



Gestion de Processus : Découverte de Processus

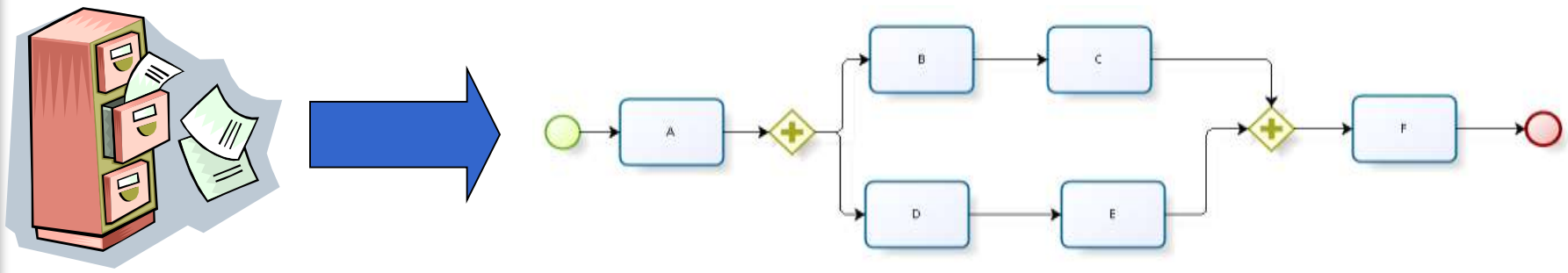
E. Andonoff

MF 206

Eric.Andonoff@ut-capitole.fr

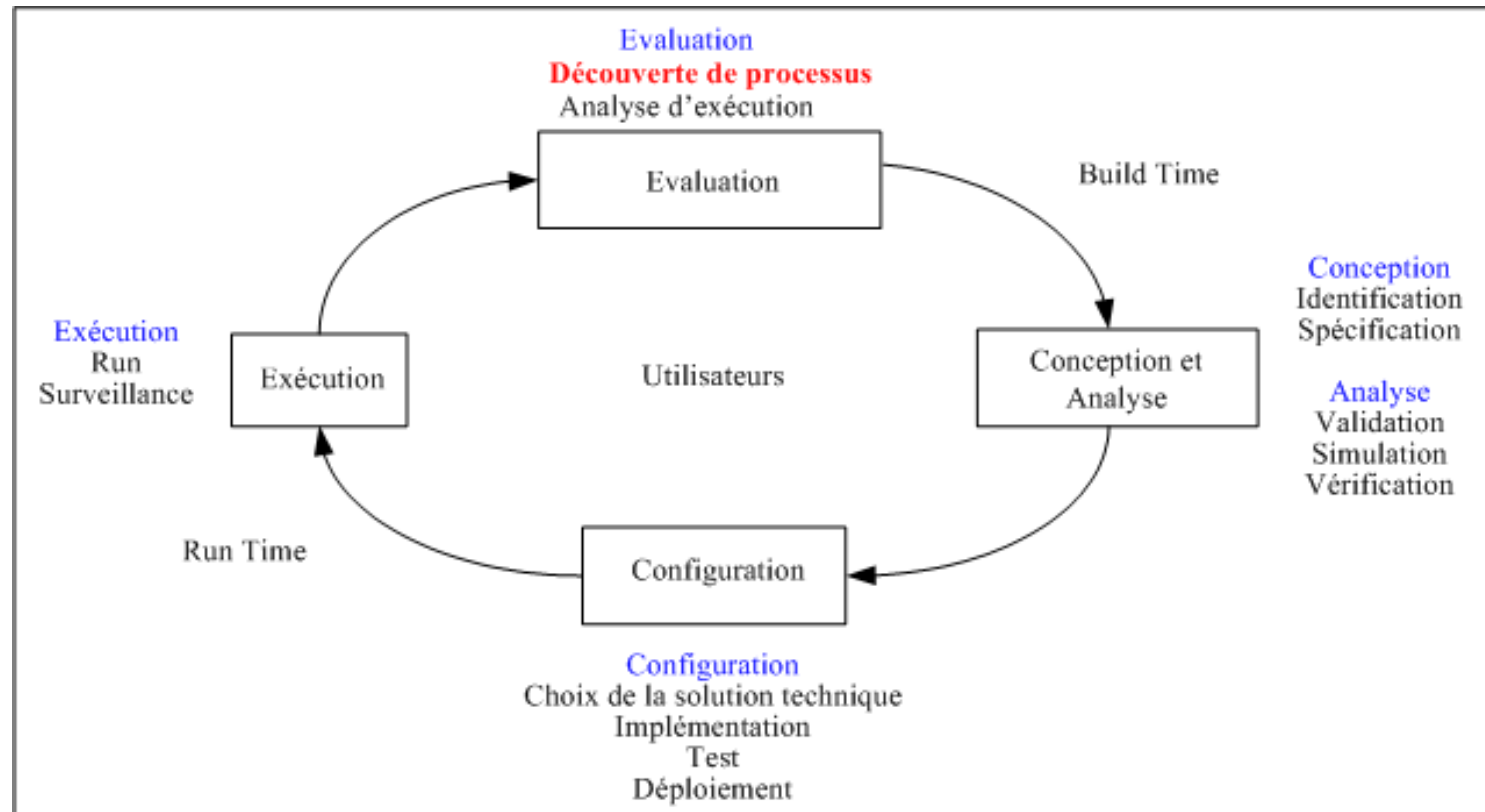
Introduction

- Découverte de schémas de processus
 - à partir d'instances de processus décrites dans des fichiers Log
 - Découverte du
 - **Modèle Comportemental** : activités, enchainement d'activités
 - **Modèle Organisationnel** : acteurs, rôles (réseaux sociaux, structures organisationnelles)
 - **Modèle Informationnel**



Introduction

- Découverte versus cycle de vie des processus



Introduction

- Découverte de processus : pourquoi ?
 - Modélisation : découverte d'un premier schéma de processus
 - Amélioration
 - Analyse Delta : se passe-t-il ce qui a été prévu ?
 - Analyse des performances : comment peut on améliorer le schéma du processus ?

Introduction

- Découverte de processus
 - Un site :
 - www.processmining.org
 - Un logiciel libre :
 - ProM : www.promtools.org/prom6/
 - Un livre :
 - W. van der Aalst, Process Mining: Discovery, Conformance and Enhancement of Business Processes. Springer Verlag, 2011

Découverte du comportement

■ Algorithme Alpha

— Entrée : fichier Log

- Contient uniquement les cas (instances de processus) terminés
- Numéro du cas, nom de l'activité (tâche)
- Informations supplémentaires : type d'événement, date, ressources et données

— Sortie : Réseau de Pétri représentant le comportement des cas considérés

— Principe :

- Identifier des relations entre activités
- Appliquer les 8 étapes de l'algorithme Alpha en exploitant les relations identifiées

Découverte du comportement

■ Algorithme Alpha

— Relation entre activités

- Succession directe

$x > y$ si et seulement si il existe un cas dans lequel x est directement suivi de y

- **Causalité**

$x \rightarrow y$ si et seulement si $x > y$ et not $y > x$

- **Parallélisme**

$x || y$ si et seulement si $x > y$ et $y > x$

- **Indépendance**

$x \# y$ si et seulement si not $x > y$ et not $y > x$

Découverte du comportement

- Algorithme Alpha
 - Identification des relations entre activités
 - Identifier les mots
ABCD, ACBD, EF
 - Analyser les mots pour identifier les relations de succession directe ($>$)
 - Dédire les relations de causalité (\rightarrow) et de parallélisme ($||$)
 - Dédire les relations d'indépendance ($\#$)

Découverte du comportement

■ Algorithme Alpha

- Identification des mots
ABCD, ACBD, EF
- Identification des
successions directes
- Dédution des causalités
et parallélisme
- Dédution des
indépendances

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

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

A→B
A→C
B→D
C→D
E→F

B||C
C||B

```
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
```

Découverte du comportement

- Algorithme Alpha
 - Représentation des relations

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

$A \rightarrow B$

$A \rightarrow C$

$B \rightarrow D$

$C \rightarrow D$

$E \rightarrow F$

$B || C$

$C || B$

$A \# E, A \# F, A \# D$

$B \# E, B \# F$

$C \# E, C \# F$

$D \# E, D \# F, \dots$

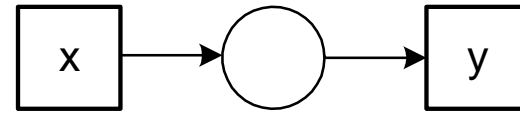
Découverte du comportement

■ Algorithme Alpha

— Intuitions

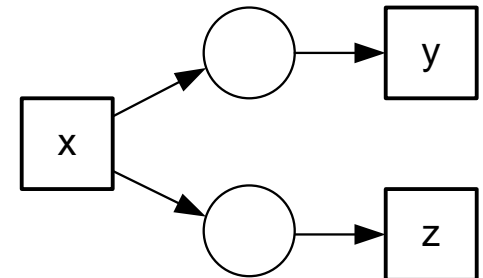
- Causalité seule

$x \rightarrow y$



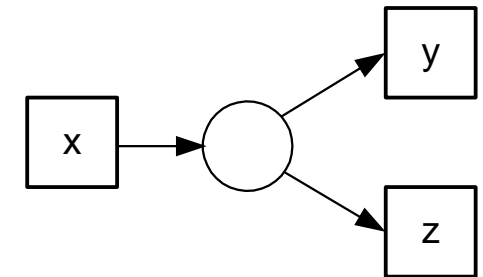
- Causalité et Parallélisme (1)

$x \rightarrow y$, $x \rightarrow z$, and $y \perp\!\!\!\perp z$



- Causalité et indépendance(1)

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



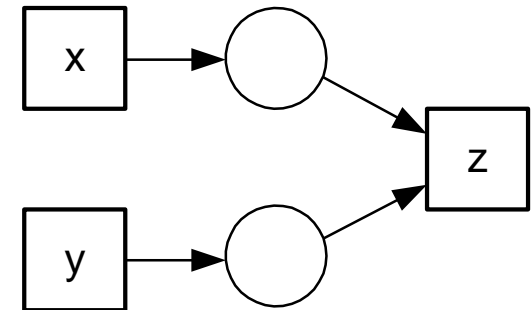
Découverte du comportement

■ Algorithme Alpha

— Intuitions

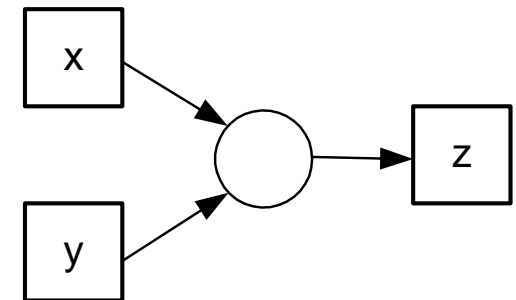
- Causalité et Parallélisme (2)

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



- Causalité et indépendance (2)

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



Découverte du comportement

■ Algorithme Alpha

Soit W un fichier log sur T (ensemble des tâches).

1. $T_W = \{ t \in T \mid \exists_{\sigma \in W} t \in \sigma \}$, ensemble des tâches
2. $T_I = \{ t \in T \mid \exists_{\sigma \in W} t = \text{first}(\sigma) \}$, tâches initiales
3. $T_O = \{ t \in T \mid \exists_{\sigma \in W} t = \text{last}(\sigma) \}$, tâches terminales
4. $X_W = \{ (A,B) \mid A \subseteq T_W \wedge B \subseteq T_W \wedge \forall_{a \in A} \forall_{b \in B} a \rightarrow_W b \wedge \forall_{a_1, a_2 \in A} a_1 \#_W a_2 \wedge \forall_{b_1, b_2 \in B} b_1 \#_W b_2 \}$, relations de causalité
5. $Y_W = \{ (A,B) \in X \mid \forall_{(A',B') \in X} A \subseteq A' \wedge B \subseteq B' \Rightarrow (A,B) = (A',B') \}$, relations de causalité minimale
6. $P_W = \{ p_{(A,B)} \mid (A,B) \in Y_W \} \cup \{ i_W, o_W \}$, places reliant les transitions de A à B , + places de début et de fin
7. $F_W = \{ (a, p_{(A,B)}) \mid (A,B) \in Y_W \wedge a \in A \} \cup \{ (p_{(A,B)}, b) \mid (A,B) \in Y_W \wedge b \in B \} \cup \{ (i_W, t) \mid t \in T_I \} \cup \{ (t, o_W) \mid t \in T_O \}$, Arcs du réseau
8. $\alpha(W) = (P_W, T_W, F_W)$, réseau de Petri final

Découverte du comportement

■ Algorithme Alpha

— Relations entre activités

— Etapes

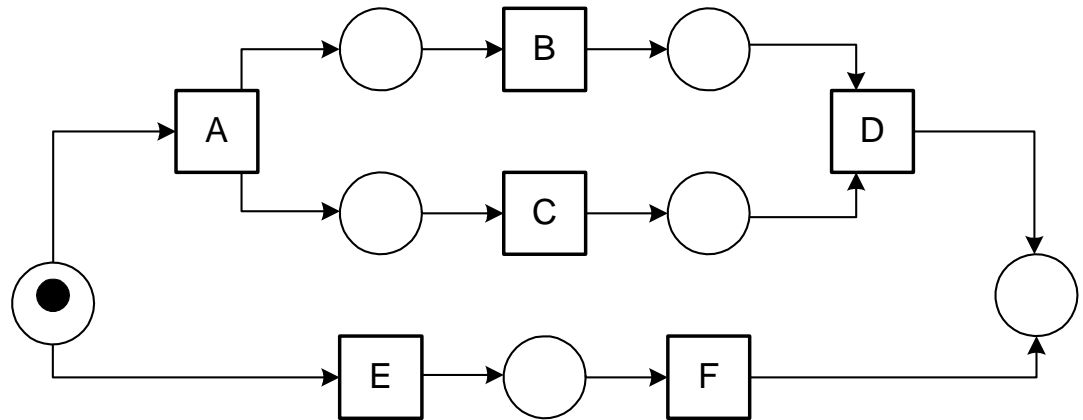
- $T = \{A, B, C, D, E, F\}$
- $I = \{A, E\}$
- $O = \{D, F\}$
- $X_w = \{(\{A\}, \{B\}), (\{A\}, \{C\}), (\{E\}, \{F\}), (\{B\}, \{D\}), (\{C\}, \{D\})\}$
- $Y_w = X_w$
- $P_w = \{P_{A,B}, P_{A,C}, P_{E,F}, P_{B,D}, P_{C,D}, I_w, O_w\}$
- $F_w = \{\text{arcs}\}$

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

Découverte du comportement

■ Algorithme Alpha

— Exemple 1



— Vérification des mots

- ABCD ? ok
- ACBD ? ok
- EF ? ok

Découverte du comportement

- Algorithme Alpha
 - Mots
ABCD, ACBD, AED
 - Relations entre activités

A>B
A>C
A>E
B>C
C>B
C>D
B>D
E>D

A→B
A→C
A→E
C→D
B→D
E→D

B||C
C||B

A#D
B#E
C#E

```
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 A
case 4 : task C
case 1 : task D
case 3 : task C
case 3 : task D
case 4 : task B
case 5 : task E
case 5 : task D
case 4 : task D
```


Découverte du comportement

■ Algorithme Alpha

– Relations entre activités

– Etapes

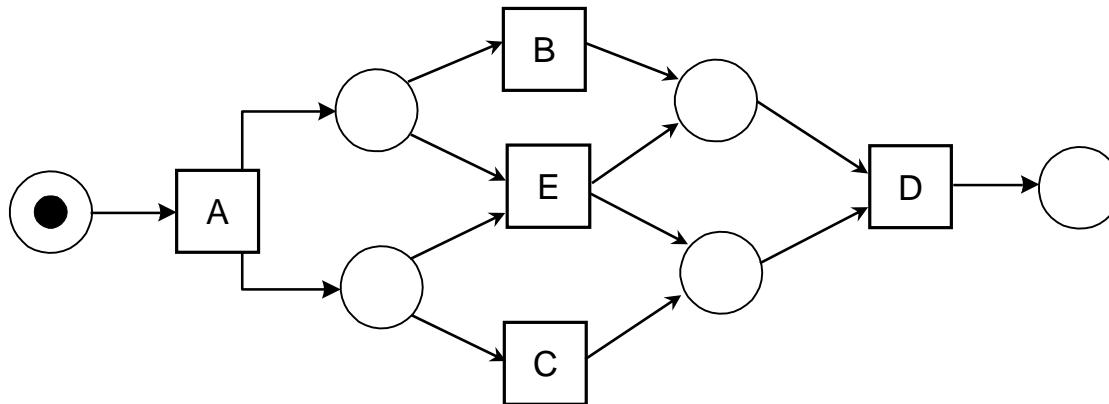
- $T = \{A, B, C, D, E\}$
- $I = \{A\}$
- $O = \{D\}$
- $X_w = \{(\{A\}, \{B\}), (\{A\}, \{C\}), (\{A\}, \{E\}), (\{B\}, \{D\}), (\{C\}, \{D\}), (\{E\}, \{D\}), (\{A\}, \{B, E\}), (\{A\}, \{C, E\}), (\{B, E\}, \{D\}), (\{C, E\}, \{D\})\}$
- $Y_w = \{(\{A\}, \{B, E\}), (\{A\}, \{C, E\}), (\{B, E\}, \{D\}), (\{C, E\}, \{D\})\}$
- $P_w = \{P_{A, BE}, P_{A, CE}, P_{BE, D}, P_{CE, D}, I_w, O_w\}$
- $F_w = \{\text{arcs}\}$

	A	B	C	D	E
A		→	→	#	→
B	←			→	#
C	←			→	#
D	#	←	←		←
E	←	#	#	→	

Découverte du comportement

- Algorithme Alpha

- Exemple 2



- Vérification des mots

- ABCD ?ok
 - ACBD ? ok
 - AED ? ok