Universiteit Leiden

Business Intelligence & Process Modelling

Frank Takes

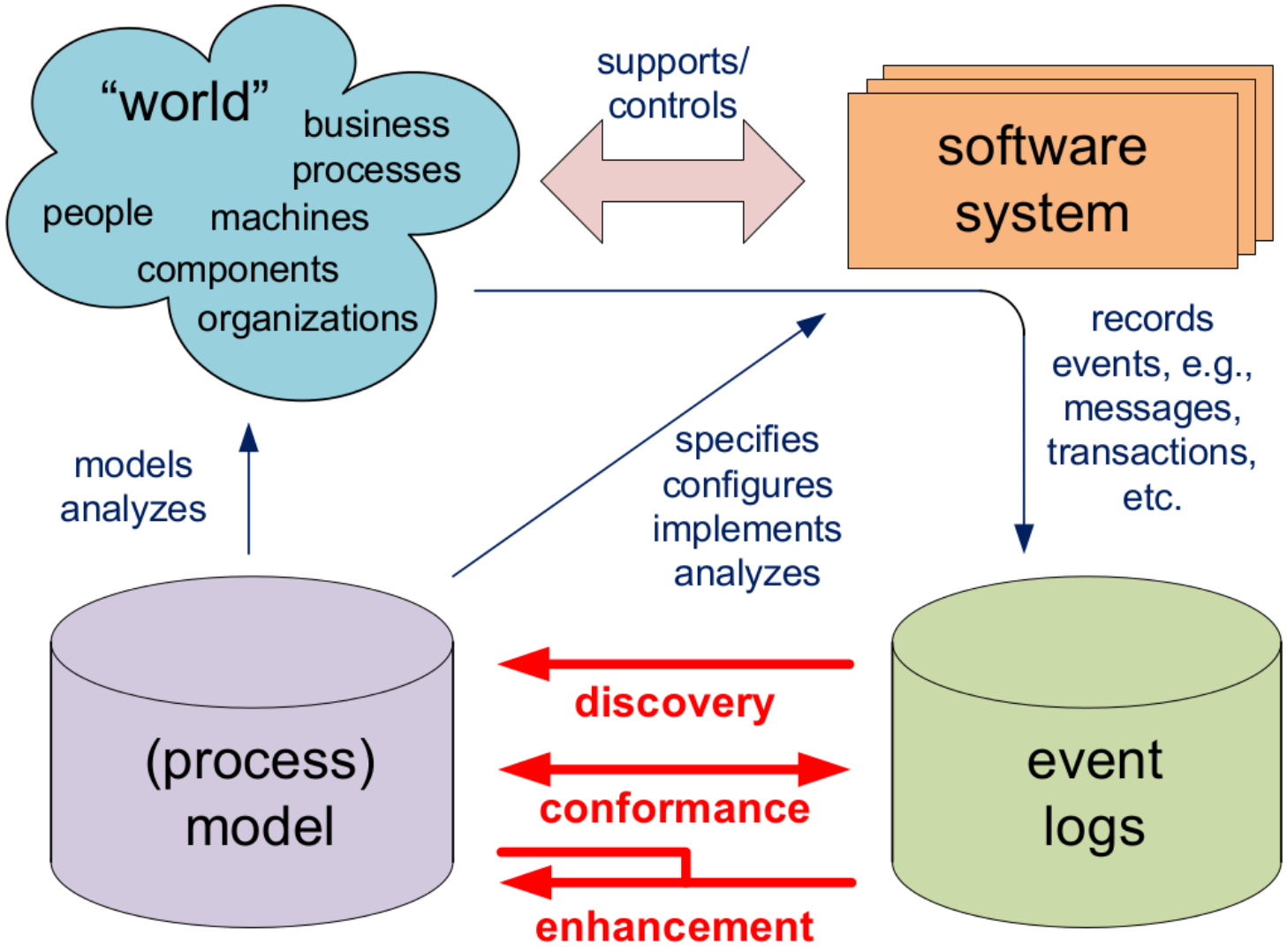
Universiteit Leiden

Lecture 10 — Process Discovery

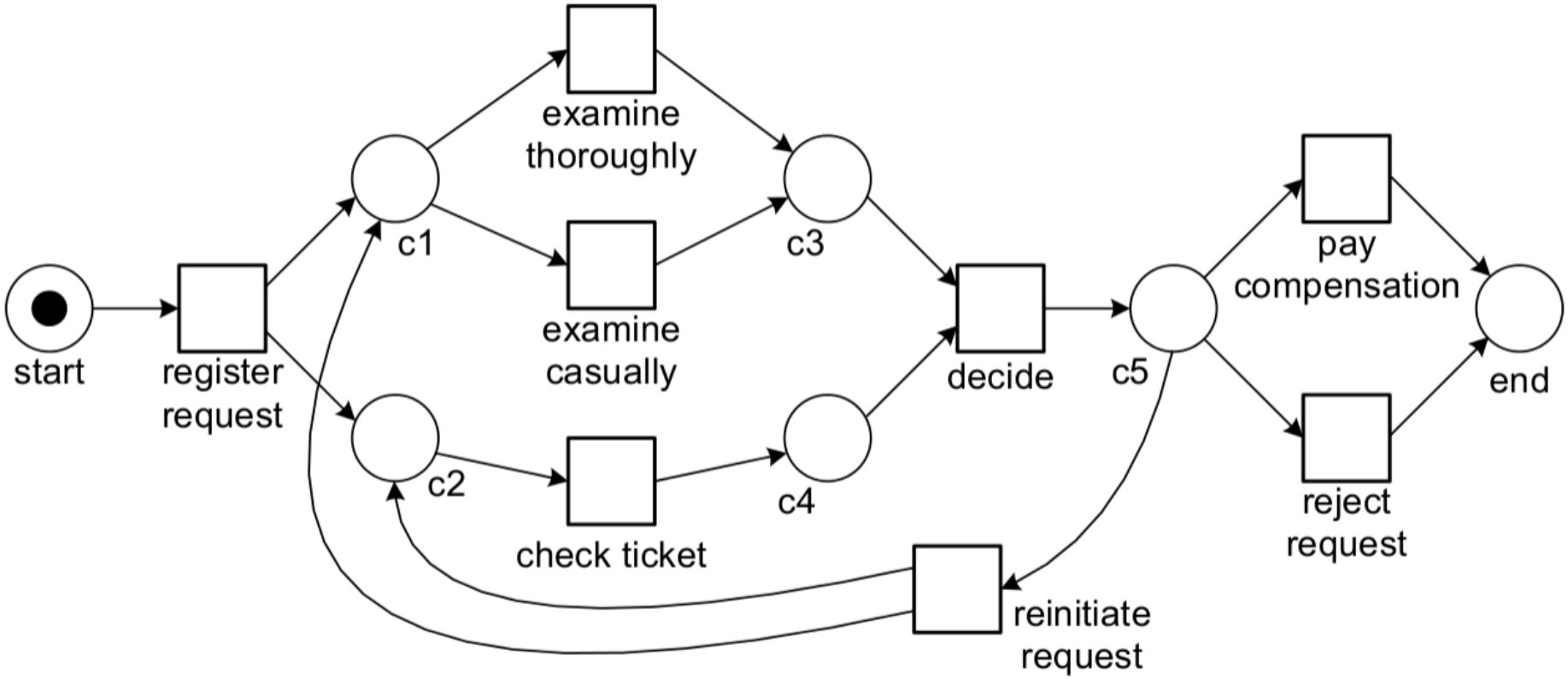
Recap

- Business Intelligence: anything that aims at providing actionable information that can be used to support business decision making (February and March)
  - Business Intelligence
  - Visual Analytics
  - Descriptive Analytics
  - Predictive Analytics
- **Process Modelling** (April and May)

Process Mining (recap)



Petri Nets (recap)



Simplified event log (recap)

Case ID	Trace
1	$\langle a, b, d, e, h \rangle$
2	$\langle a, d, c, e, g \rangle$
3	$\langle a, c, d, e, f, b, d, e, g \rangle$
4	$\langle a, d, b, e, h \rangle$
5	$\langle a, c, d, e, f, d, c, e, f, c, d, e, h \rangle$
6	$\langle a, c, d, e, g \rangle$

Table : Simplified event log of a support desk handling customer compensations (a = register request, b = examine thoroughly, c = examine casually, d = check ticket, e = decide, f = reinitiate request, g = pay compensation, h = reject request)

Simplified event log (recap)

Case ID	Trace
1	$\langle a, b, d, e, h \rangle$
2	$\langle a, d, c, e, g \rangle$
3	$\langle a, c, d, e, f, b, d, e, g \rangle$
4	$\langle a, d, b, e, h \rangle$
5	$\langle a, c, d, e, f, d, c, e, f, c, d, e, h \rangle$
6	$\langle a, c, d, e, g \rangle$

Table : Simplified event log of a support desk handling customer compensations (a = register request, b = examine thoroughly, c = examine casually, d = check ticket, e = decide, f = reinitiate request, g = pay compensation, h = reject request)

In short: $\{\langle a, b, d, e, h \rangle, \langle a, d, c, e, g \rangle, \langle a, c, d, e, f, b, d, e, g \rangle, \langle a, d, b, e, h \rangle, \langle a, c, d, e, f, d, c, e, f, c, d, e, h \rangle, \langle a, c, d, e, g \rangle\}$

Labeled Petri Nets (recap)

- Petri net $N = (P, T, F, A, \ell)$
- P is a finite set of **places**
- T is a finite set of **transitions**
- $F \subseteq (P \times T) \cup (T \times P)$ is a finite set of directed **arcs** called the **flow relation**
- A is a set of activity **labels**
- $\ell : T \to A$ is a **labeling function**

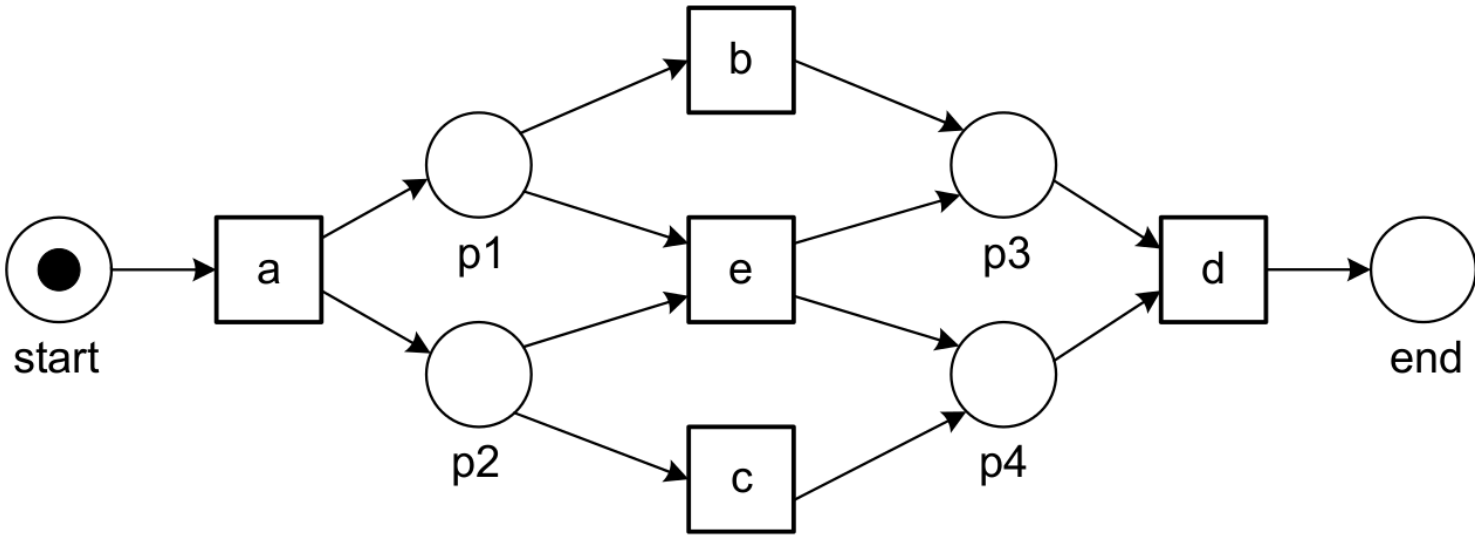
Universiteit Leiden

Process Mining

Petri net variants

- **WorkFlow-net** (WF-net): Petri net with fixed source $i \in P$ (without inputs) and target $o \in P$ (without outputs)
- **Sound** Petri net:
  - is **safe**: places cannot hold multiple tokens at the same time
  - has **proper completion**: if the output is marked, it is the only marked place
  - has the **option to complete**: for any marking, it is possible to reach the output
  - has **no dead transitions**: each transition can be reached from the input marking

Example

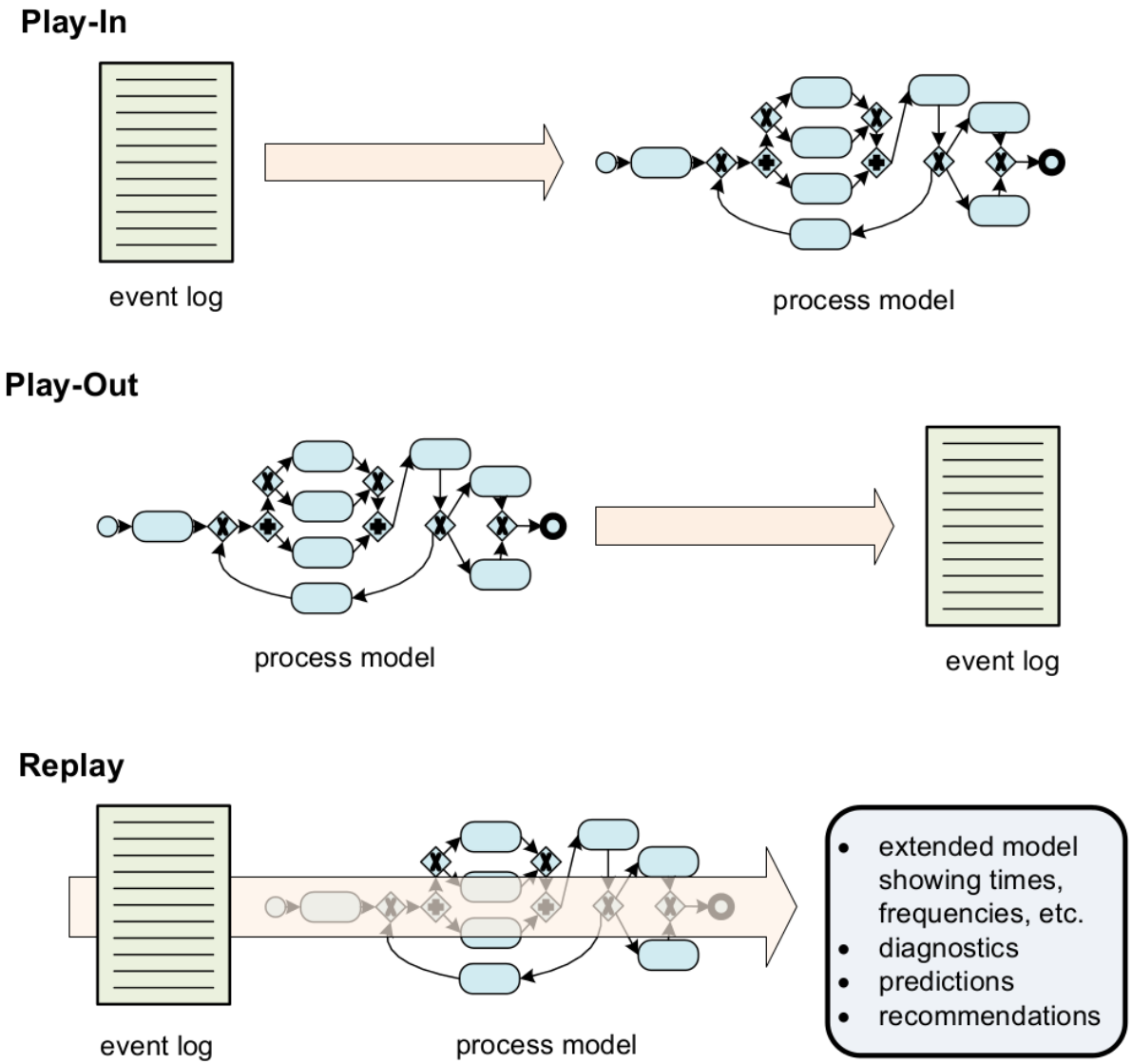


$$L_1 = [\langle a,b,c,d\rangle^3, \langle a,c,b,d\rangle^2, \langle a,e,d\rangle]$$

Process discovery

- **Process discovery algorithm** (general): given an event log, find a way to map this event log onto a process model such that the model is "representative" for the behavior seen in the event log
- **Process discovery algorithm** (formal): given an event log L, find a function γ which maps L onto a marked Petri Net $\gamma(L) = (N, M)$ such that all traces in L correspond to possible firing sequences of (N, M).
- Input: event log L containing a multi-set of traces, e.g., $L = [\langle a, b, c, d \rangle^3, \langle a, c, b, d \rangle^2, \langle a, e, d \rangle]$
- Output: marked Petri net (N, M)

Play-in!


Play-In
event log
process model
Play-Out
process model
event log
Replay
event log
process model
extended model showing times, frequencies, etc.
diagnostics
predictions
recommendations

Requirements

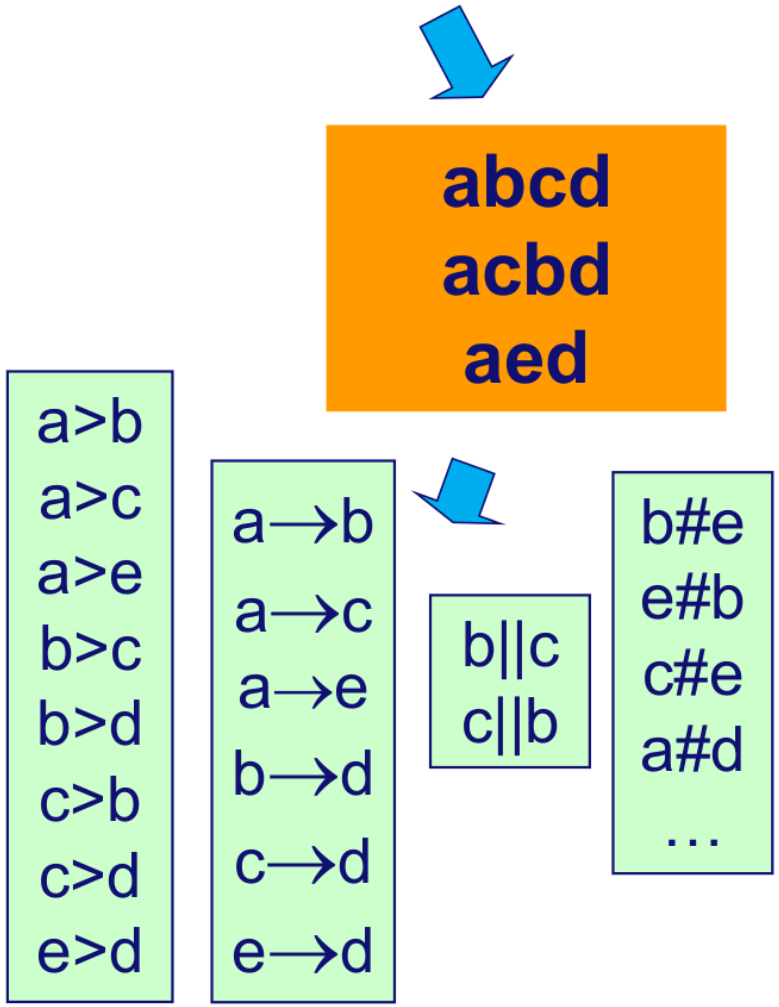
- **Fitness**: the discovered model should allow for the behavior seen in the event log
- **Precision**: the discovered model should not allow for behavior completely unrelated to what was seen in the event log (prevent **underfitting**)
- **Generalization**: the discovered model should generalize the example behavior seen in the event log (prevent **overfitting**)
- **Simplicity**: the discovered model should be as simple as possible

Log-based ordering relations

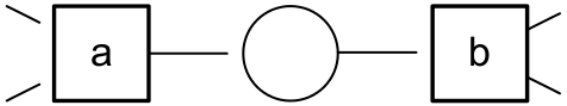
$$L = [\langle a, b, c, d \rangle^3, \langle a, c, b, d \rangle^2, \langle a, e, d \rangle]$$

- **Direct succession**: $a >_L b$
- **Causality**: if $a >_L b$ and $b \not> _L a$ $a \rightarrow_L b$
- **Choice**: if $a \not> _L b$ and $b \not> _L a$ $a \#_L b$
- **Parallel**: if $a >_L b$ and $b >_L a$ $a \|_L b$

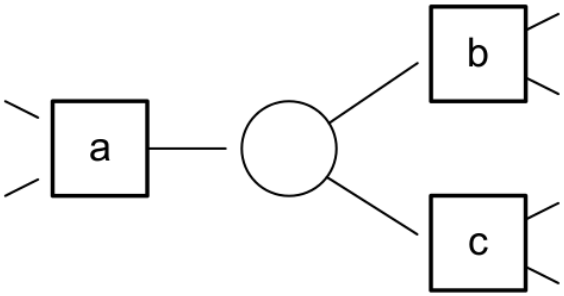
$$L_1 = [\langle a,b,c,d\rangle^3, \langle a,c,b,d\rangle^2, \langle a,e,d\rangle]$$



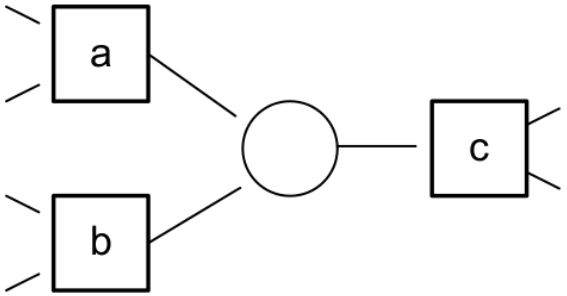
Relations of simple patterns



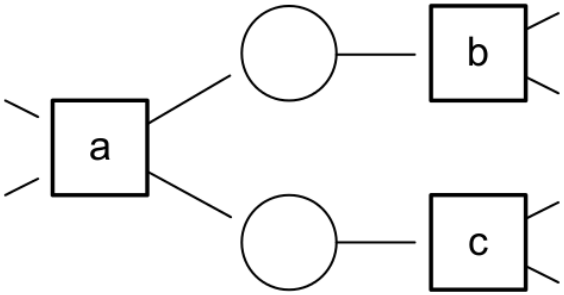
(a) sequence pattern: a→b



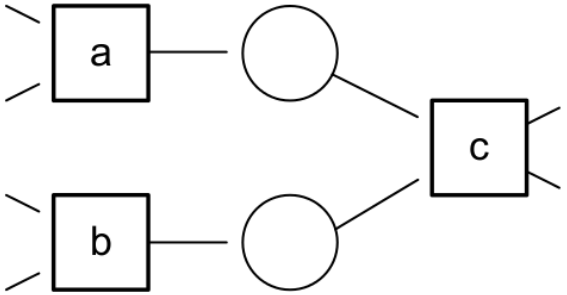
(b) XOR-split pattern:
a→b, a→c, and b#c



(c) XOR-join pattern:
a→c, b→c, and a#b

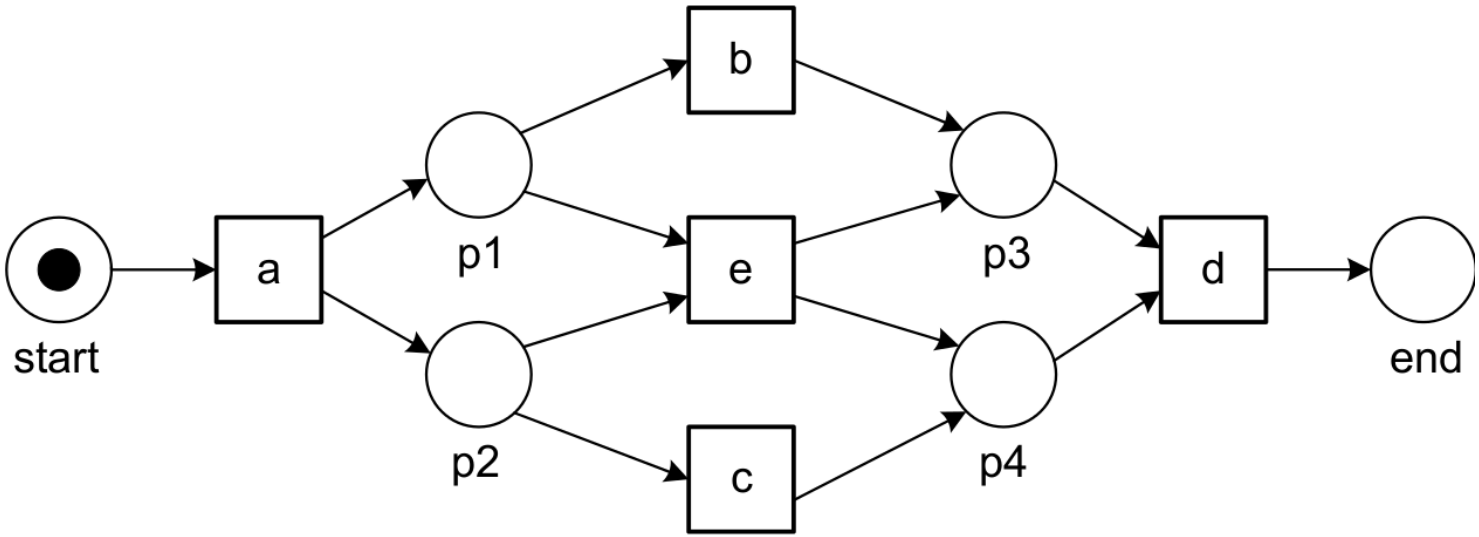


(d) AND-split pattern:
a→b, a→c, and b||c



(e) AND-join pattern:
a→c, b→c, and a||b

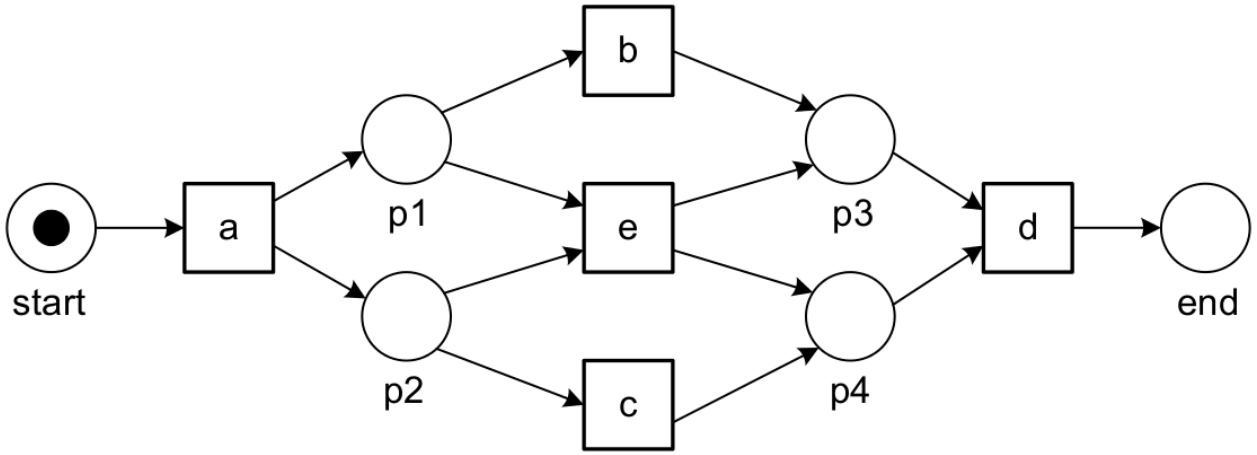
Example



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

Footprint

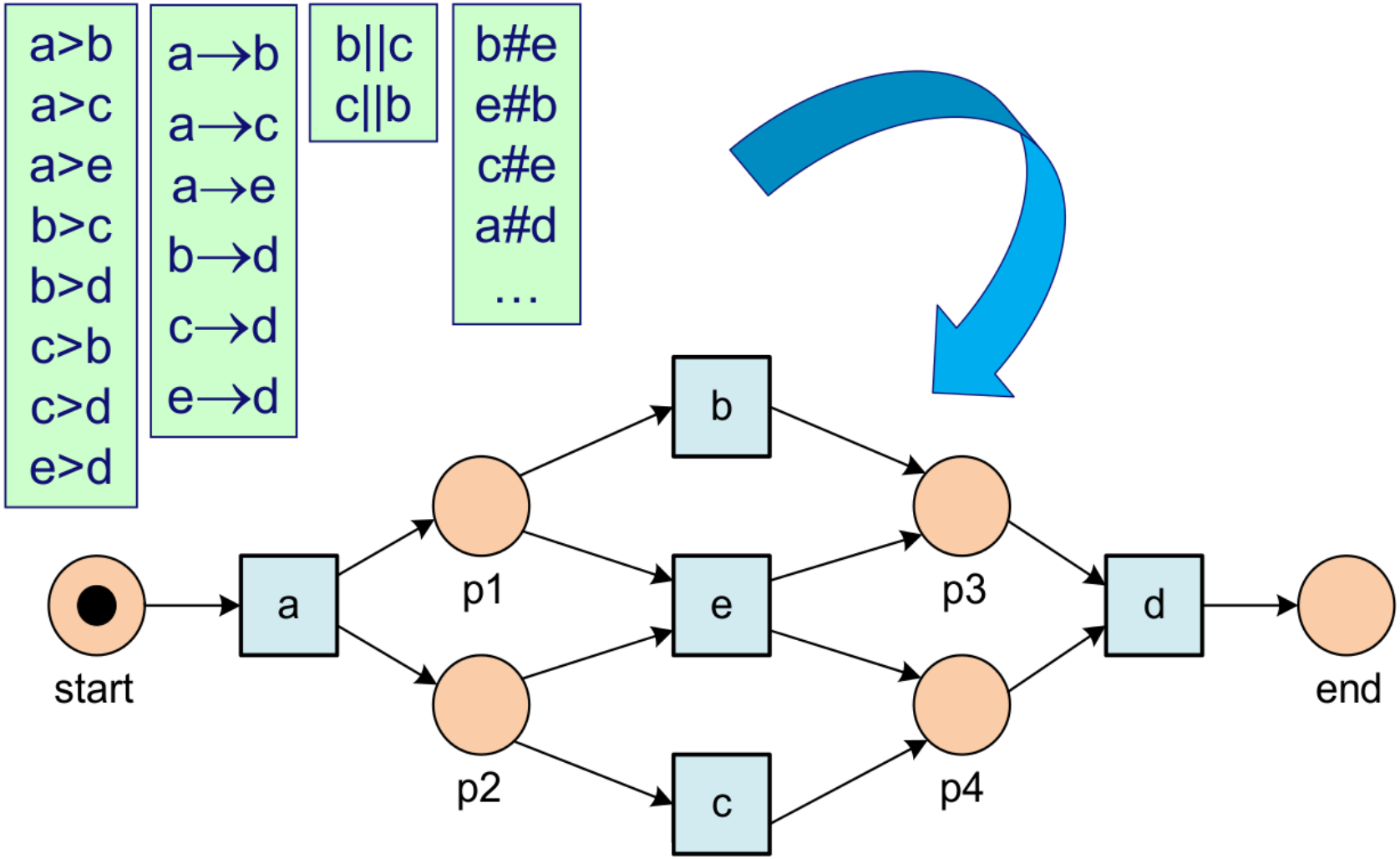
$$L_1 = [\langle a,b,c,d\rangle^3, \langle a,c,b,d\rangle^2, \langle a,e,d\rangle]$$



	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	$\leftarrow_{L_1}$	$\#_{L_1}$	$\#_{L_1}$	$\rightarrow_{L_1}$	$\#_{L_1}$

The final step?

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



α-algorithm

Universiteit Leiden

Let L be an event log over T. $\alpha(L)$ is defined as follows.

1. $T_L = \{ t \in T \mid \exists_{\sigma \in L}\, t \in \sigma \}$,
2. $T_I = \{ t \in T \mid \exists_{\sigma \in L}\, t = \mathit{first}(\sigma) \}$,
3. $T_O = \{ t \in T \mid \exists_{\sigma \in L}\, t = \mathit{last}(\sigma) \}$,
4. $X_L = \{ (A,B) \mid A \subseteq T_L \wedge A \neq \emptyset \wedge B \subseteq T_L \wedge B \neq \emptyset \wedge \forall_{a \in A} \forall_{b \in B}\, a \rightarrow_L b \wedge \forall_{a1,a2 \in A}\, a_1 \#_L a_2 \wedge \forall_{b1,b2 \in B}\, b_1 \#_L b_2 \}$,
5. $Y_L = \{ (A,B) \in X_L \mid \forall_{(A',B') \in X_L}\, A \subseteq A' \wedge B \subseteq B' \Rightarrow (A,B) = (A',B') \}$,
6. $P_L = \{ p_{(A,B)} \mid (A,B) \in Y_L \} \cup \{i_L, o_L\}$,
7. $F_L = \{ (a, p_{(A,B)}) \mid (A,B) \in Y_L \wedge a \in A \} \cup \{ (p_{(A,B)}, b) \mid (A,B) \in Y_L \wedge b \in B \} \cup \{ (i_L, t) \mid t \in T_I \} \cup \{ (t, o_L) \mid t \in T_O \}$, and
8. $\alpha(L) = (P_L, T_L, F_L)$.

α-algorithm

1. All events are mapped directly to the set of **transitions** T_L
2. All events occuring first in some trace are stored in T_I
3. All events occuring last in some trace are stored in T_O

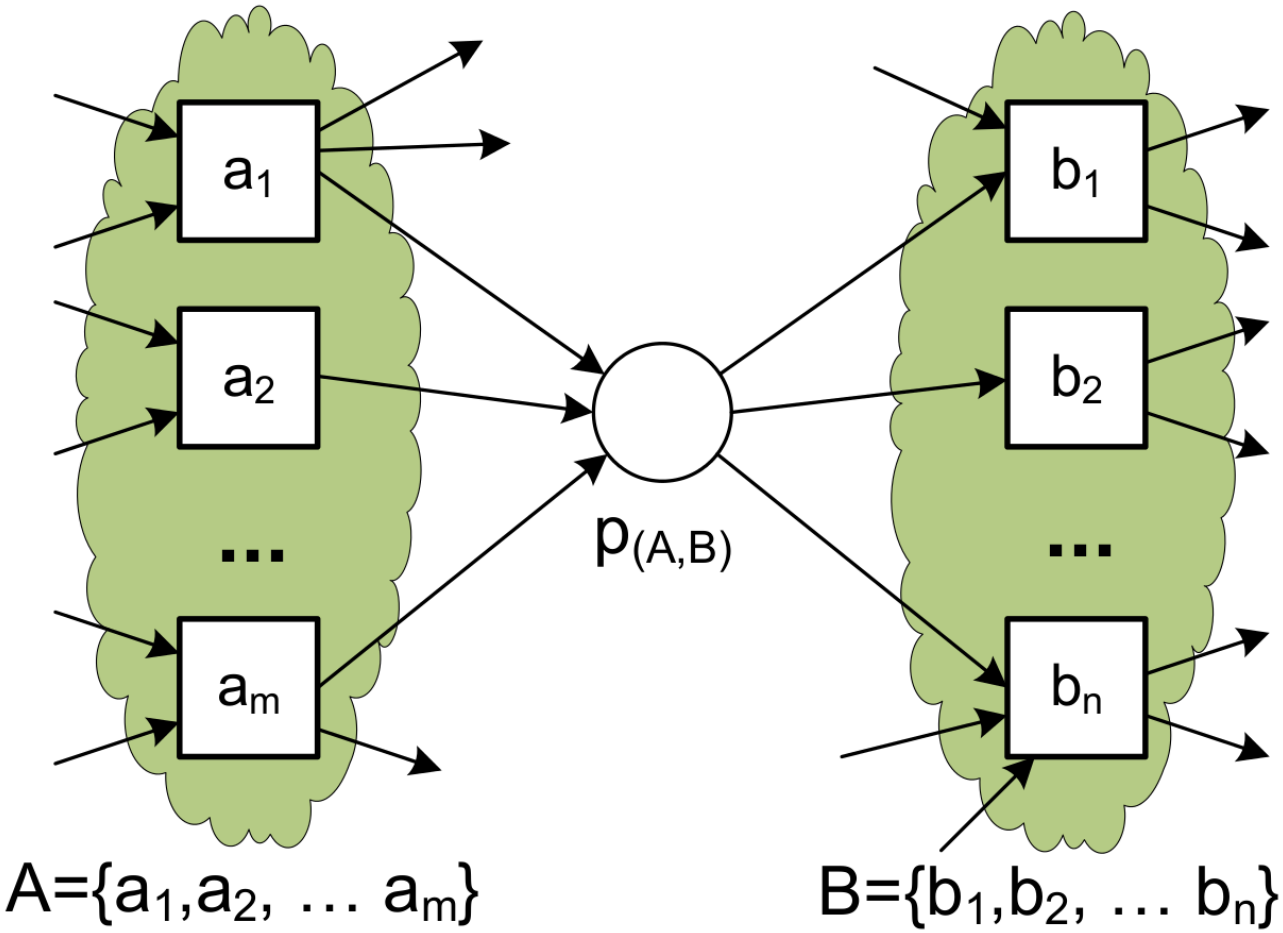
α-algorithm

1. All events are mapped directly to the set of **transitions** T_L
2. All events occuring first in some trace are stored in T_I
3. All events occuring last in some trace are stored in T_O
4. Generate X_L, containing all possible divisions (A, B) of events into sets A and B such that
   - for all $a \in A$ and $b \in B$ there is a causal relation $a \rightarrow_L b$ and
   - for all $a_1, a_2 \in A$ it holds that $a_1 \#_L a_2$ (and analogously for all $b_1, b_2 \in B$)
5. Define Y_L as the "maximal pairs' from X_L

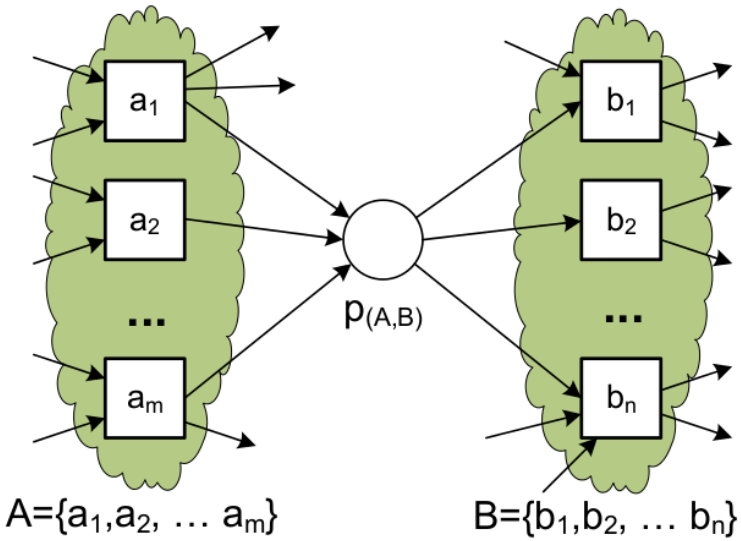
α-algorithm

1. All events are mapped directly to the set of **transitions** T_L
2. All events occuring first in some trace are stored in T_I
3. All events occuring last in some trace are stored in T_O
4. Generate X_L, containing all possible divisions (A, B) of events into sets A and B such that
   - for all $a \in A$ and $b \in B$ there is a causal relation $a \rightarrow_L b$ and
   - for all $a_1, a_2 \in A$ it holds that $a_1 \#_L a_2$ (and analogously for all $b_1, b_2 \in B$)
5. Define Y_L as the "maximal pairs' from X_L
6. Define set of **places** $P_L = \{p_{(A,B)} \mid (A, B) \in Y_L\} \cup \{i_L, o_L\}$
7. Generate **arcs** defining the flow relation F_L:
   - from the input place i_L to starting transitions in T_I and
   - for each of the pairs in $(A, B) \in Y_L$, one place with input arcs from all $i \in A$ and output arcs to all $j \in B$
   - from the ending transitions in T_O to output place o_L
8. Create the **Petri net** $N_L = \alpha(L) = (P_L, T_L, F_L)$

Step 4



Step 4, using the footprint



	a_1	a_2	...	a_m	b_1	b_2	...	b_n
a_1	#	#	...	#	→	→	...	→
a_2	#	#	...	#	→	→	...	→
...	...	...	...	...	...	...	...	...
a_m	#	#	...	#	→	→	...	→
b_1	←	←	...	←	#	#	...	#
b_2	←	←	...	←	#	#	...	#
...	...	...	...	...	...	...	...	...
b_n	←	←	...	←	#	#	...	#

Example run

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

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

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

1. **Transitions** $T_L = \{a, b, c, d, e\}$
2. $T_I = \{a\}$
3. $T_O = \{d\}$

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

1. **Transitions** $T_L = \{a, b, c, d, e\}$
2. $T_I = \{a\}$
3. $T_O = \{d\}$

	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	$\leftarrow_{L_1}$	$\#_{L_1}$	$\#_{L_1}$	$\rightarrow_{L_1}$	$\#_{L_1}$

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

1. **Transitions** $T_L = \{a, b, c, d, e\}$
2. $T_I = \{a\}$
3. $T_O = \{d\}$

	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	$\leftarrow_{L_1}$	$\#_{L_1}$	$\#_{L_1}$	$\rightarrow_{L_1}$	$\#_{L_1}$

4. $X_{L_1} = \{$
$(\{a\}, \{b\}), (\{a\}, \{c\}), (\{a\}, \{e\}), (\{a\}, \{b, e\}), (\{a\}, \{c, e\}),$
$(\{b\}, \{d\}), (\{c\}, \{d\}), (\{e\}, \{d\}), (\{b, e\}, \{d\}), (\{c, e\}, \{d\})$ $\}$

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

1. **Transitions** $T_L = \{a, b, c, d, e\}$
2. $T_I = \{a\}$
3. $T_O = \{d\}$

	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	$\leftarrow_{L_1}$	$\#_{L_1}$	$\#_{L_1}$	$\rightarrow_{L_1}$	$\#_{L_1}$

4. $X_{L_1} = \{$
   $(\{a\}, \{b\}), (\{a\}, \{c\}), (\{a\}, \{e\}), (\{a\}, \{b, e\}), (\{a\}, \{c, e\}),$
   $(\{b\}, \{d\}), (\{c\}, \{d\}), (\{e\}, \{d\}), (\{b, e\}, \{d\}), (\{c, e\}, \{d\})$ $\}$
5. $Y_{L_1} = \{$ $(\{a\}, \{b, e\}), (\{a\}, \{c, e\}), (\{b, e\}, \{d\}), (\{c, e\}, \{d\})$ $\}$

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

1. **Transitions** $T_L = \{a, b, c, d, e\}$
2. $T_I = \{a\}$
3. $T_O = \{d\}$

	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	$\leftarrow_{L_1}$	$\#_{L_1}$	$\#_{L_1}$	$\rightarrow_{L_1}$	$\#_{L_1}$

4. $X_{L_1} = \{$
   $(\{a\}, \{b\}), (\{a\}, \{c\}), (\{a\}, \{e\}), (\{a\}, \{b, e\}), (\{a\}, \{c, e\}),$
   $(\{b\}, \{d\}), (\{c\}, \{d\}), (\{e\}, \{d\}), (\{b, e\}, \{d\}), (\{c, e\}, \{d\})$ $\}$
5. $Y_{L_1} = \{$ $(\{a\}, \{b, e\}), (\{a\}, \{c, e\}), (\{b, e\}, \{d\}), (\{c, e\}, \{d\})$ $\}$
6. $P_L = \{p_{1(\{a\},\{b,e\})}, p_{2(\{a\},\{c,e\})}, p_{3(\{b,e\},\{d\})}, p_{4(\{c,e\},\{d\})}, i_L, o_L\}$

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

1. **Transitions** $T_L = \{a, b, c, d, e\}$
2. $T_I = \{a\}$
3. $T_O = \{d\}$

	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	$\leftarrow_{L_1}$	$\#_{L_1}$	$\#_{L_1}$	$\rightarrow_{L_1}$	$\#_{L_1}$

4. $X_{L_1} = \{$
   $(\{a\}, \{b\}), (\{a\}, \{c\}), (\{a\}, \{e\}), (\{a\}, \{b, e\}), (\{a\}, \{c, e\}),$
   $(\{b\}, \{d\}), (\{c\}, \{d\}), (\{e\}, \{d\}), (\{b, e\}, \{d\}), (\{c, e\}, \{d\})$ $\}$
5. $Y_{L_1} = \{$ $(\{a\}, \{b, e\}), (\{a\}, \{c, e\}), (\{b, e\}, \{d\}), (\{c, e\}, \{d\})$ $\}$
6. $P_L = \{p_{1(\{a\},\{b,e\})}, p_{2(\{a\},\{c,e\})}, p_{3(\{b,e\},\{d\})}, p_{4(\{c,e\},\{d\})}, i_L, o_L\}$
7. **Arcs** F_L (trivial given step 2, 3 and 6)

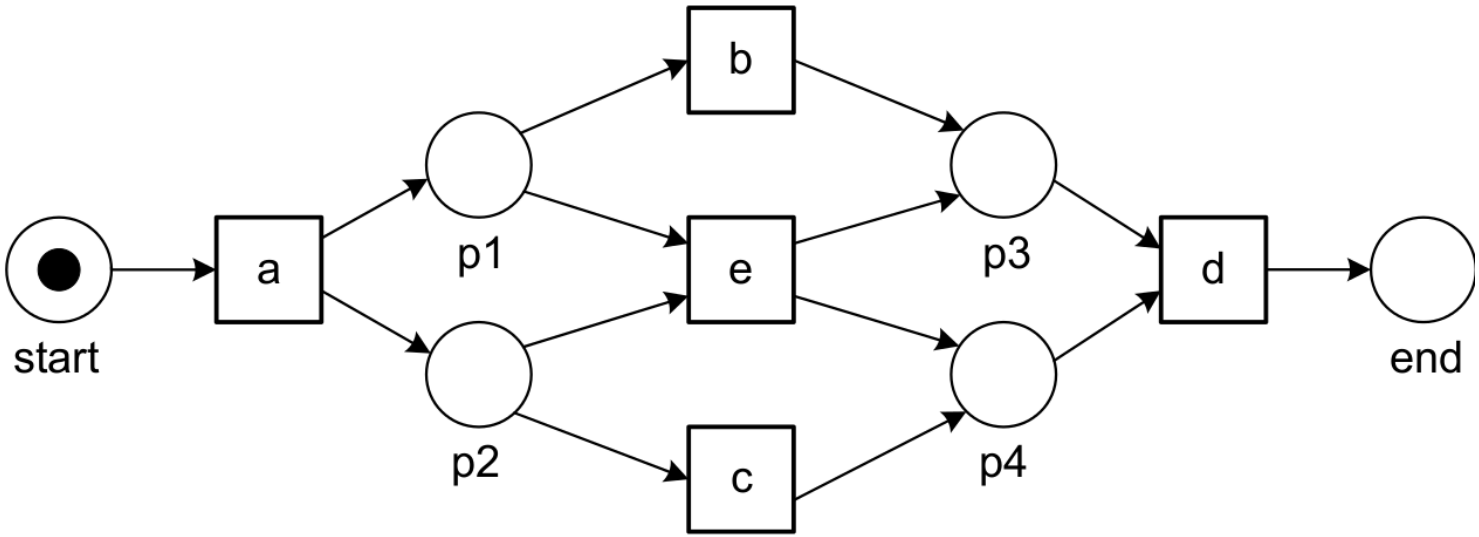
$$L_1 = [\langle a, b, c, d\rangle^3, \langle a, c, b, d\rangle^2, \langle a, e, d\rangle]$$

1. **Transitions** $T_L = \{a, b, c, d, e\}$
2. $T_I = \{a\}$
3. $T_O = \{d\}$

	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	$\leftarrow_{L_1}$	$\#_{L_1}$	$\#_{L_1}$	$\rightarrow_{L_1}$	$\#_{L_1}$

4. $X_{L_1} = \{$
   $(\{a\}, \{b\}), (\{a\}, \{c\}), (\{a\}, \{e\}), (\{a\}, \{b, e\}), (\{a\}, \{c, e\}),$
   $(\{b\}, \{d\}), (\{c\}, \{d\}), (\{e\}, \{d\}), (\{b, e\}, \{d\}), (\{c, e\}, \{d\})$ $\}$
5. $Y_{L_1} = \{$ $(\{a\}, \{b, e\}), (\{a\}, \{c, e\}), (\{b, e\}, \{d\}), (\{c, e\}, \{d\})$ $\}$
6. $P_L = \{p_{1(\{a\},\{b,e\})}, p_{2(\{a\},\{c,e\})}, p_{3(\{b,e\},\{d\})}, p_{4(\{c,e\},\{d\})}, i_L, o_L\}$
7. **Arcs** F_L (trivial given step 2, 3 and 6)
8. **Petri net** $N_L = (P_L, T_L, F_L)$

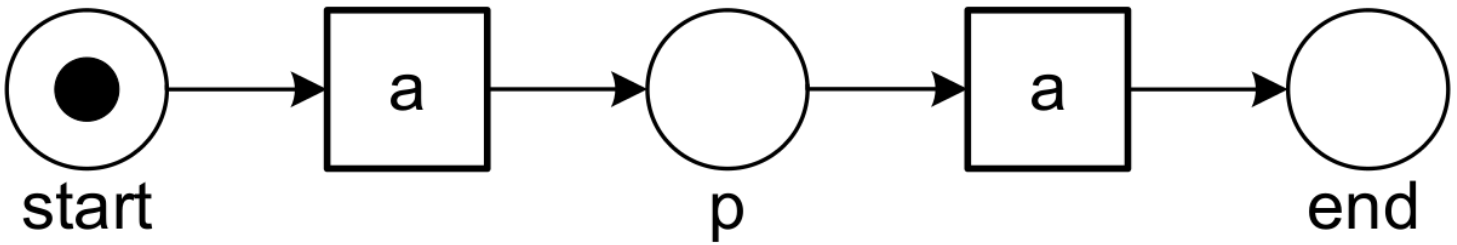
Example run



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

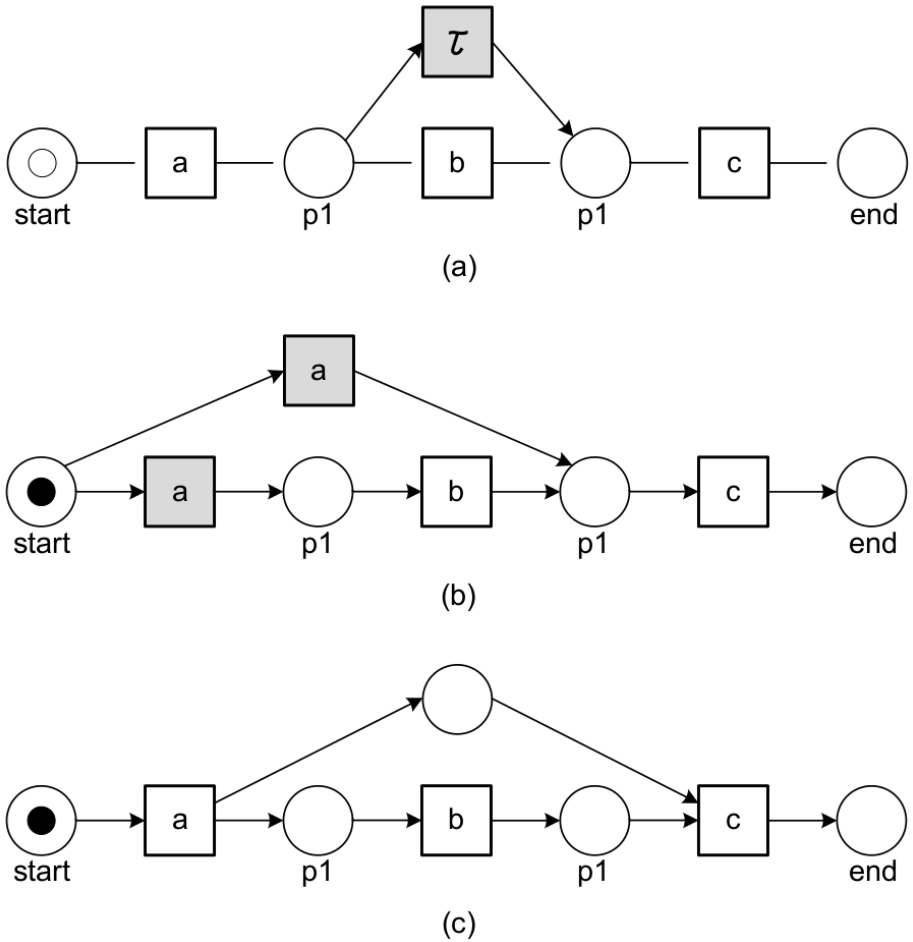
No WF-net with unique labels possible (1)

$$L_{10} = [\langle a, a \rangle^{55}]$$



No WF-net with unique labels possible (2)

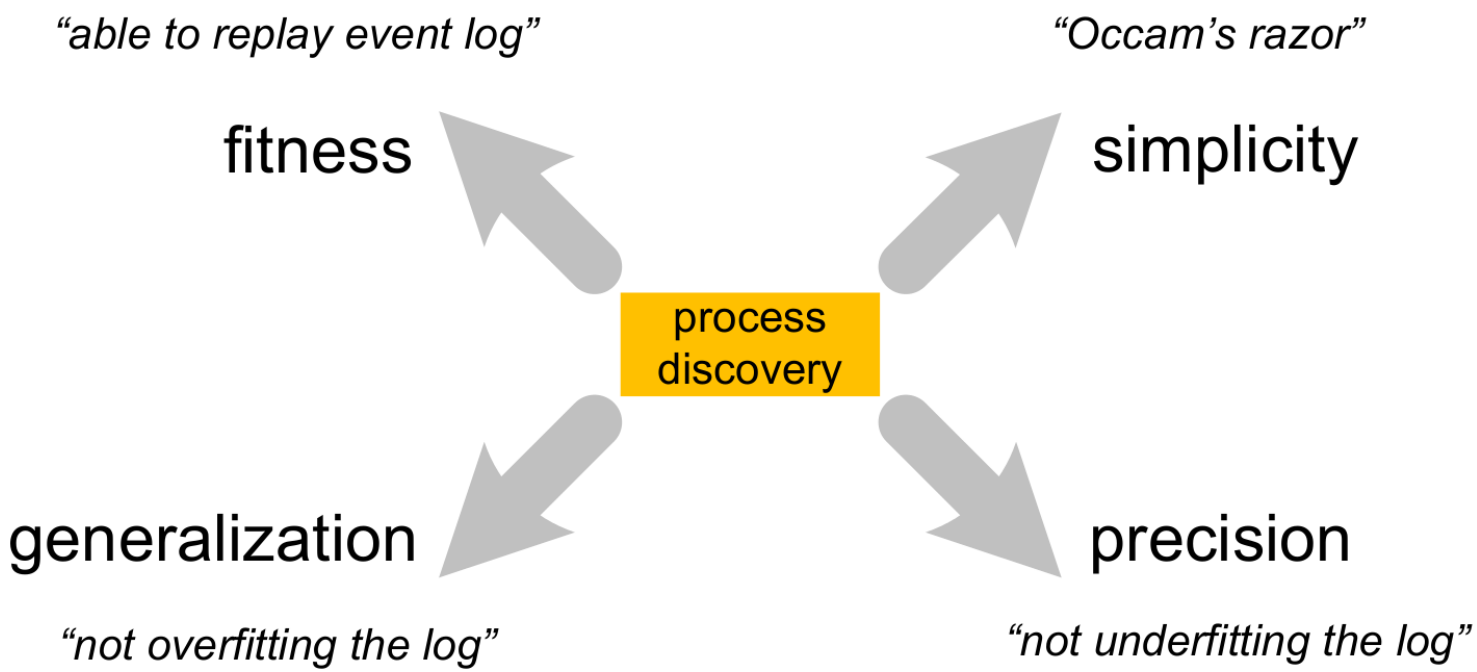
$$L_{11} = [\langle a, b, c \rangle^{20}, \langle a, c \rangle^{30}]$$



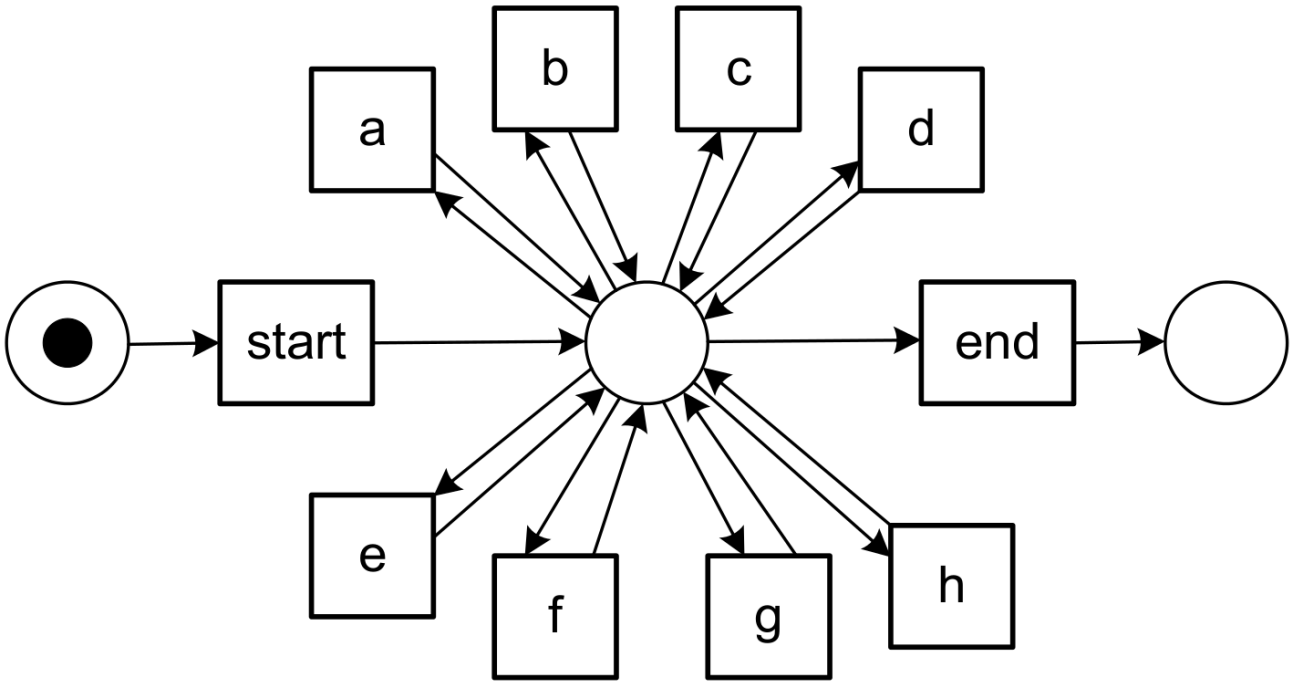
Challenge: noise and incompleteness

- To discover a suitable process model it is assumed that the event log contains a representative sample of behavior
- Two related phenomena:
  - **Noise**: the event log contains rare and infrequent behavior not representative for the typical behavior of the process
  - **Incompleteness**: the event log contains too few events to be able to discover some of the underlying control-flow structures

Balancing Underfitting and Overfitting

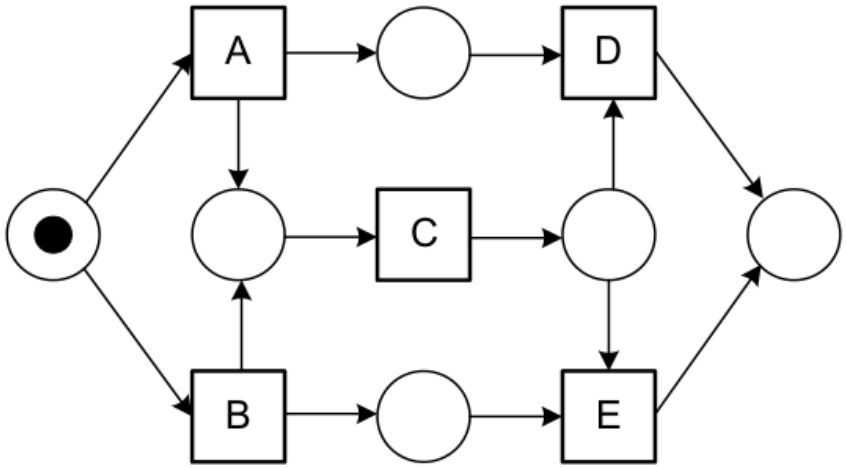


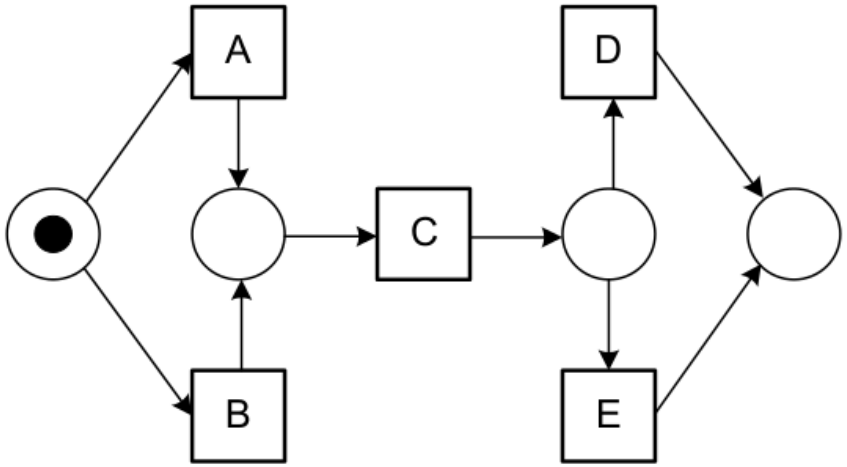
Flowermodel



What is the best model? (1)

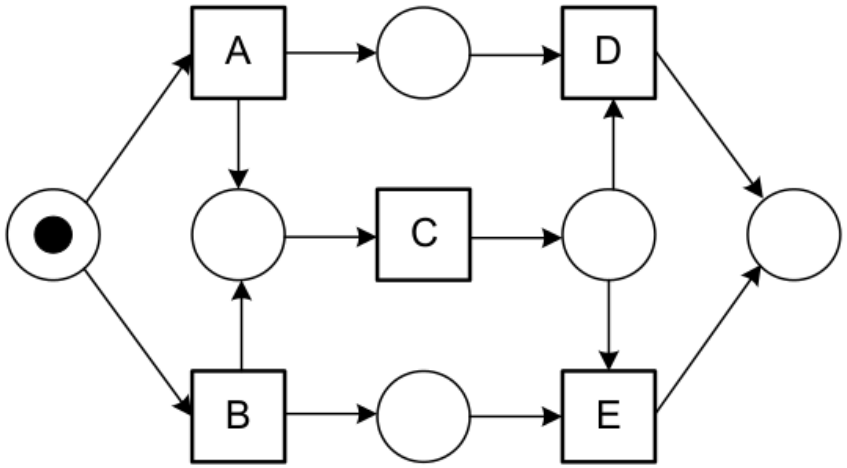
ACD	99
ACE	0
BCE	85
BCD	0

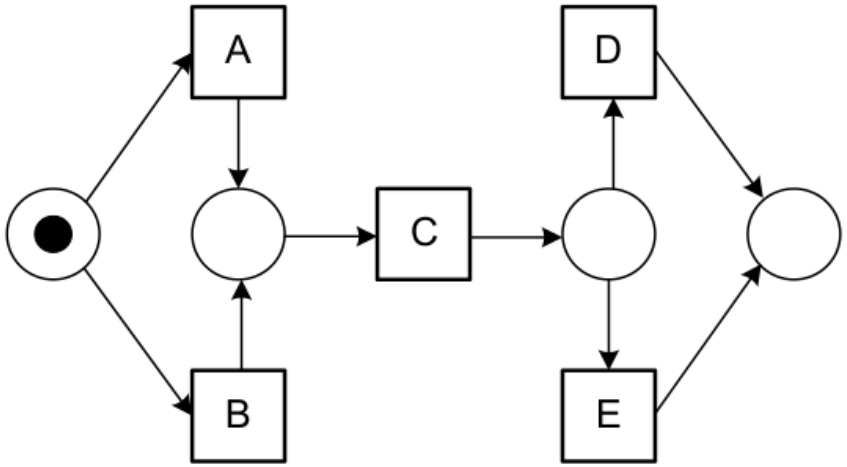




What is the best model? (1)

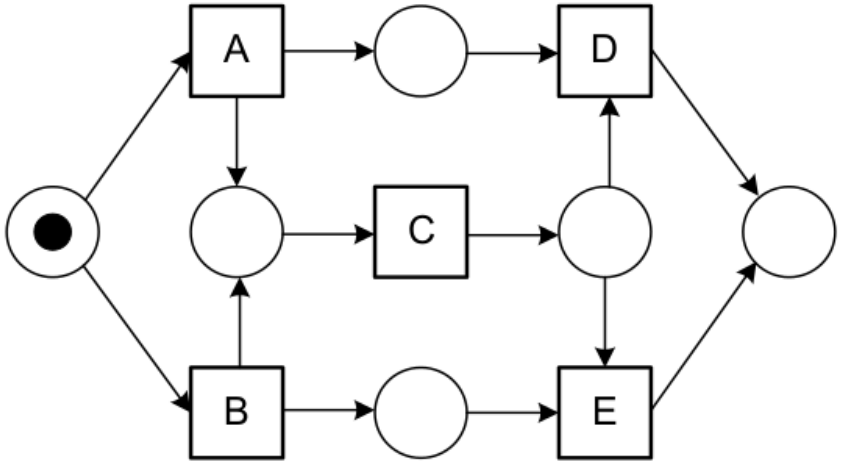
ACD	99
ACE	0
BCE	85
BCD	0

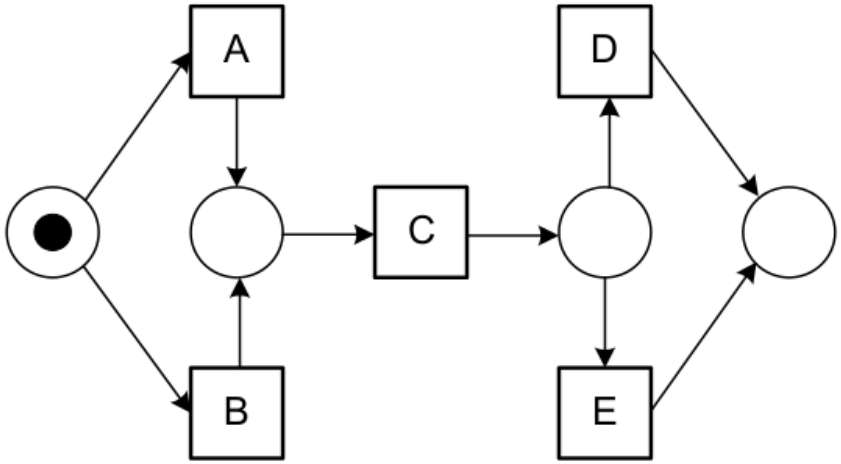




What is the best model? (2)

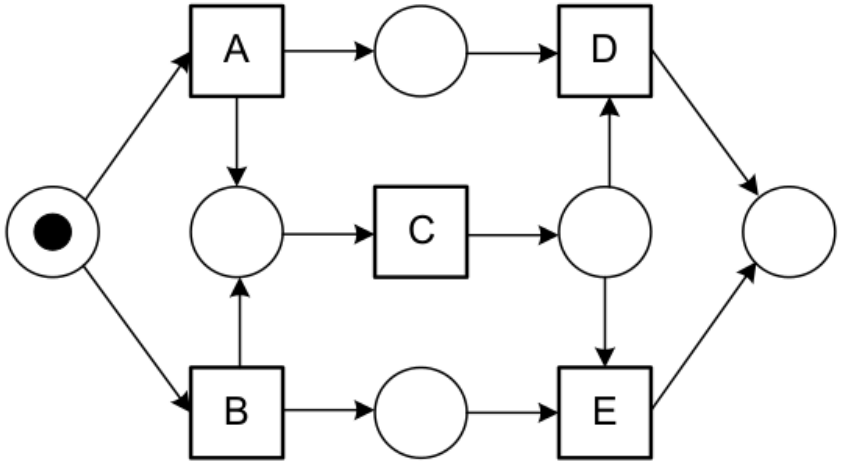
ACD	99
ACE	88
BCE	85
BCD	78

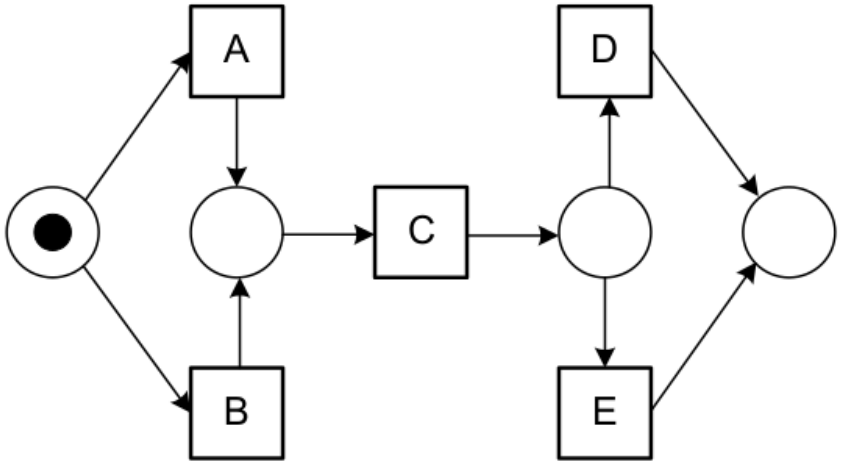




What is the best model? (2)

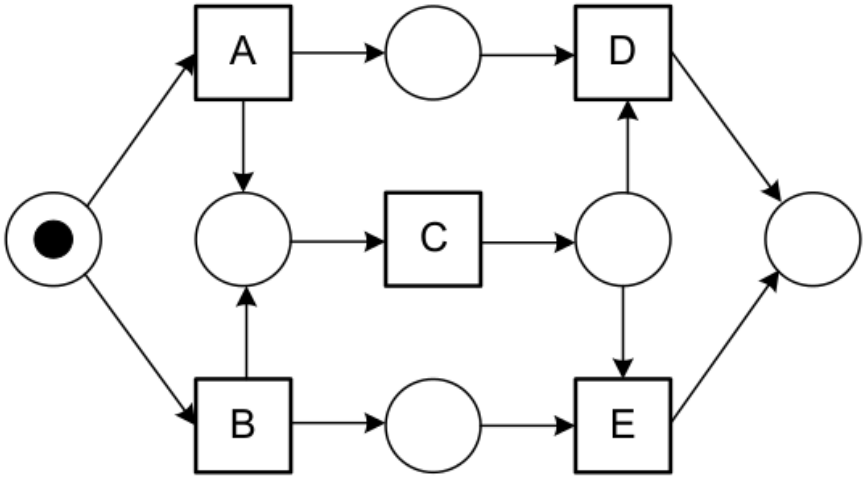
ACD	99
ACE	88
BCE	85
BCD	78

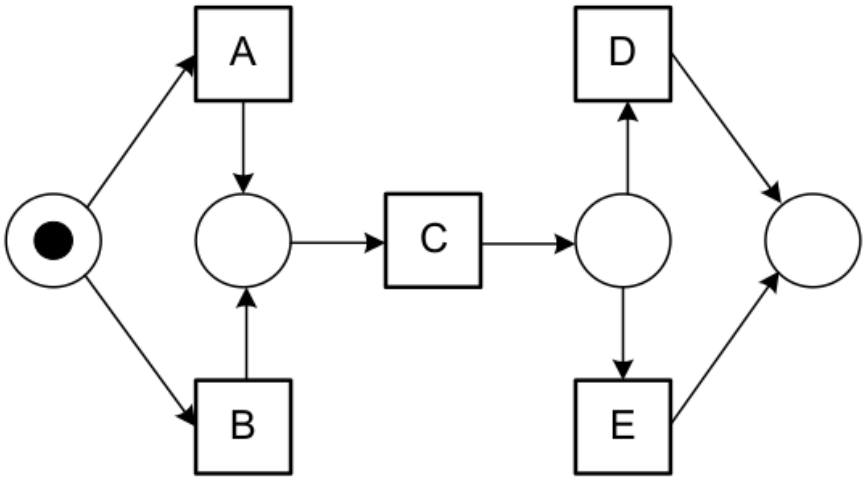




What is the best model? (3)

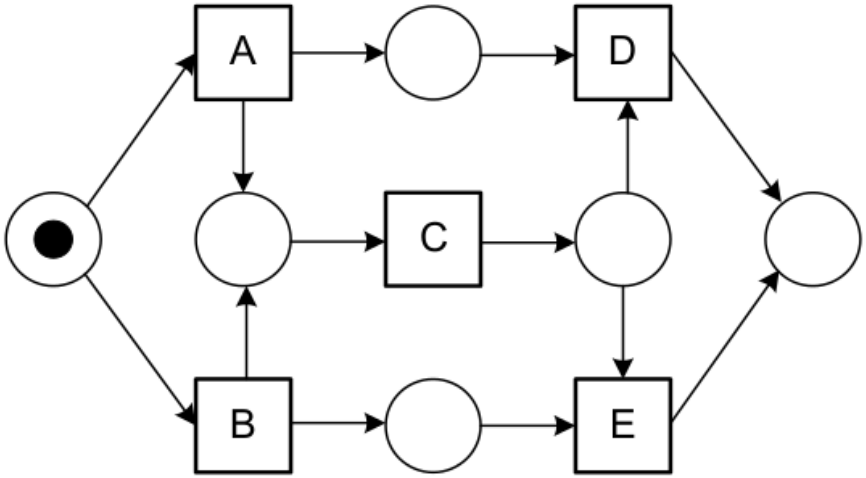
ACD	99
ACE	2
BCE	85
BCD	3

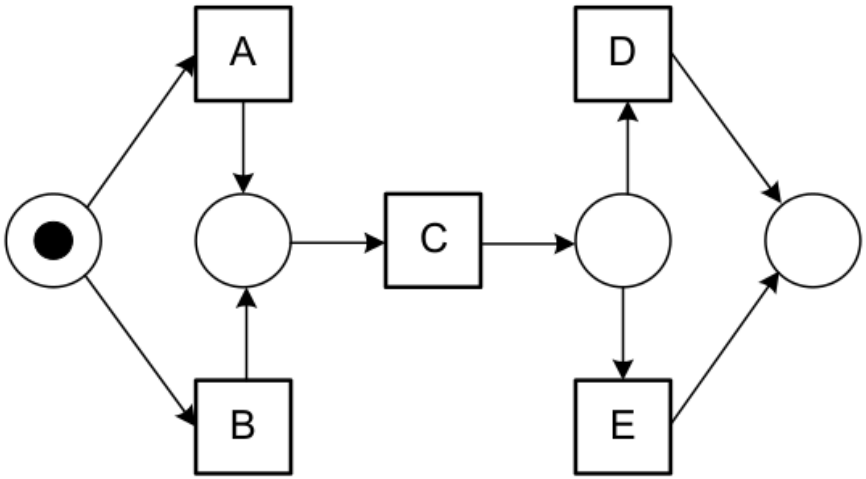




What is the best model? (3)

ACD	99
ACE	2
BCE	85
BCD	3



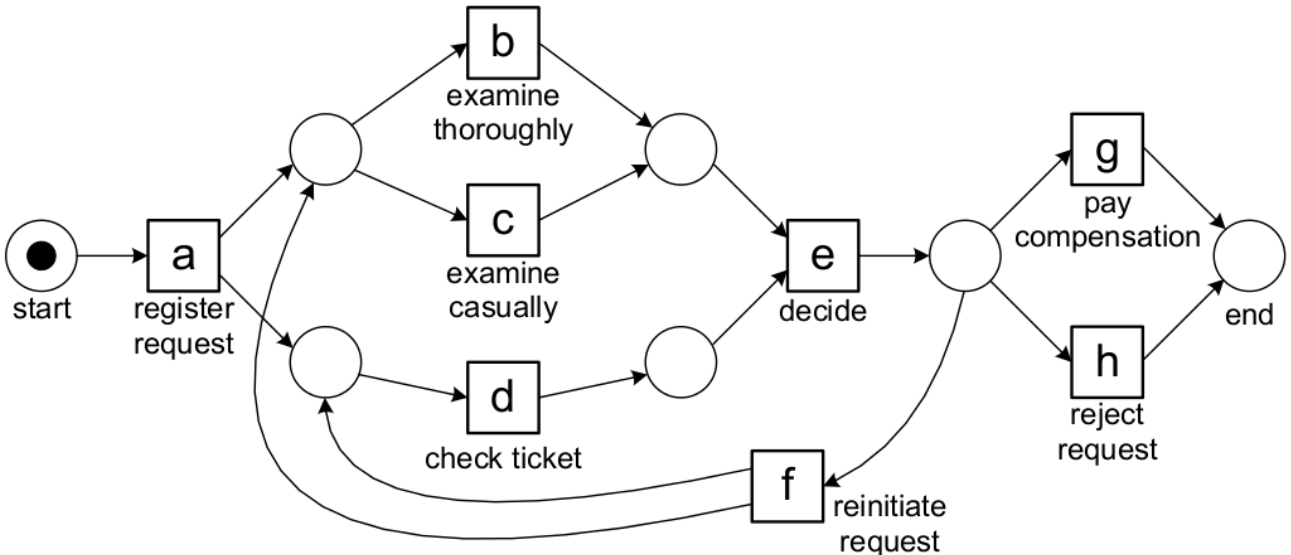


Four models, one log (1)

Universiteit Leiden

. . .

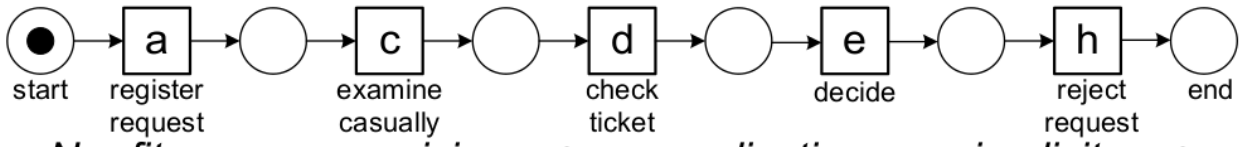
Model N_1



N_1 : fitness = +, precision = +, generalization = +, simplicity = +

#	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	

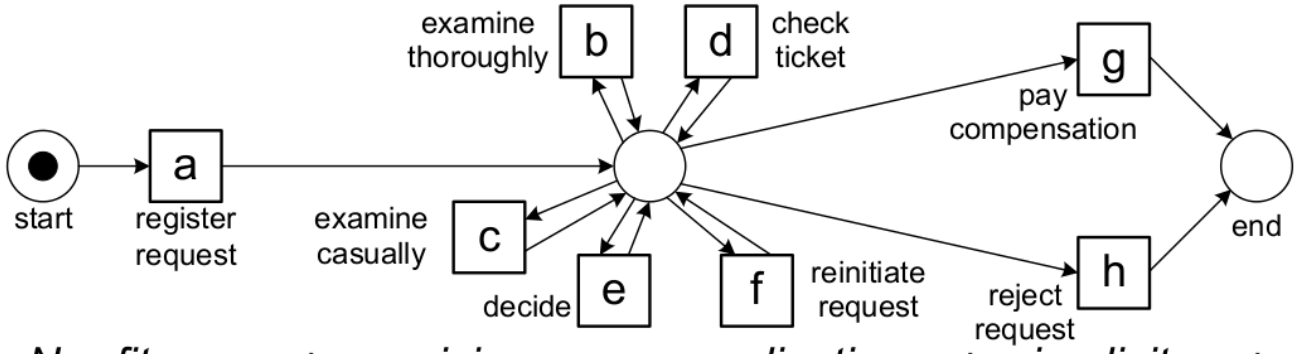
Model N_2



N_2 : fitness = -, precision = +, generalization = -, simplicity = +

#	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	

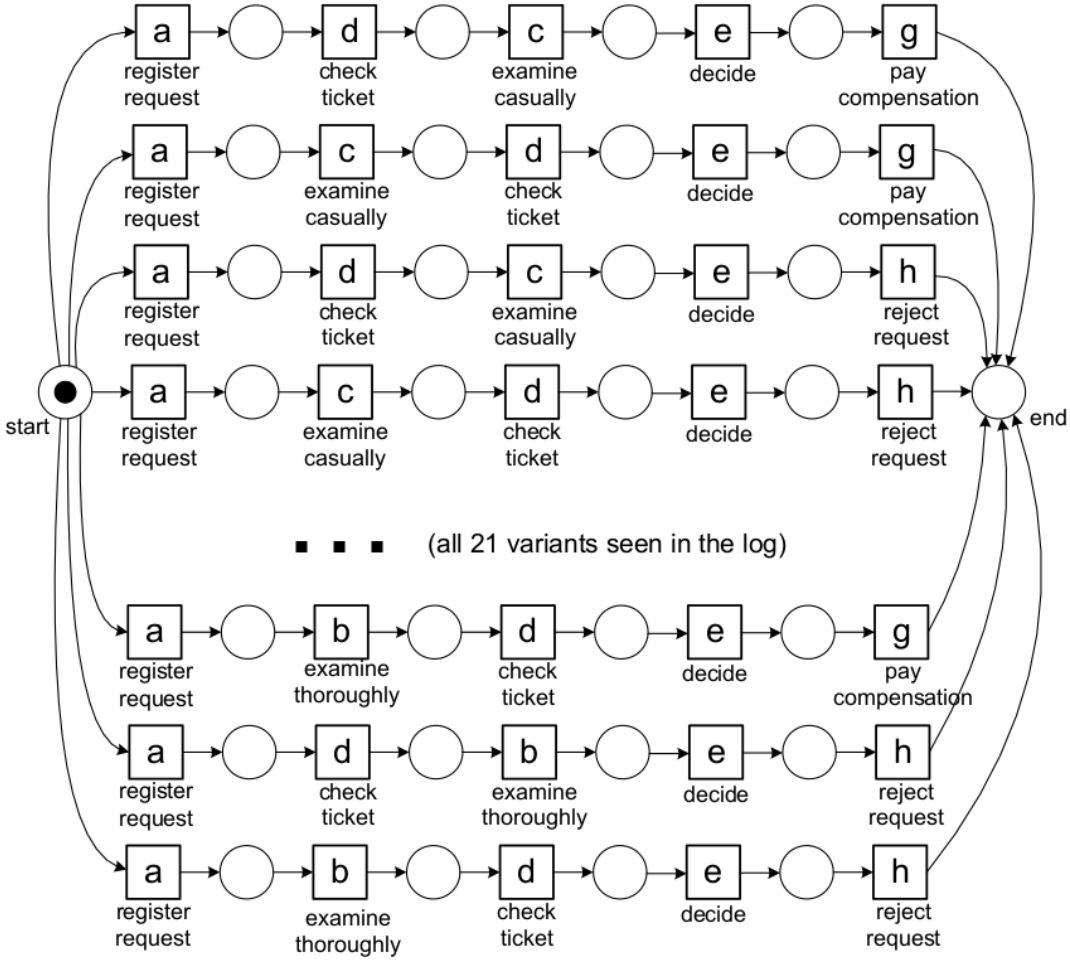
Model N_3



N_3 : fitness = +, precision = -, generalization = +, simplicity = +

#	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	

Model N_4



N_4 : fitness = +, precision = +, generalization = -, simplicity = -

#	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	

Why is process mining so difficult?

Universiteit Leiden

- There are **no negative examples**
- Due to concurrency, loops, and choices the search space has a complex structure and the log typically contains only a **fraction of all possible behaviors**
- There is **no clear relation between the size of a model and its behavior**: a smaller model may generate more or less behavior although classical analysis and evaluation methods typically assume some monotonicity property
- "Creating a 2D slice of a 3D reality"

Second example (1)

$$L_5 = [\langle a,b,e,f\rangle^2, \langle a,b,e,c,d,b,f\rangle^3, \langle a,b,c,e,d,b,f\rangle^2,$$
$$\langle a,b,c,d,e,b,f\rangle^4, \langle a,e,b,c,d,b,f\rangle^3]$$

Second example (1)

$$L_5 = [\langle a,b,e,f\rangle^2, \langle a,b,e,c,d,b,f\rangle^3, \langle a,b,c,e,d,b,f\rangle^2,$$
$$\langle a,b,c,d,e,b,f\rangle^4, \langle a,e,b,c,d,b,f\rangle^3]$$

- Create footprint
- Apply α-algorithm
- Draw Petri net

Second example (2)

$$L_5 = [\langle a,b,e,f\rangle^2, \langle a,b,e,c,d,b,f\rangle^3, \langle a,b,c,e,d,b,f\rangle^2, \langle a,b,c,d,e,b,f\rangle^4, \langle a,e,b,c,d,b,f\rangle^3]$$

	a	b	c	d	e	f
a	#	$\rightarrow$	#	#	$\rightarrow$	#
b	$\leftarrow$	#	$\rightarrow$	$\leftarrow$	$\|\|$	$\rightarrow$
c	#	$\leftarrow$	#	$\rightarrow$	$\|\|$	#
d	#	$\rightarrow$	$\leftarrow$	#	$\|\|$	#
e	$\leftarrow$	$\|\|$	$\|\|$	$\|\|$	#	$\rightarrow$
f	#	$\leftarrow$	#	#	$\leftarrow$	#

- **Create footprint**
- Apply α-algorithm
- Draw Petri net

Second example (3)

$$T_L = \{a,b,c,d,e,f\}$$

$$T_I = \{a\}$$

$$T_I = \{f\}$$

$$X_L = \{(\{a\},\{b\}),(\{a\},\{e\}),(\{b\},\{c\}),(\{b\},\{f\}),(\{c\},\{d\}),$$
$$(\{d\},\{b\}),(\{e\},\{f\}),(\{a,d\},\{b\}),(\{b\},\{c,f\})\}$$

$$Y_L = \{(\{a\},\{e\}),(\{c\},\{d\}),(\{e\},\{f\}),(\{a,d\},\{b\}),(\{b\},\{c,f\})\}$$

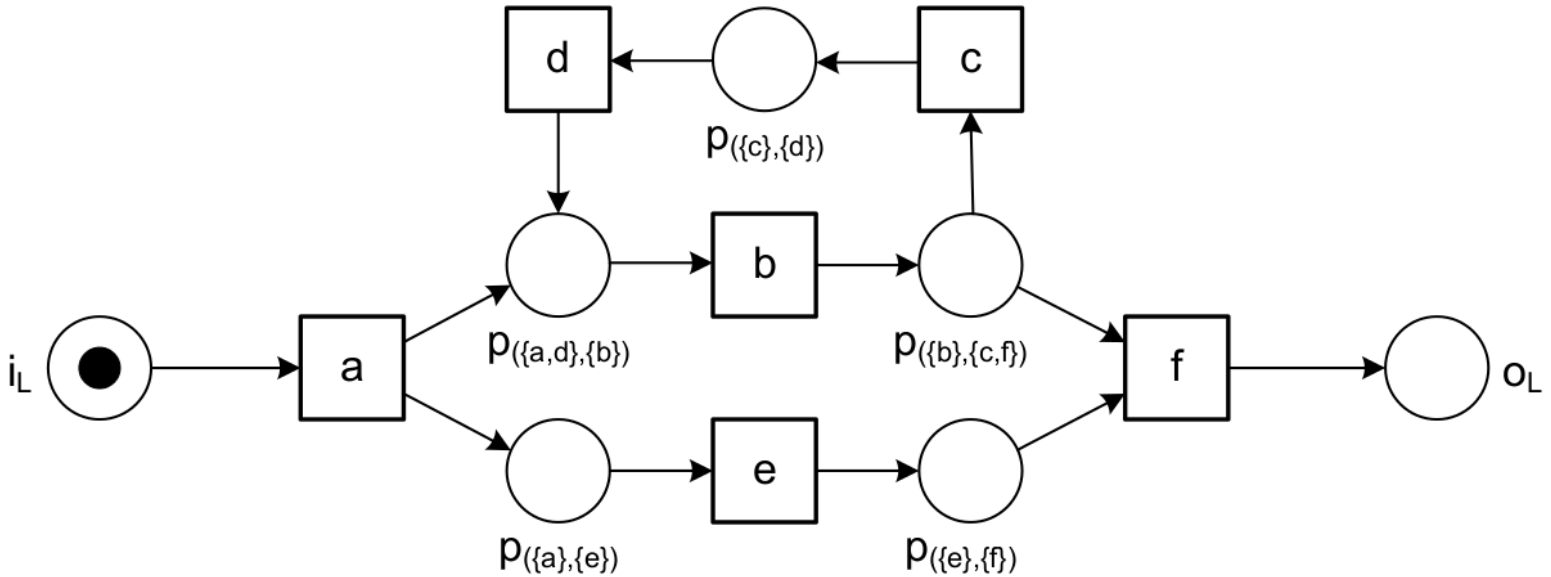
$$P_L = \{p_{(\{a\},\{e\})},p_{(\{c\},\{d\})},p_{(\{e\},\{f\})},p_{(\{a,d\},\{b\})},p_{(\{b\},\{c,f\})},i_L,o_L\}$$

$$F_L = \{(a,p_{(\{a\},\{e\})}),(p_{(\{a\},\{e\})},e),(c,p_{(\{c\},\{d\})}),(p_{(\{c\},\{d\})},d),$$
$$(e,p_{(\{e\},\{f\})}),(p_{(\{e\},\{f\})},f),(a,p_{(\{a,d\},\{b\})}),(d,p_{(\{a,d\},\{b\})}),$$
$$(p_{(\{a,d\},\{b\})},b),(b,p_{(\{b\},\{c,f\})}),(p_{(\{b\},\{c,f\})},c),(p_{(\{b\},\{c,f\})},f),$$
$$(i_L,a),(f,o_L)\}$$

$$\alpha(L) = (P_L,T_L,F_L)$$

Second example (4)

$$L_5 = [\langle a,b,e,f\rangle^2, \langle a,b,e,c,d,b,f\rangle^3, \langle a,b,c,e,d,b,f\rangle^2, \langle a,b,c,d,e,b,f\rangle^4, \langle a,e,b,c,d,b,f\rangle^3]$$



$$X_L = \{(\{a\},\{b\}),(\{a\},\{e\}),(\{b\},\{c\}),(\{b\},\{f\}),(\{c\},\{d\}),(\{d\},\{b\}),(\{e\},\{f\}),(\{a,d\},\{b\}),(\{b\},\{c,f\})\}$$

$$Y_L = \{(\{a\},\{e\}),(\{c\},\{d\}),(\{e\},\{f\}),(\{a,d\},\{b\}),(\{b\},\{c,f\})\}$$

Lab session May 4

- Install ProM
- Master ProM by completing the relevant tutorials
- Make serious progress with Assignment 3
- Ask questions

Credits

Lecture partially based on (slides of the (previous edition of the)) course book: W. van der Aalst, *Process Mining: Data Science in Action*, 2nd edition, Springer, 2016.