Eksploracja logów procesów

Process mining

Eksploracja logów procesów

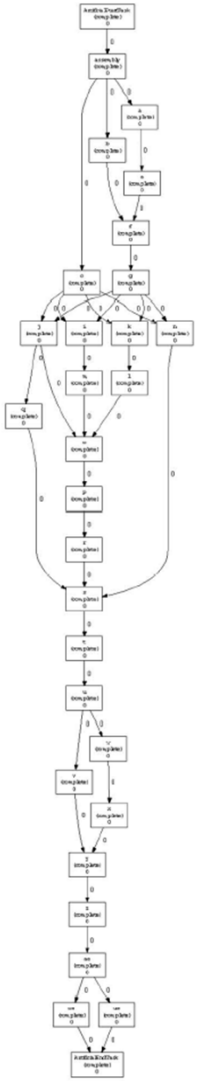
Celem eksploracji logów procesów biznesowych jest:

- Odkrywanie modelu procesów biznesowych
- Analiza procesów biznesowych
- Ulepszanie procesów biznesowych

Złożoność rzeczywistych procesów

Modele procesów,
a rzeczywiste wystąpienia procesów

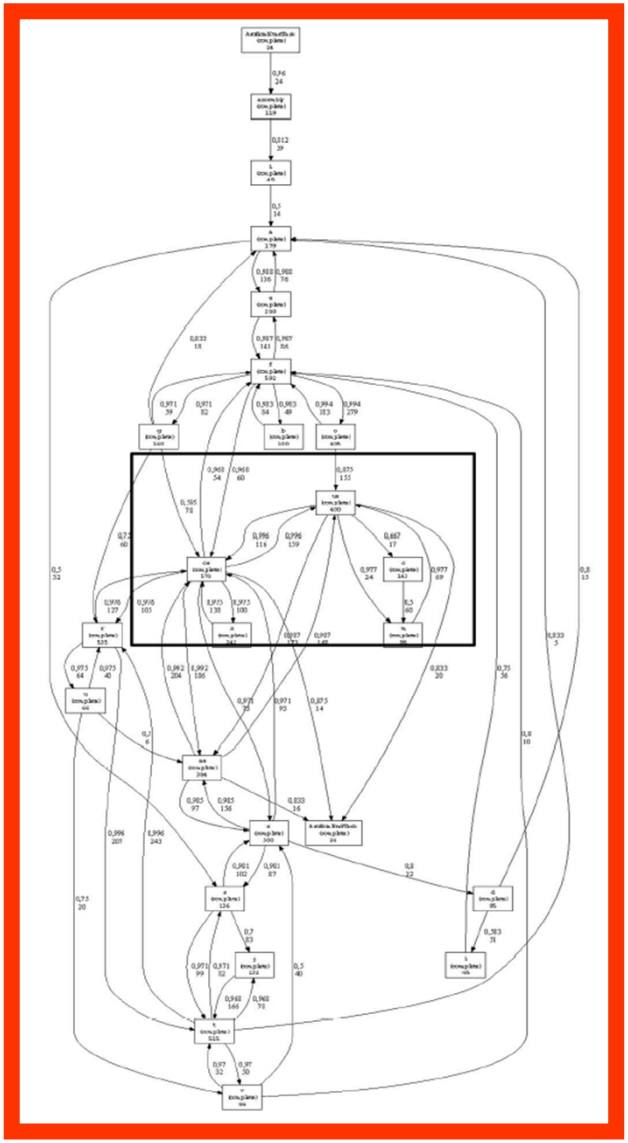
model



rzeczywistość

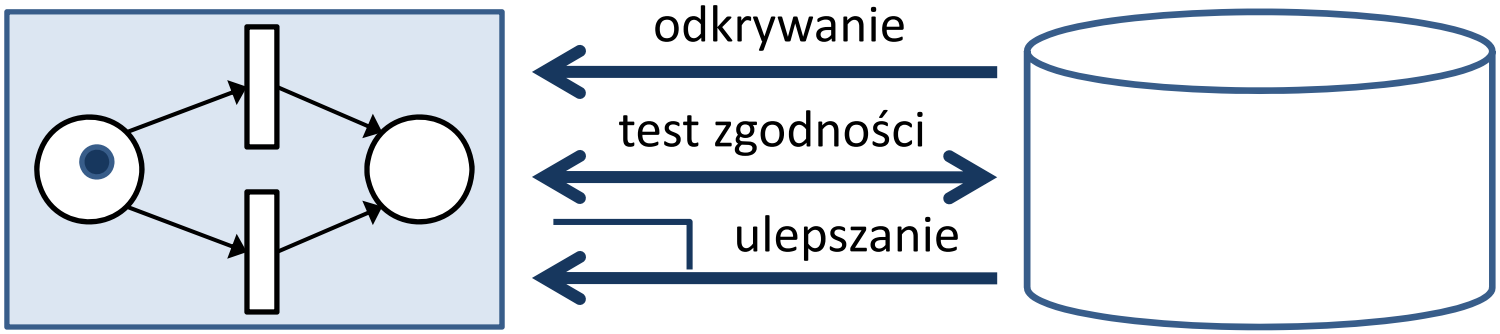
Wystąpienia zgodne z modelem: 37,5%

Wystąpienia niezgodne z modelem: 62,5%



Typy eksploracji procesów

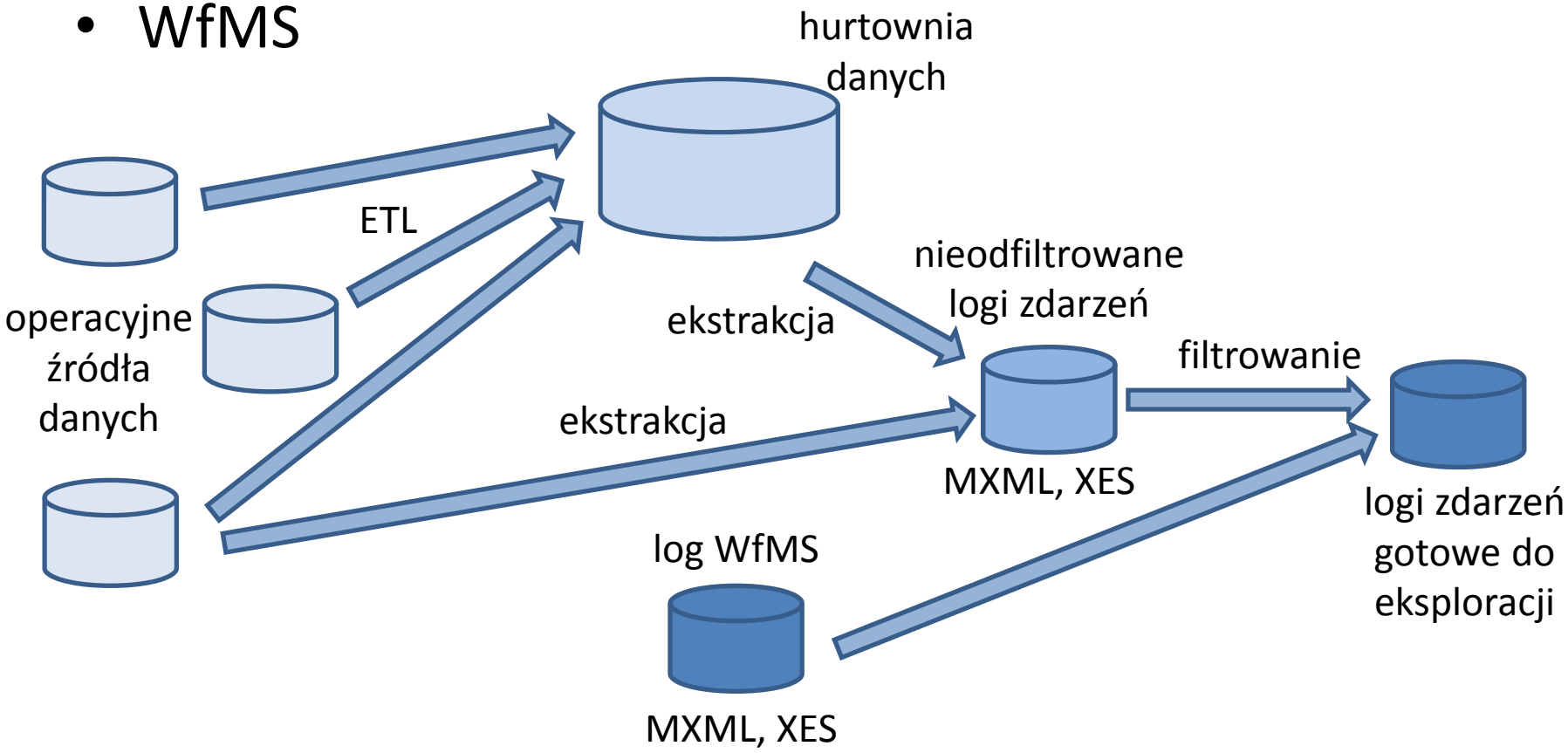
- **Odkrywanie** – dla procesów, które nie posiadają zdefiniowanych modeli, model może być zdefiniowany jako wynik eksploracji zebranych logów; np. ProM tworzy automatycznie model sieci Petriego
- **Test zgodności** – modelu teoretycznego z rzeczywistym przebiegiem procesów wynikającym z logów procesów
- **Ulepszanie** modelu teoretycznego w oparciu o wiedzę wynikającą z logów procesów



Pozyskiwanie informacji do logów

Źródła informacji o przebiegu procesów

- operacyjne bazy danych
- hurtownie danych
- WfMS



Informacje w logach zdarzeń

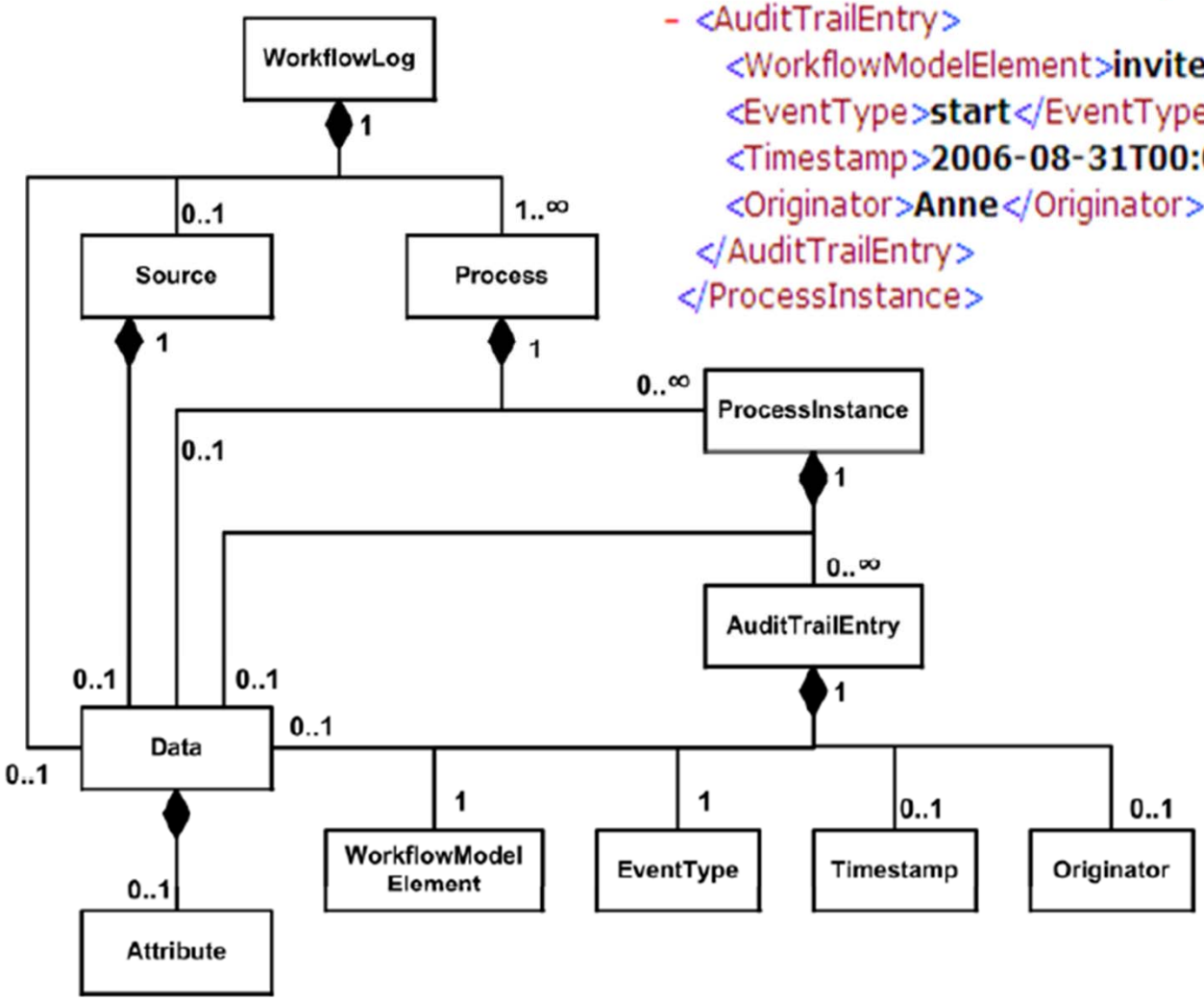
Trzy podstawowe perspektywy analizy danych

- Procesowa – jak wykonywane są procesy
- Przypadków – co dzieje się w firmie
- Organizacyjna - kto uczestniczył w realizacji procesu

Logi rejestrują **wystąpienia procesów. Wystąpienia** są uporządkowanymi sekwencjami zdarzeń. Zdarzenia są opisane przez atrybuty, np. czas wystąpienia, nazwa działania, koszt.

Id wystąpienia	Id zdarzenia	Działanie	Wykonawca	Czas
107	A	Rejestracja zamówienia	Nowak	7-1-2008 9:45
108	B	Wystawienie faktury	Kowalski	7-1-2008 10:05
109	A	Rejestracja zamówienia	Tarzan	7-1-2008 12:17
110	C	Wysłanie towaru	Buła	7-1-2008 14:48
107	B	Wystawienie faktury	Kowalski	8-1-2008 8:15
110	E	Potwierdzenie odbioru	Nowak	8-1-2008 9:32

Zawartość logu MXML

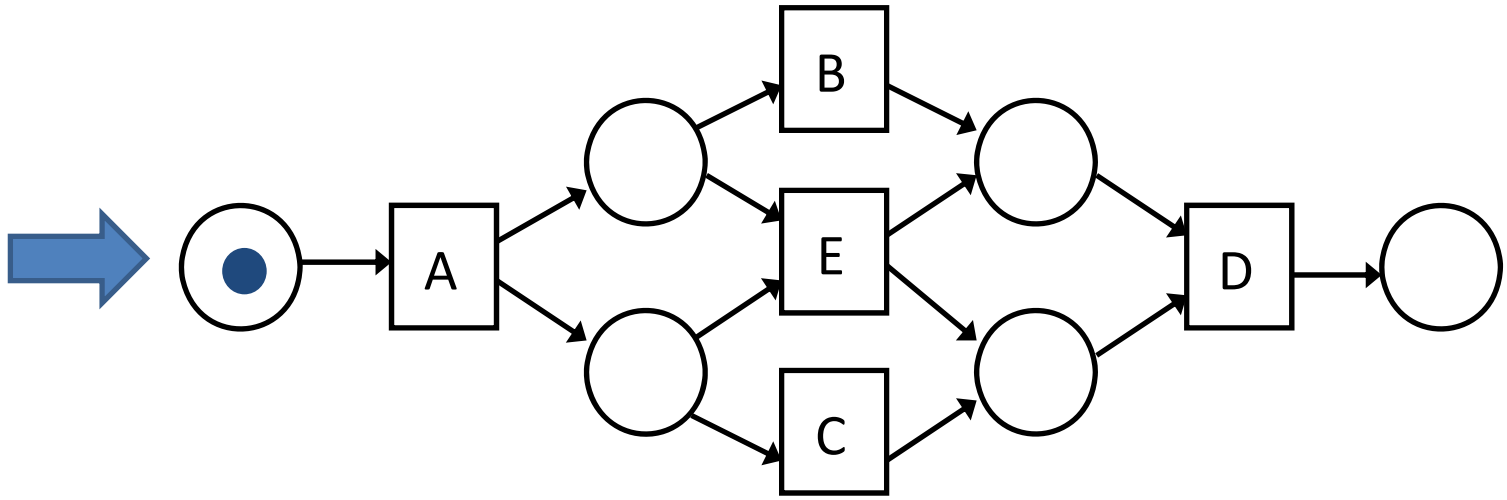



```
<ProcessInstance id="52" description="">
- <AuditTrailEntry>
    <WorkflowModelElement>invite reviewers</WorkflowModelElement>
    <EventType>start</EventType>
    <Timestamp>2006-08-31T00:00:00.000+01:00</Timestamp>
    <Originator>Anne</Originator>
  </AuditTrailEntry>
 </ProcessInstance>
```

Odkrywanie modelu procesów

Odtwarzanie model procesu na podstawie zarejestrowanych sekwencji zdarzeń

ABCD
ACBD
AED
ABCD
ABCD
AED
ACBD



Kryteria jakości procesu odkrywania

- **Trafność**: Odkryty model powinien przewidywać możliwość wykonania wszystkich wystąpień procesów zapamiętanych w logu.
- **Precyzja**: Odkryty model nie powinien wspierać wystąpień procesu niepotwierdzonych przez informacje w logu (underfitting).
- **Generalizacja**: Odkryty model powinien generalizować przykłady wystąpień zarejestrowanych w logu (overfitting).
- **Minimalność**: Odkryty model powinien być tak prosty jak to możliwe.

Wystąpienie	Zadanie
case 1	task A
case 2	task A
case 3	task A
case 3	task B
case 1	task B
case 1	task C
case 2	task C
case 4	task A
case 2	task B
case 2	task D
case 5	task E
case 4	task B
case 1	task D
case 3	task C
case 3	task D
case 4	task C
case 5	task F
case 4	task D

Algorytm α

Odkrywanie sekwencji w logu

- Sekwencje są projekcjami zdarzeń należących do tego samego wystąpienia

case 1(Log) = ABCD
case 2(Log) = ACBD
…

ABCD - cases: 1, 2, 3; razem (3)
ACBD – case: 4; razem (1)
EF – case: 5; razem (1)

Algorytm α

Relacje występujące między zdarzeniami

- **Bezpośrednie następstwo**
  - x > y jeżeli w logu istnieje sekwencja, w której zdarzenie x występuje bezpośrednio przed zdarzeniem y.
- **Przyczynowość**
  - x → y jeżeli istnieją sekwencje, w których występuje x > y, a nie ma sekwencji, w których występuje y > x.
- **Współbieżność**
  - x||y jeżeli istnieją sekwencje, w których występuje x > y i równocześnie istnieją wystąpienia procesów, w których występuje y > x
- **Wybór**
  - x#y jeżeli w żadnej sekwencji nie ma przypadków x>y, ani y>x.

Algorytm α - relacje między zdarzeniami

Odkrywanie relacji między zdarzeniami

- Bezpośrednie następstwo:

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

- Przyczynowość:

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

- Współbieżność:

  B||C

- Wybór:

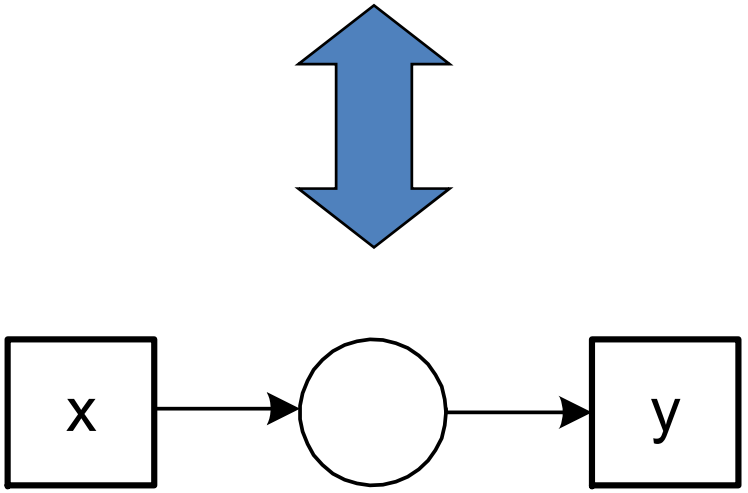
  A#D, A#E, A#F, B#E, B#F, C#E, C#F, D#E, D#F

- **ABCD**
- **ACBD**
- **EF**

Algorytm α - reguły transformacji

Wzorzec sekwencji

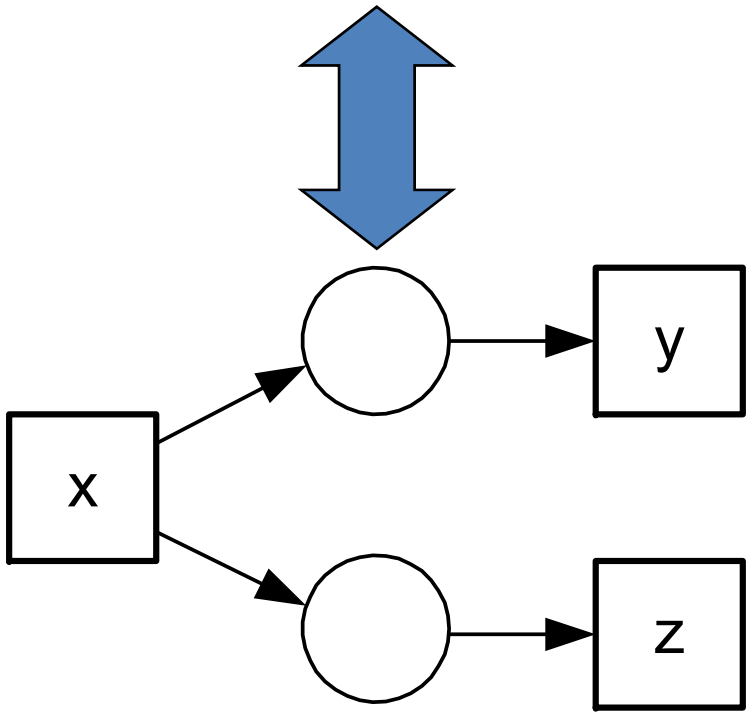
$x \rightarrow y$



Algorytm α - reguły transformacji

wzorzec rozwidlenia AND

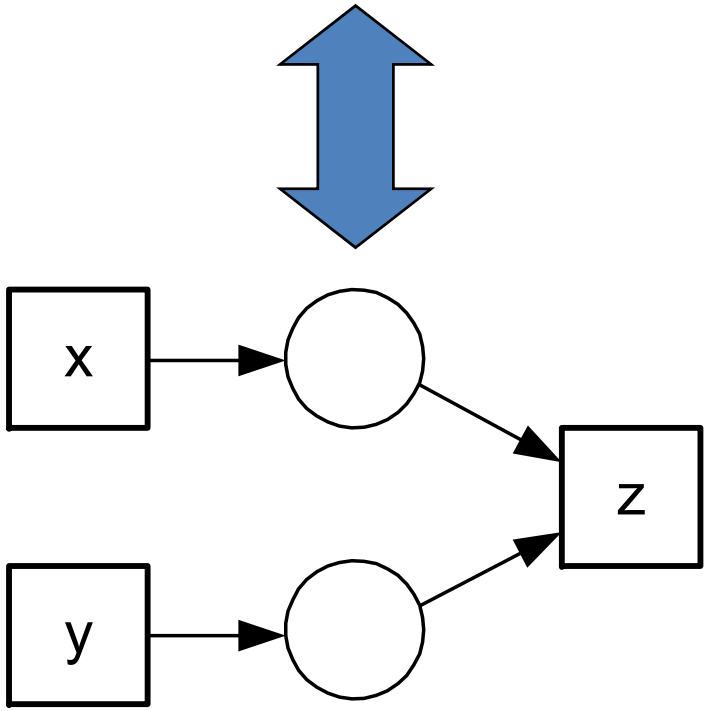
$x \rightarrow y$, $x \rightarrow z$, and $y || z$



Algorytm α - reguły transformacji

wzorzec połączenia AND

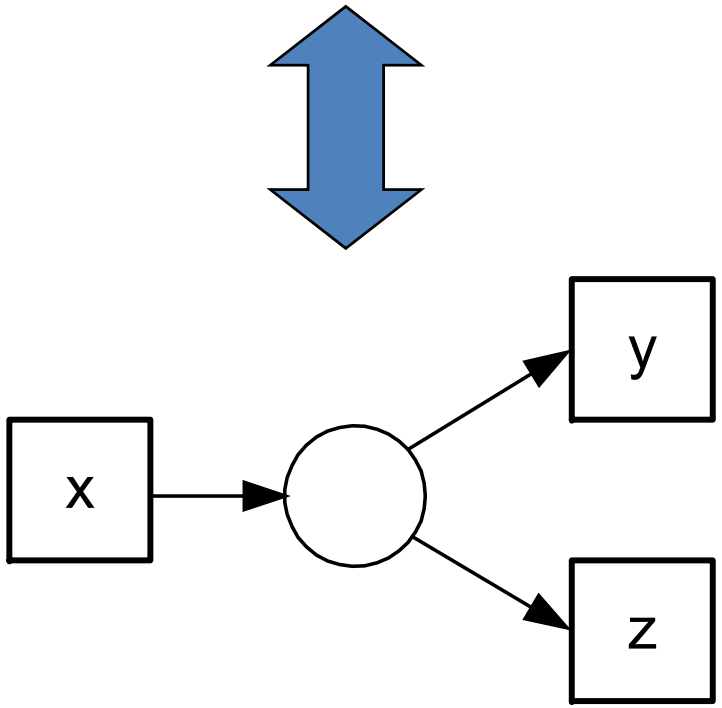
$x \rightarrow z$, $y \rightarrow z$, and $x||y$



Algorytm α - reguły transformacji

wzorzec rozwidlenia XOR

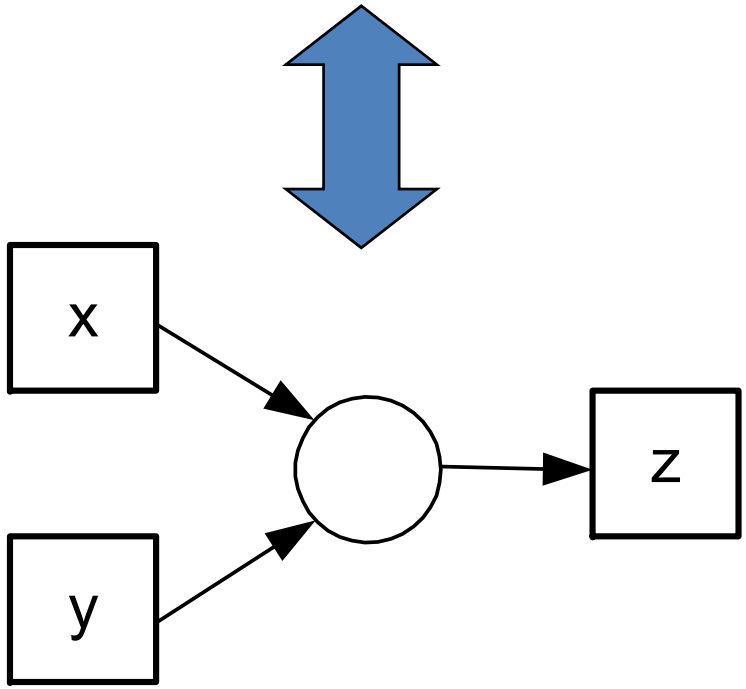
$x \rightarrow y$, $x \rightarrow z$, and $y \# z$



Algorytm α - reguły transformacji

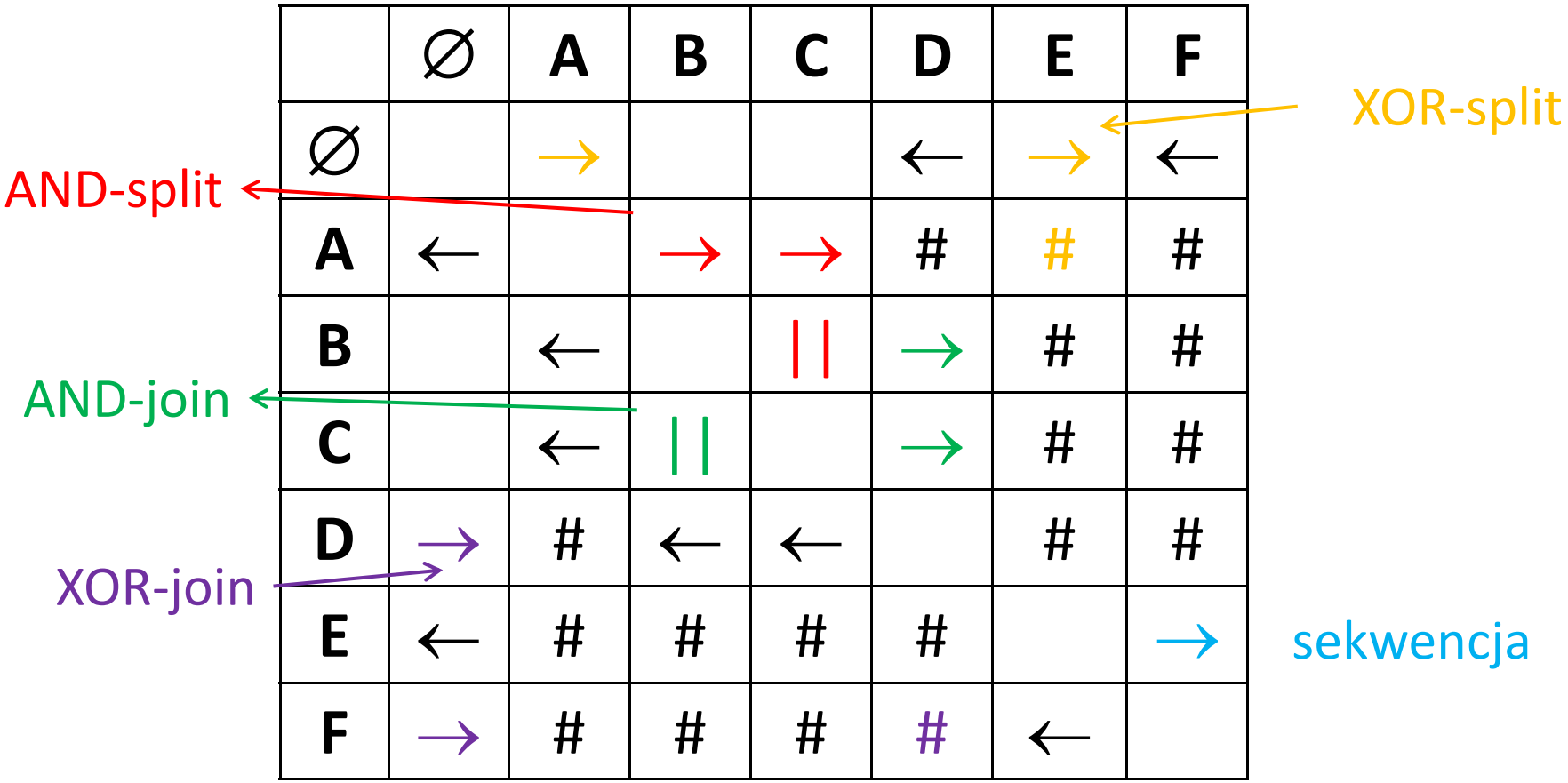
wzorzec połączenia XOR

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



Algorytm α - konstrukcja modelu

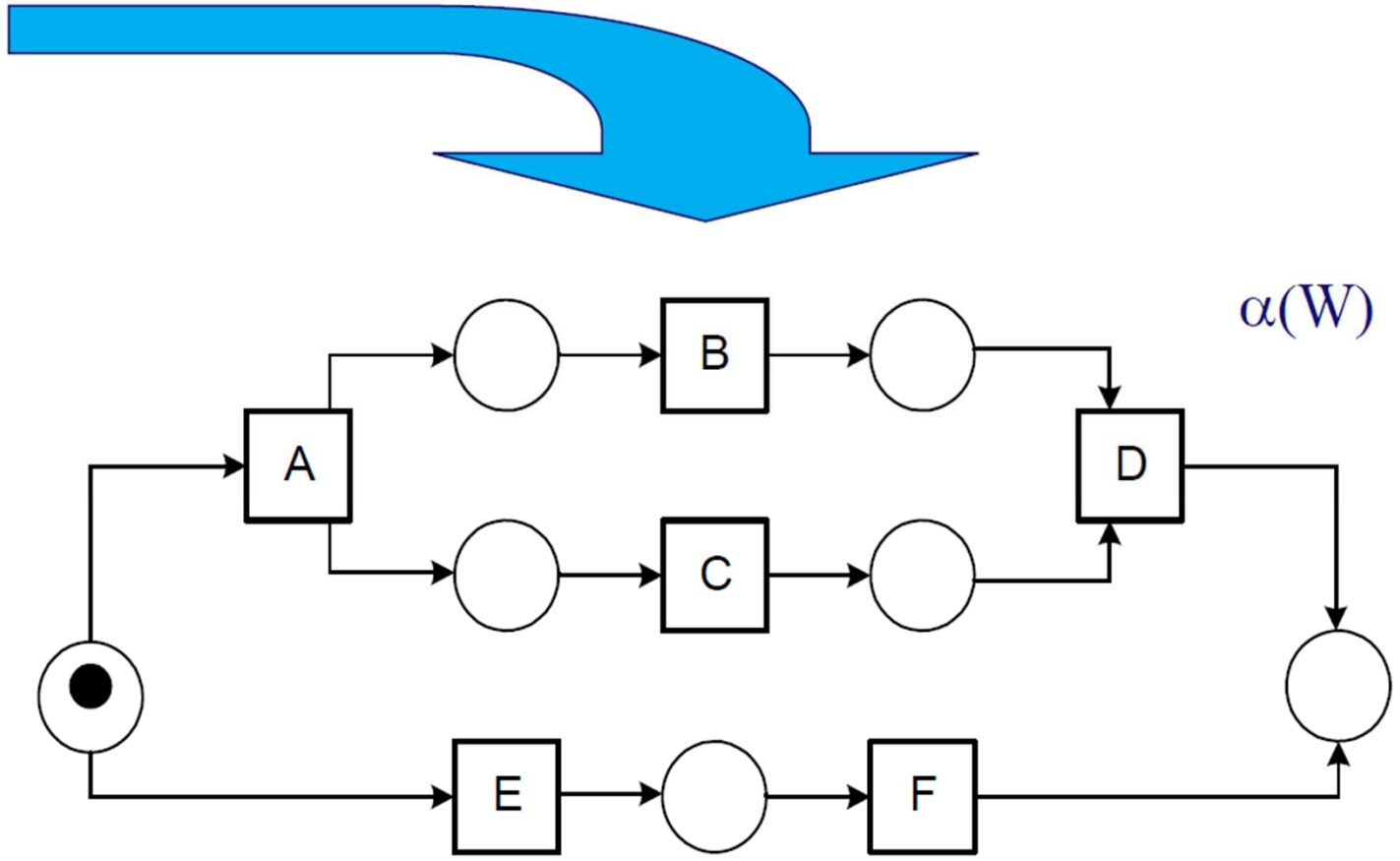
Matryca zależności zdarzeń



	∅	A	B	C	D	E	F
∅		→			←	→	←
A	←		→	→	#	#	#
B		←		||	→	#	#
C		←	||		→	#	#
D	→	#	←	←		#	#
E	←	#	#	#	#		→
F	→	#	#	#	#	←	

Case	Task
case 1	task A
case 2	task A
case 3	task A
case 3	task B
case 1	task B
case 1	task C
case 2	task C
case 4	task A
case 2	task B
case 2	task D
case 5	task E
case 4	task C
case 1	task D
case 3	task C
case 3	task D
case 4	task B
case 5	task F
case 4	task D

Wynik odkrywania modelu



Formalna definicja algorytmu α

Niech W będzie logiem procesu zdefiniowanym na zbiorze elementarnych zadań T, σ jest dowolną sekwencją w tym logu.

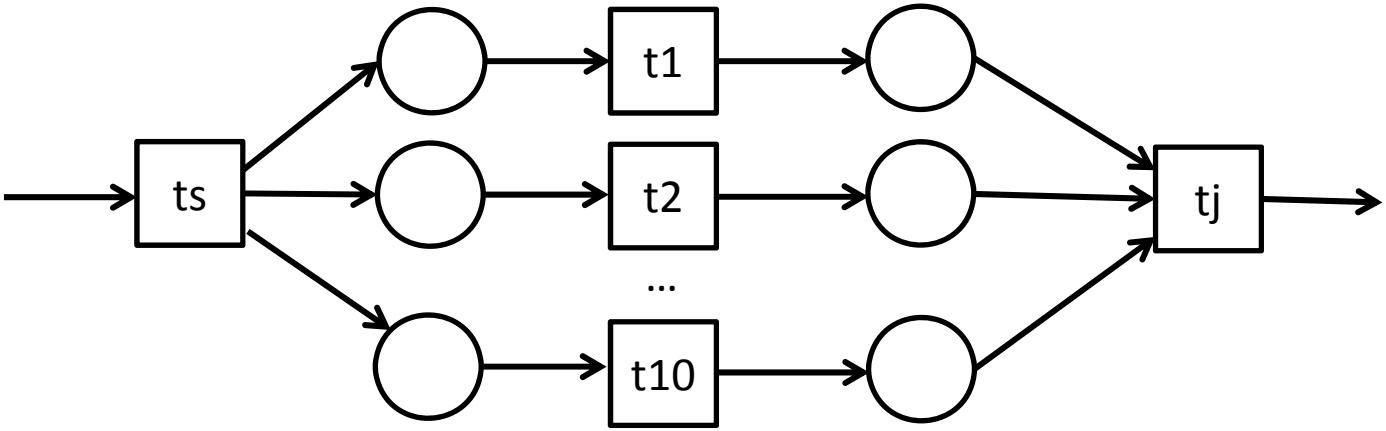
$\alpha(W)$ jest zdefiniowane następująco:

1. $T_W = \{ t \in T \mid \exists_{\sigma \in W}\, t \in \sigma \}$,
2. $T_I = \{ t \in T \mid \exists_{\sigma \in W}\, t = \mathit{first}(\sigma) \}$,
3. $T_O = \{ t \in T \mid \exists_{\sigma \in W}\, t = \mathit{last}(\sigma) \}$,
4. $X_W = \{ (A,B) \mid A \subseteq T_W \wedge B \subseteq T_W \wedge \forall_{a \in A} \forall_{b \in B}\, a \rightarrow_W b \wedge \forall_{a1,a2 \in A}\, a_1 \#_W a_2 \wedge \forall_{b1,b2 \in B}\, b_1 \#_W b_2 \}$,
5. $Y_W = \{ (A,B) \in X \mid \forall_{(A',B') \in X} A \subseteq A' \wedge B \subseteq B' \Rightarrow (A,B) = (A',B') \}$,
6. $P_W = \{ p_{(A,B)} \mid (A,B) \in Y_W \} \cup \{i_W, o_W\}$,
7. $F_W = \{ (a,p_{(A,B)}) \mid (A,B) \in Y_W \wedge a \in A \} \cup \{ (p_{(A,B)},b) \mid (A,B) \in Y_W \wedge b \in B \} \cup \{ (i_W,t) \mid t \in T_I \} \cup \{ (t,o_W) \mid t \in T_O \}$, and
8. $\alpha(W) = (P_W, T_W, F_W)$.

Ograniczenia algorytmu

- **Kompletność logów** – Poprawny model procesu możemy uzyskać, jedynie jeżeli log zawiera ślady wszystkich możliwych wystąpień procesu.

  Przykładowo, log pozwalający odtworzyć poniższy proces musi zawierać 10! różnych sekwencji.



- **Błędne zapisy w logu** – na poprawność modelu będą miały wpływ błędne informacje zapisane w logu. Rzadko występujące przypadki będą miały taki sam wpływ na kształt modelu jak częste przypadki.

Eksploracja zależności organizacyjnych

Sekwencje znalezione w logu:

id sekwencji	**zawartość sekwencji**
1	Nowak(A), Kowalski(B), Tarzan(C), Nowak(D)
2	Nowak(A), Kowalski(B), Tarzan(C)
3	Buła(A), Frąckowiak(B), Tarzan(C)

Aktywność wykonawców:

Wykonawca	**A**	**B**	**C**	**D**
Nowak	78%			12%
Kowalski		81%		
Buła	22%			
…				

Sieć socjalna łącząca wykonawców

Sieć socjalna zbudowana na podstawie przekazywania pracy w ramach procesów