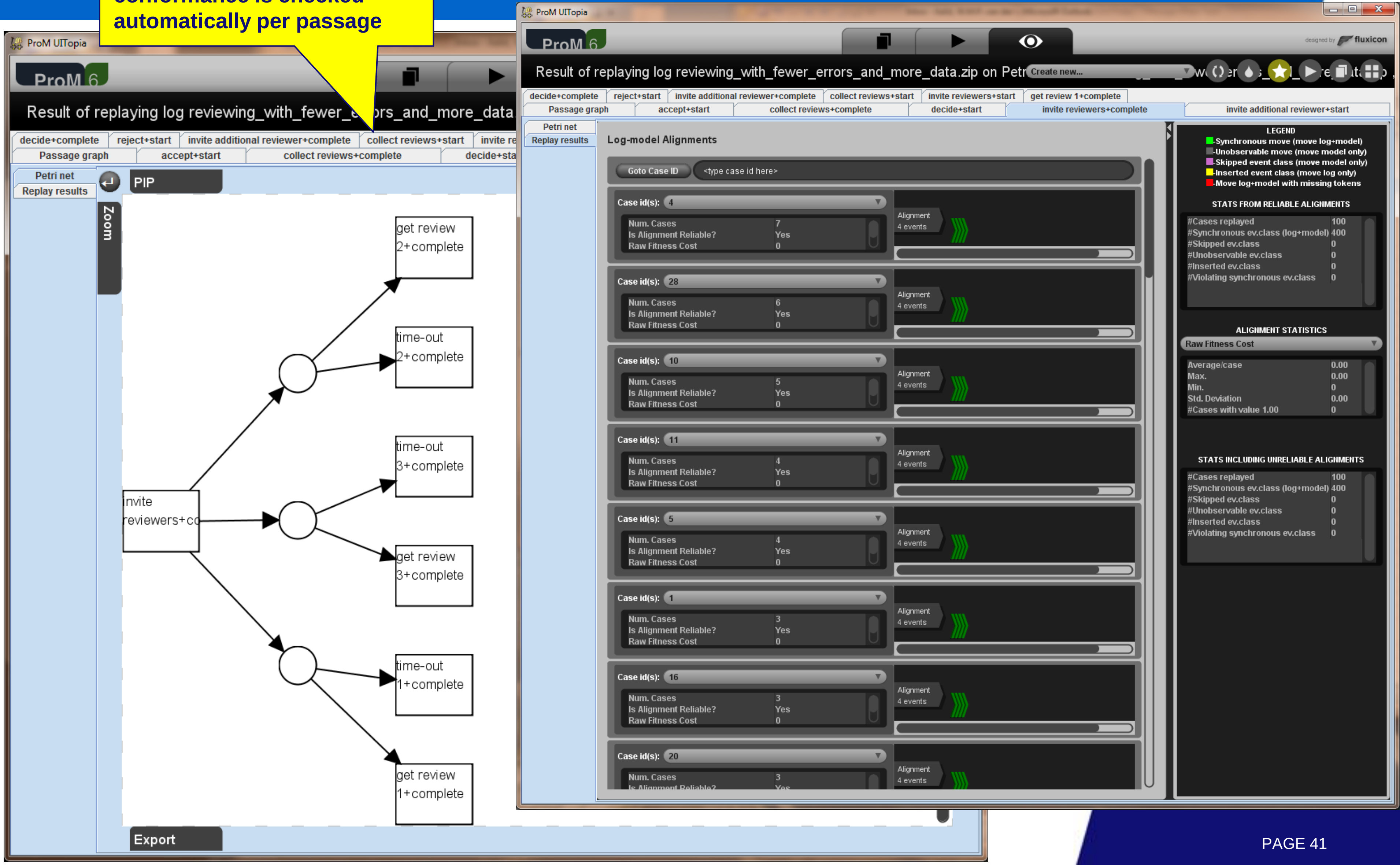


Petrinet from reviewing_with_fewer_errors_and_more_data.zip , mined with AlphaMiner

(X,Y)

Conformance Checking per Passage

conformance is checked automatically per passage



ProM UITopia

Result of replaying log reviewing_with_fewer_e…rs_and_more_data

decide+complete	reject+start	invite additional reviewer+complete	collect reviews+start	invite re
Passage graph	accept+start	collect reviews+complete	decide+sta	

Petri net

Replay results

PIP

Zoom

Export

ProM UITopia

ProM 6

designed by fluxicon

Result of replaying log reviewing_with_fewer_errors_and_more_data.zip on Petr

Create new...

decide+complete	reject+start	invite additional reviewer+complete	collect reviews+start	invite reviewers+start	get review 1+complete
Passage graph	accept+start	collect reviews+complete	decide+start	invite reviewers+complete	invite additional reviewer+start

Petri net

Replay results

Log-model Alignments

Goto Case ID <type case id here>

Case id(s)	Num. Cases	Is Alignment Reliable?	Raw Fitness Cost	Alignment
4	7	Yes	0	4 events
28	6	Yes	0	4 events
10	5	Yes	0	4 events
11	4	Yes	0	4 events
5	4	Yes	0	4 events
1	3	Yes	0	4 events
16	3	Yes	0	4 events
20	3	Yes		4 events

LEGEND

🟩-Synchronous move (move log+model)
⬛-Unobservable move (move model only)
🟪-Skipped event class (move model only)
🟨-Inserted event class (move log only)
🟥-Move log+model with missing tokens

STATS FROM RELIABLE ALIGNMENTS

#Cases replayed	100
#Synchronous ev.class (log+model)	400
#Skipped ev.class	0
#Unobservable ev.class	0
#Inserted ev.class	0
#Violating synchronous ev.class	0

ALIGNMENT STATISTICS

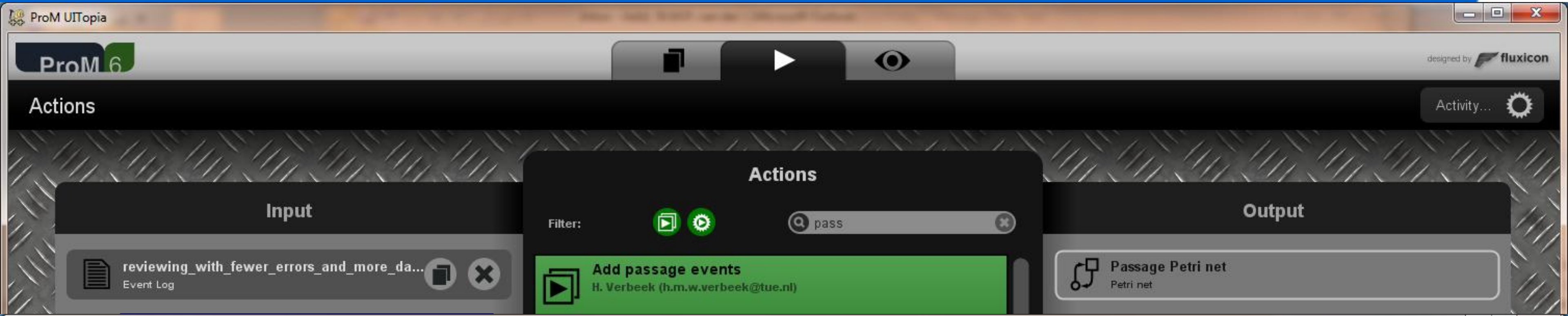
Raw Fitness Cost

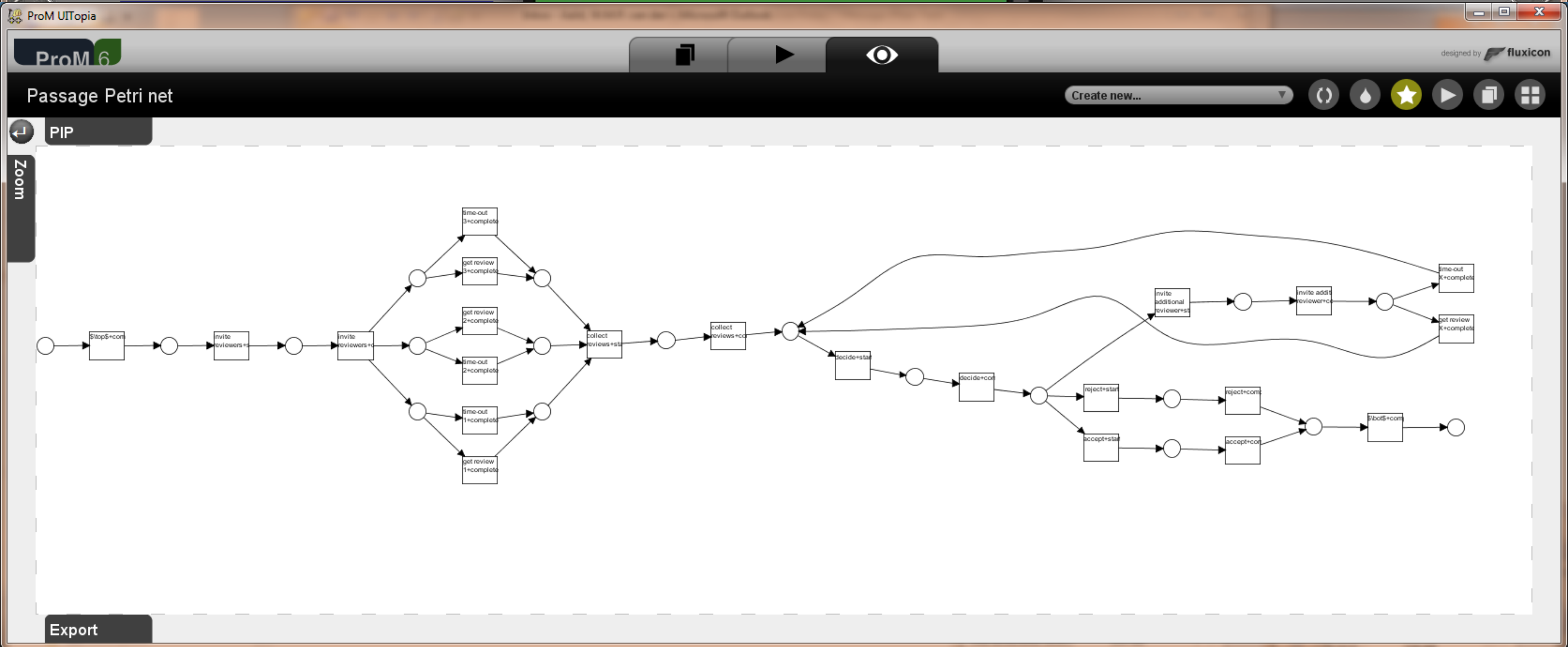
Average/case	0.00
Max.	0.00
Min.	0
Std. Deviation	0.00
#Cases with value 1.00	0

STATS INCLUDING UNRELIABLE ALIGNMENTS

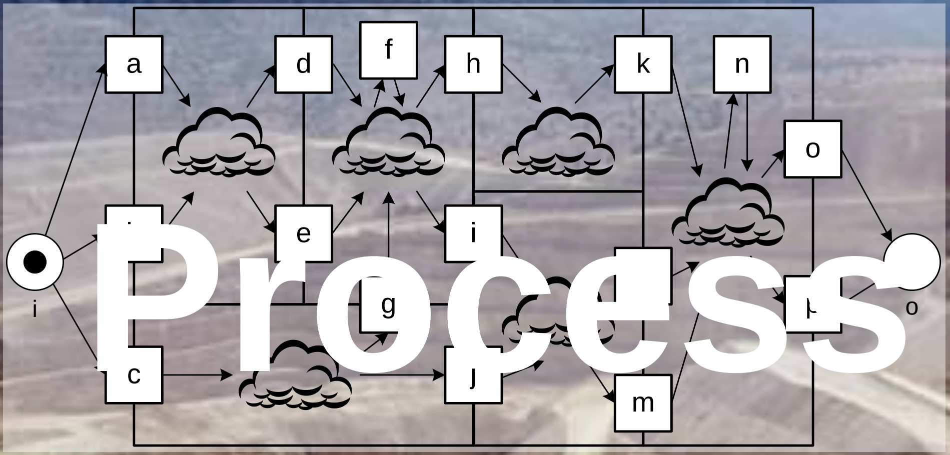
#Cases replayed	100
#Synchronous ev.class (log+model)	400
#Skipped ev.class	0
#Unobservable ev.class	0
#Inserted ev.class	0
#Violating synchronous ev.class	0

Discovery (no initial model, just events)





Summary

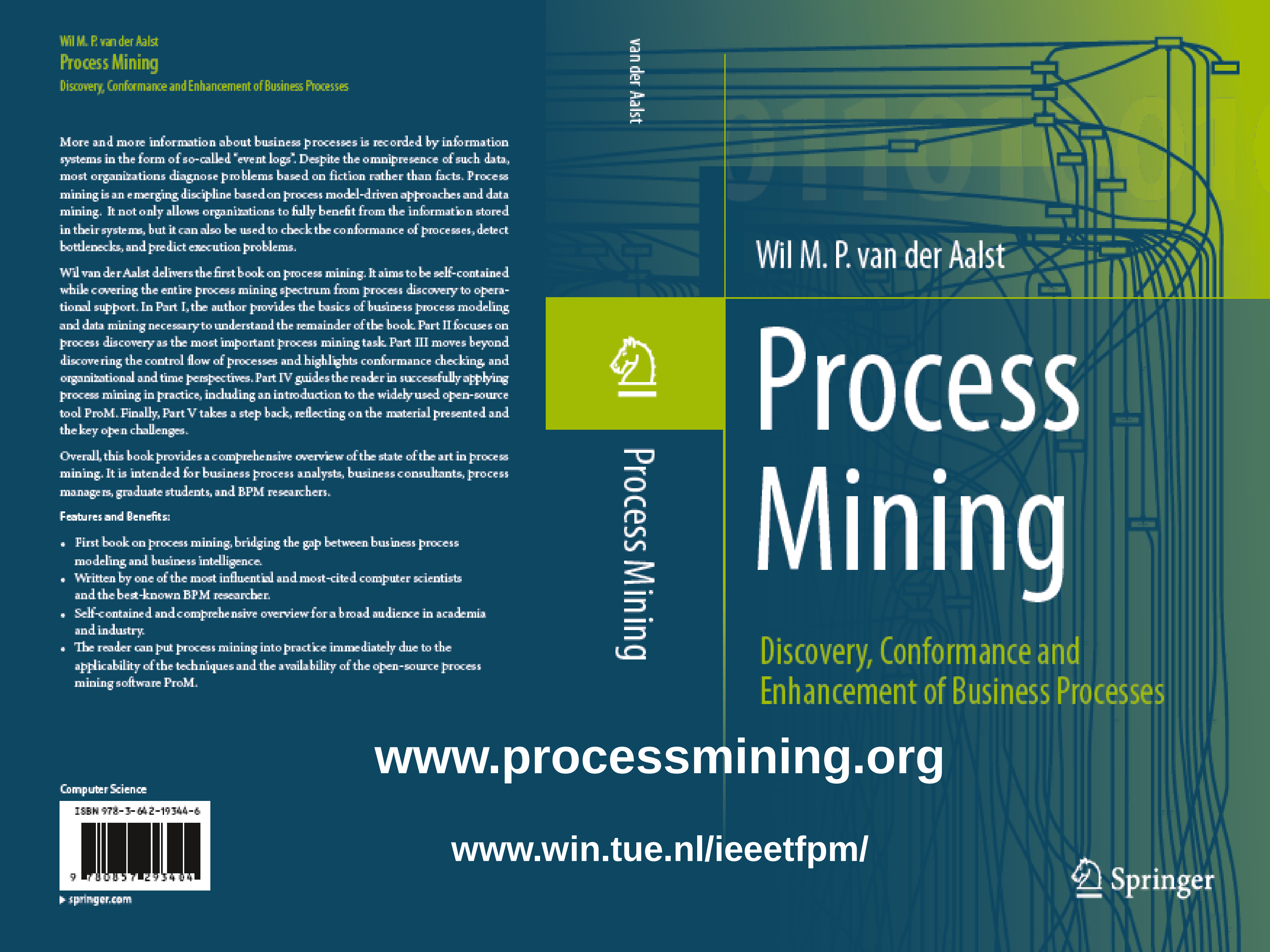


Process Mining in the Large

0010 001100110011001100100111010110

1101 00101010011001100110011001001 1

0100111010110010011101011011000011010101101001110001010100110011001100



Wil M. P. van der Aalst

Process Mining

Discovery, Conformance and Enhancement of Business Processes

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

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

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

Features and Benefits:

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

Computer Science

ISBN 978-3-642-19344-6

▸ springer.com

van der Aalst

Process Mining

Wil M. P. van der Aalst

Process Mining

Discovery, Conformance and Enhancement of Business Processes

Springer

www.processmining.org

www.win.tue.nl/ieeetfpm/