



Automatismes et SED

RdP

-

Réseaux de Petri

RdP = outil de modélisation

" Outil de modélisation orienté vers la représentation des SED dont les variables d'états évoluent brusquement d'une valeur à l'autre sans qu'il soit nécessaire de représenter les phénomènes transitoires. "

■ Exemples :

- Systèmes de commande
- Procédés ou systèmes commandés par modèles à ED
- Systèmes automatiques (commande et procédé)

C'est-à-dire les SED..

RdP = outil de modélisation

- Mode d'évolution asynchrone.

⇒ Modèle par excellence des applications réparties et bien sûr des protocoles de communication (parallélisme et synchronisation).

- Permet :

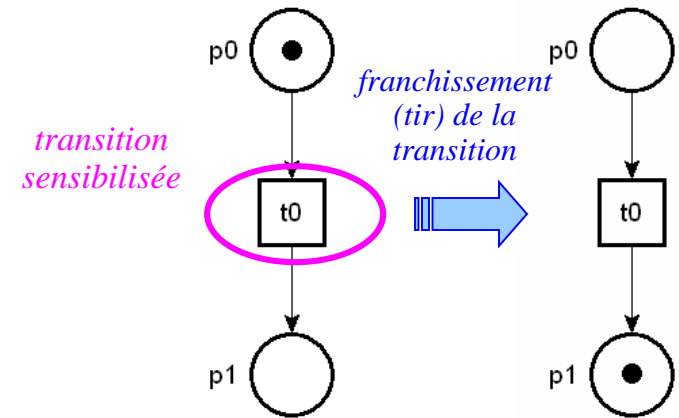
- la simulation
- la validation formelle de propriétés
- l'évaluation de performances (par simulation ou par calcul formel sur des extensions du modèle comportant des données statistiques ou stochastiques).

- Représentation graphique simple.

Réseau de Petri : base

■ RdP autonomes = la base

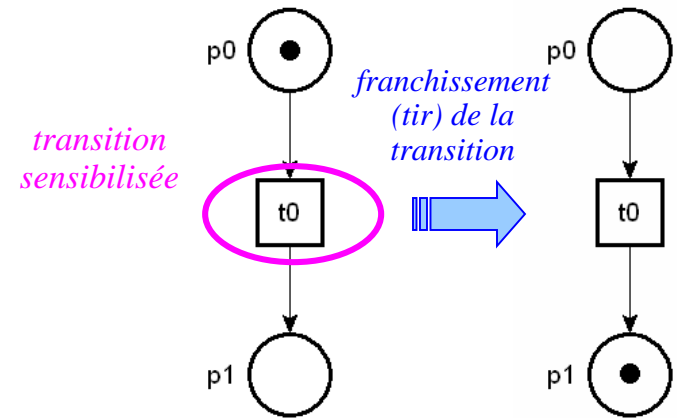
- **Place** : un état du système
- **Transition** : changement d'état
- **Jeton** : état actuel du système



Réseau de Petri : base

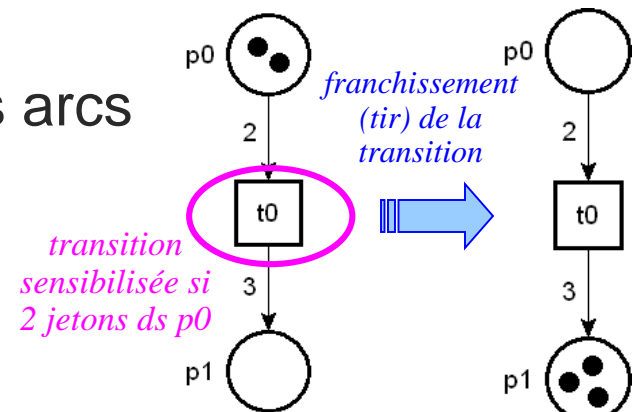
■ RdP autonomes = la base

- **Place** : un état du système
- **Transition** : changement d'état
- **Jeton** : état actuel du système



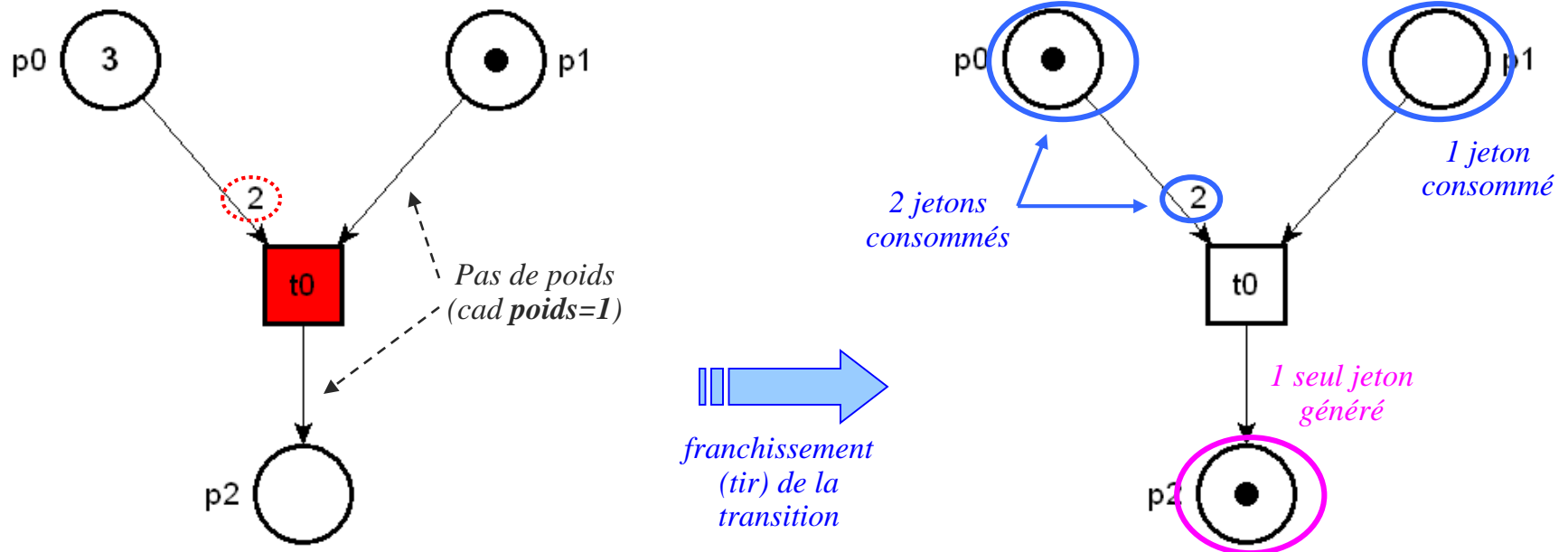
■ RdP généralisés = avec des poids sur les arcs

■ RdP ordinaires = tous les arcs de poids 1

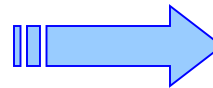
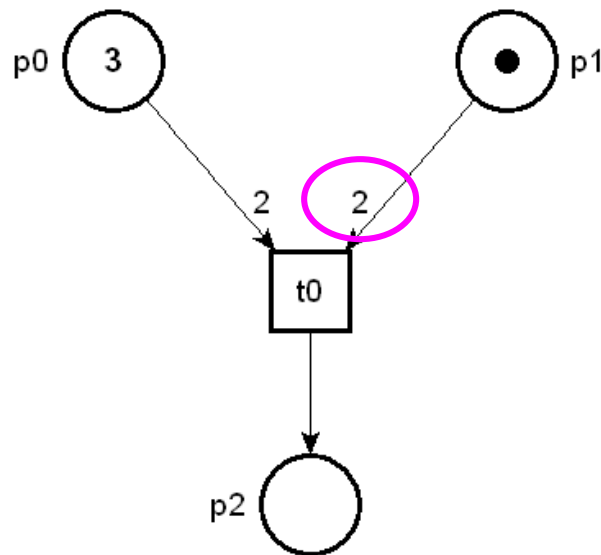


Réseau de Petri généralisés

- transition sensibilisée si 2 jetons dans p_0 et 1 jeton dans p_1



Réseau de Petri généralisés

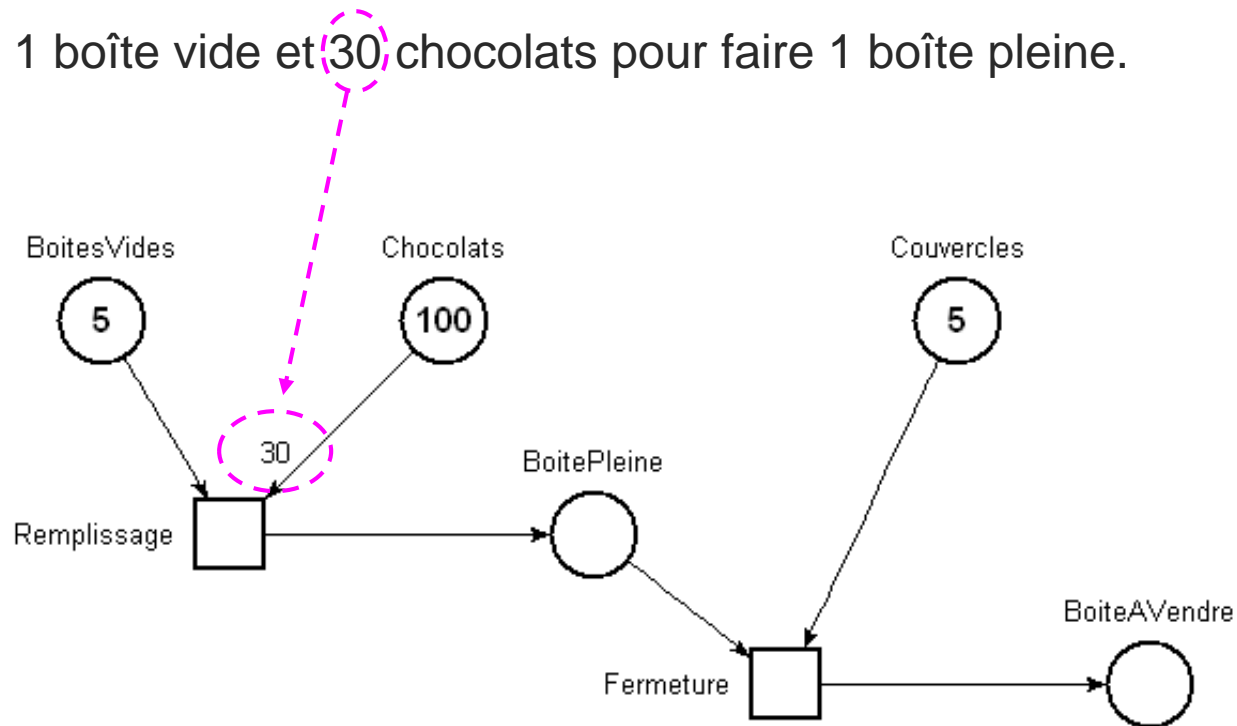



*pas de transition
sensibilisée*

Réseau de Petri généralisés

- Exemple du chocolat :

Il faut 1 boîte vide et 30 chocolats pour faire 1 boîte pleine.





Automatismes et SED

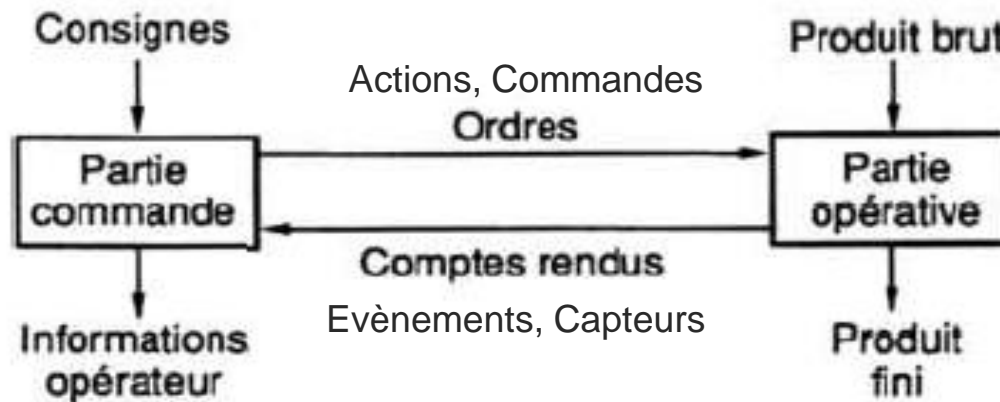
Les RdP interprétés

Extension : RdP interprétés

- Extension des RdP de base.
- Introduction de deux notions :
 - les évènements, pour décrire le comportement d'un système vis-à-vis de son environnement externe;
 - le temps, pour effectuer des évaluations dynamiques sur ce comportement;
- L'occurrence d'un évènement est associée à un instant d'occurrence défini par un repère de temps qui permet d'ordonner les évènements.

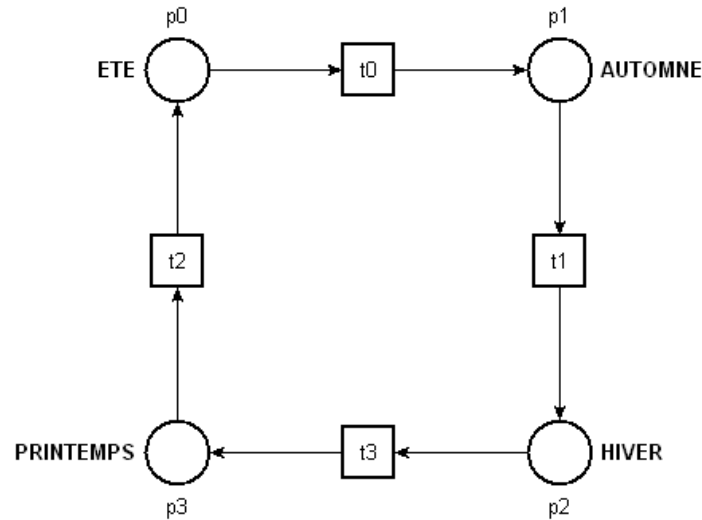
Extension : RdP interprétés

- Un système de commande temps réel est en général un système qui réagit à des évènements externes.
- Prise en considération du système "réel" :
 - **partie opérative** : capteurs associés aux transitions, et prise en compte des variables et opérateurs.
 - **partie commande** : actions exécutées sur les processus; associées aux places.



RdP autonomes / non-autonomes

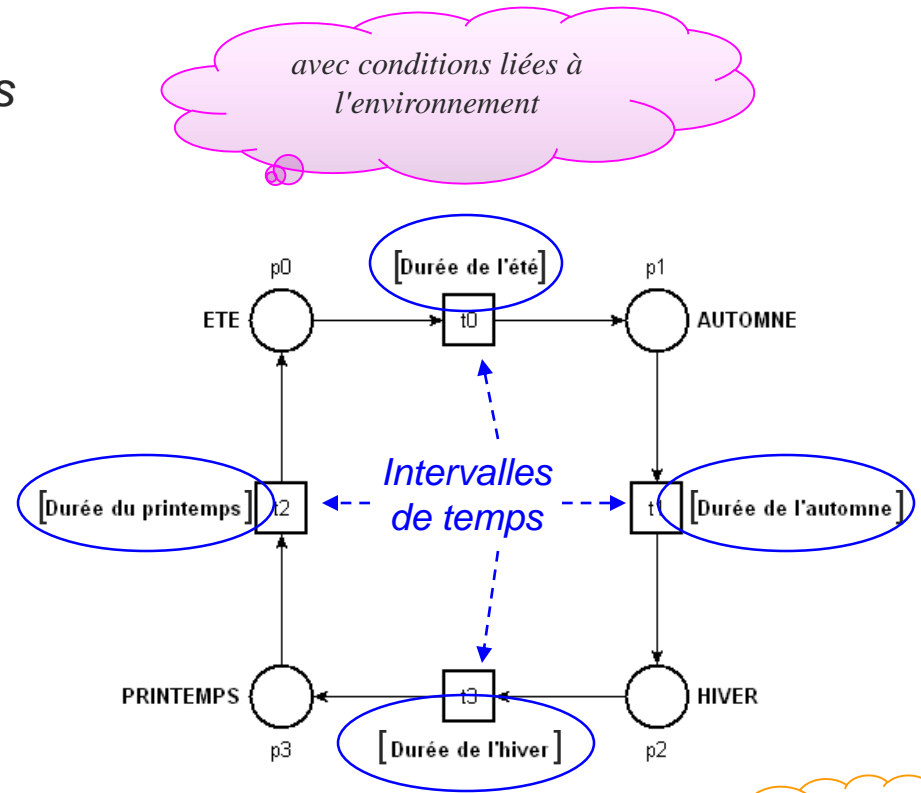
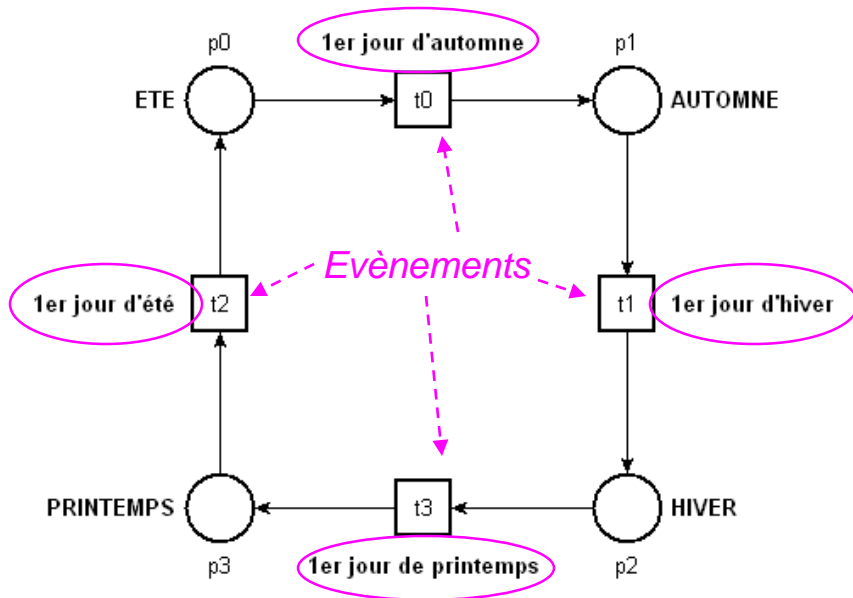
- Exemple : modélisation des saisons



- **RdP autonome (= RdP de base)** : description qualitative, sans préciser les interactions avec l'environnement ou le temps.

RdP interprétés (non autonomes)

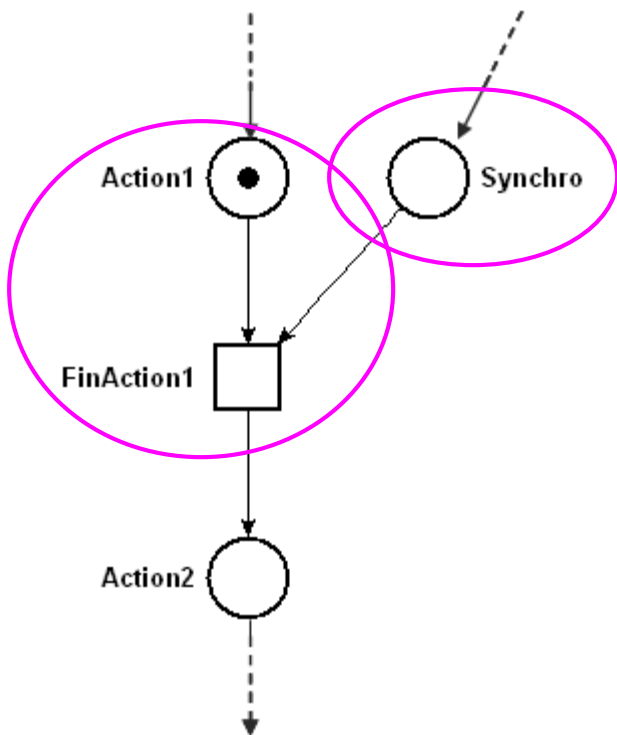
Exemple : modélisation des saisons



cf. suite
du
cours

RdP non-autonome : RdP synchronisé (sur évènements) ou temporisé

RdP interprétés : Attention piège !



■ Situation :

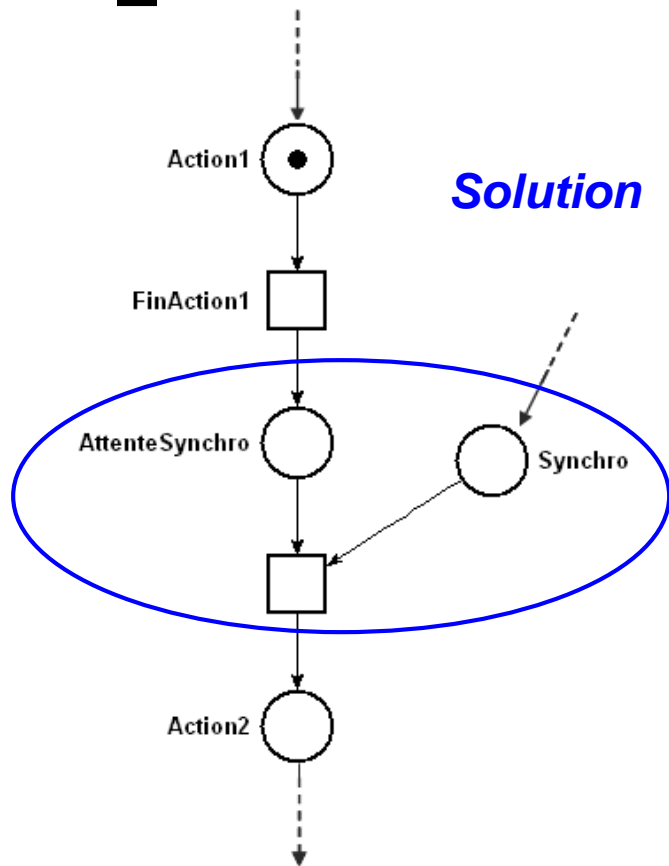
- l'Action1 est en cours : la place Action1 est marquée;
- l'évènement de fin de cette action est FinAction1, lié à la transition;
- Pour commencer l'Action2, il est nécessaire d'attendre une synchronisation.

■ Le problème :

- Si l'évènement FinAction1 se produit avant que la Synchro soit disponible, la transition ne pourra pas être franchie.
- ⇒ **Action1 sera tjs activée, alors qu'elle aurait du s'arrêter.**

➡ ***Ne jamais associer une fin d'action et une attente de synchronisation***

RdP interprétés : Attention piège !



■ Situation :

- l'Action1 est en cours : la place Action1 est marquée;
- l'évènement de fin de cette action est FinAction1, lié à la transition;
- Pour commencer l'Action2, il est nécessaire d'attendre une synchronisation.

■ Le problème :

- Si l'évènement FinAction1 se produit avant que la Synchro soit disponible, la transition ne pourra pas être franchie.
- ⇒ **Action1 sera tjs activée, alors qu'elle aurait du s'arrêter.**

➡ ***Ne jamais associer une fin d'action et une attente de synchronisation***

[Extension : RdP interprétés]

- Les RdPI sont des RdP non-autonomes.
- La modélisation des systèmes à évènements discrets est le plus souvent effectuée en RdPI et/ou temporisés.
- *Rmq : on utilise souvent le terme RdP au lieu de RdPI.*
- Les RdPI offrent les mêmes possibilités d'analyse que les RdP.

Extension : RdP interprétés

- Modélisation avec comme objectif **l'implémentation;**
- Décomposition du modèle en plusieurs "modules" RdP :
 - **découpage fonctionnel** : chaque "module" RdP représente une fonctionnalité du système.
 - **découpage matériel** : chaque "module" RdP représente la commande d'un élément matériel.



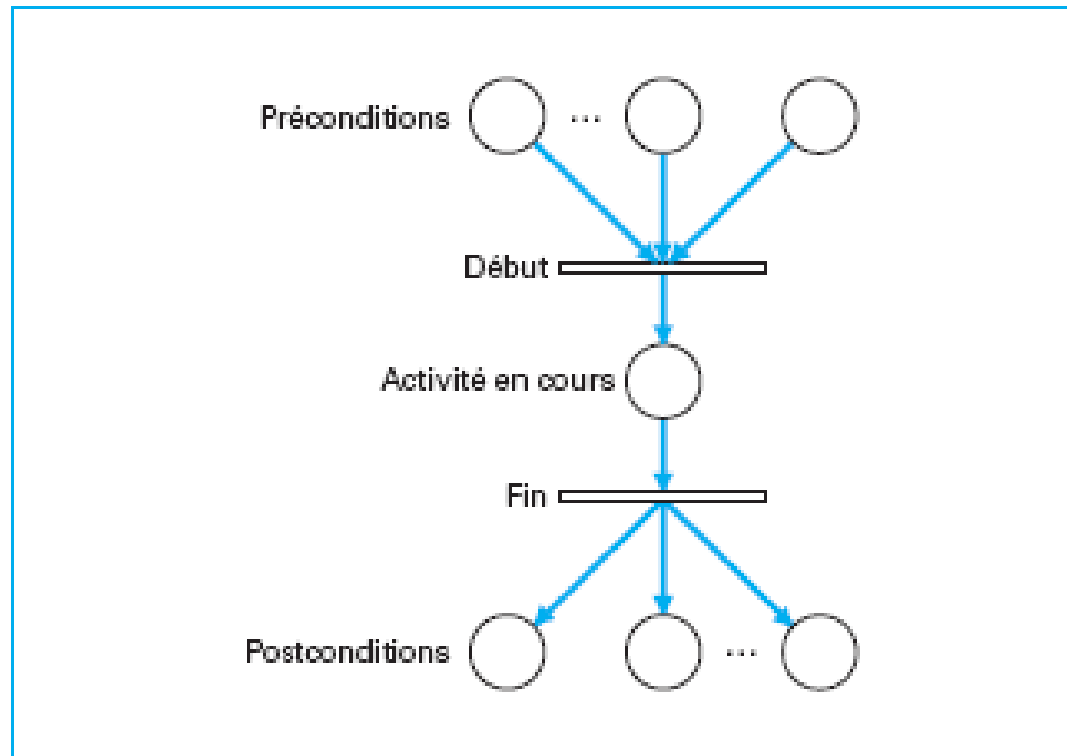
Automatismes et SED

Les RdP

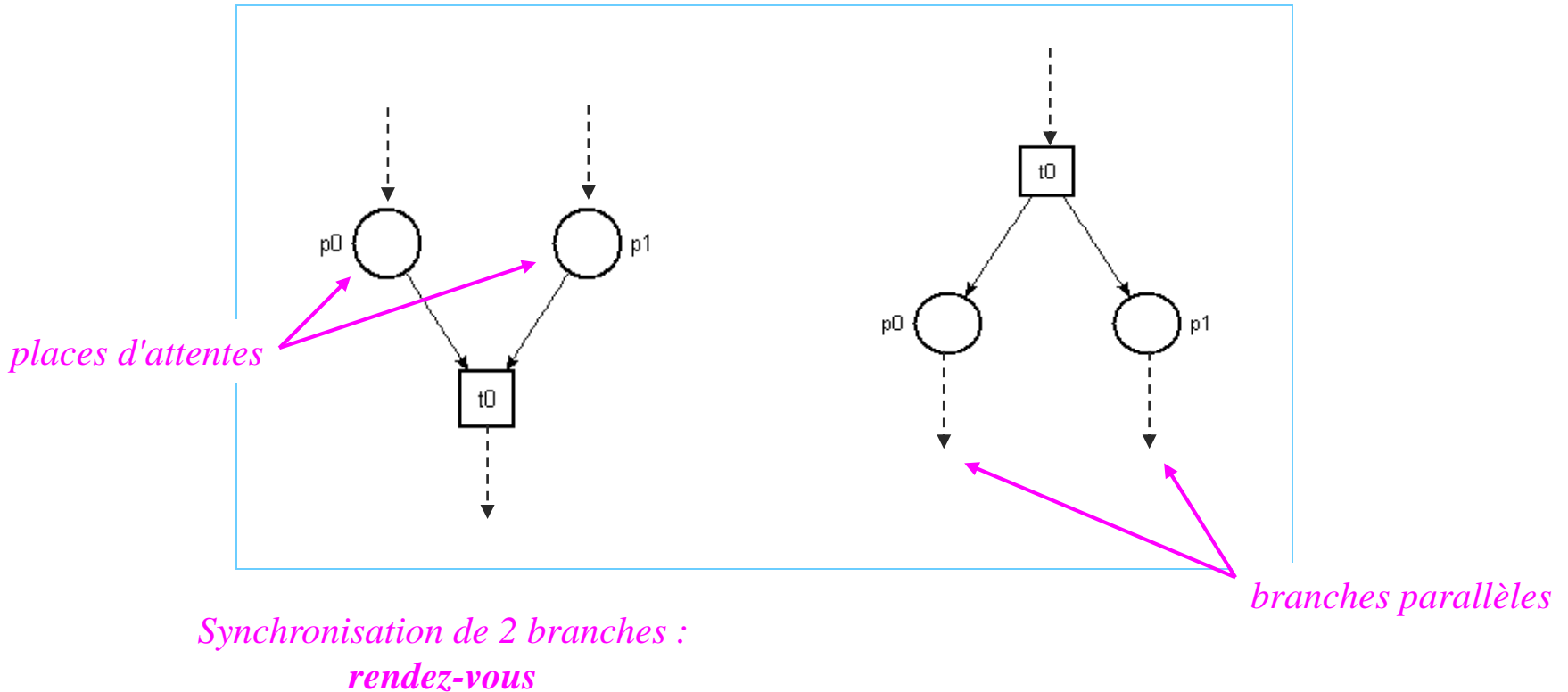
-

Structures de base

Exemples de structures – Activités

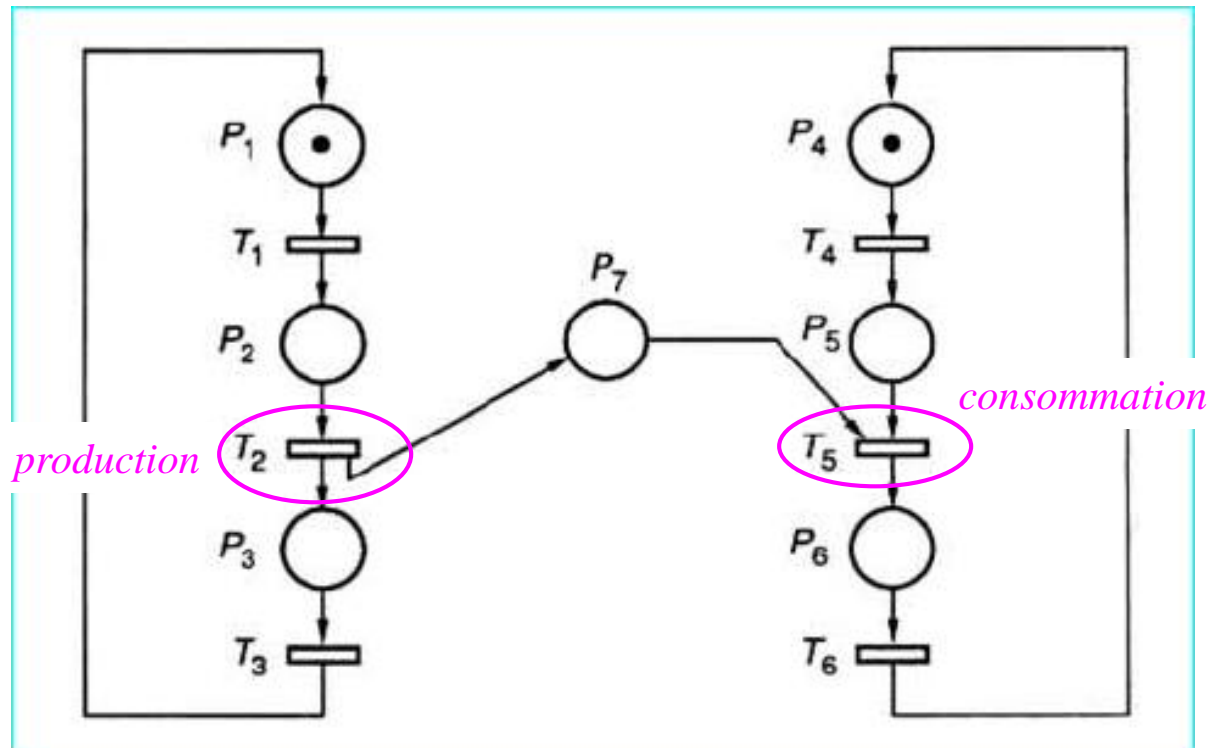


Exemples de structures – Synchronisation et parallélisme



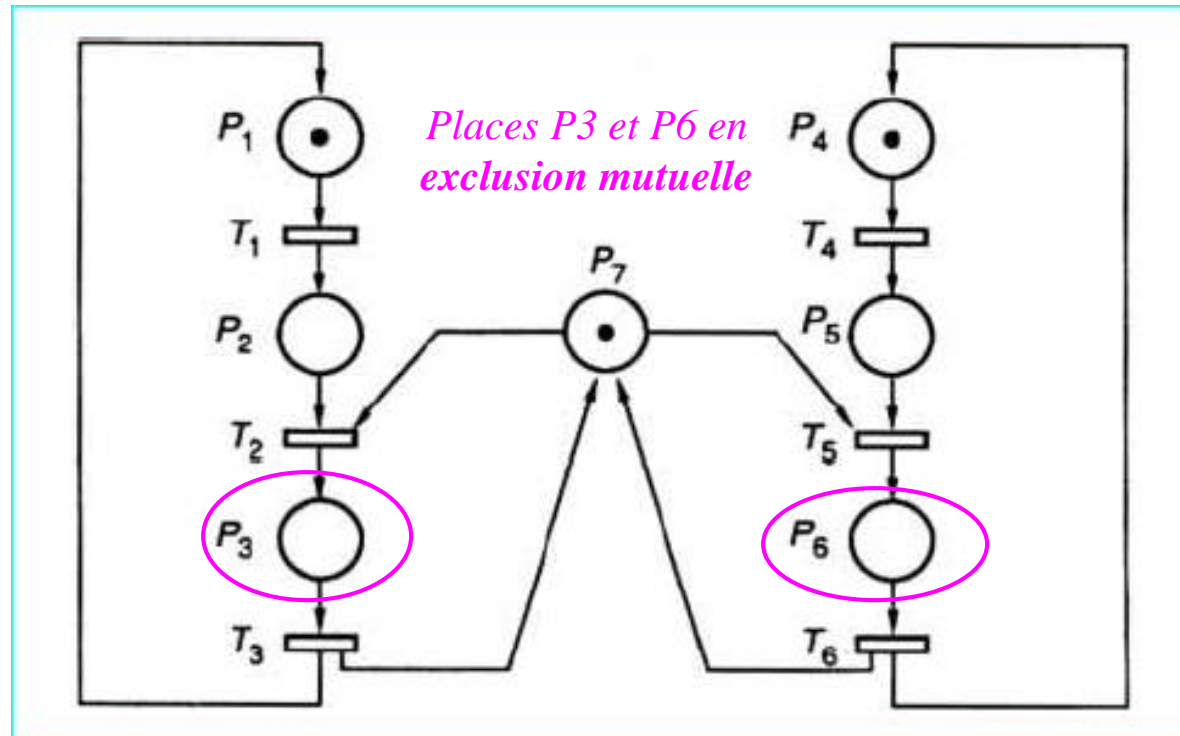
Exemples de structures – Synchronisation - Producteur / Consommateur

- *Rmq : Sans limitation de ressources..*
- *Cette structure peut aussi servir à **synchroniser 2 tâches** (P6 et P2)*



Exemples de structures – Exclusion mutuelle

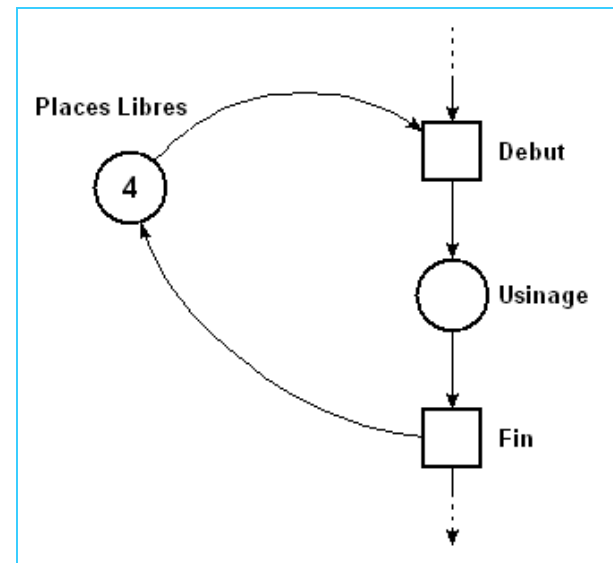
- Ressource partagée,
- Marquage simultanée de P_3 et P_6 impossible.



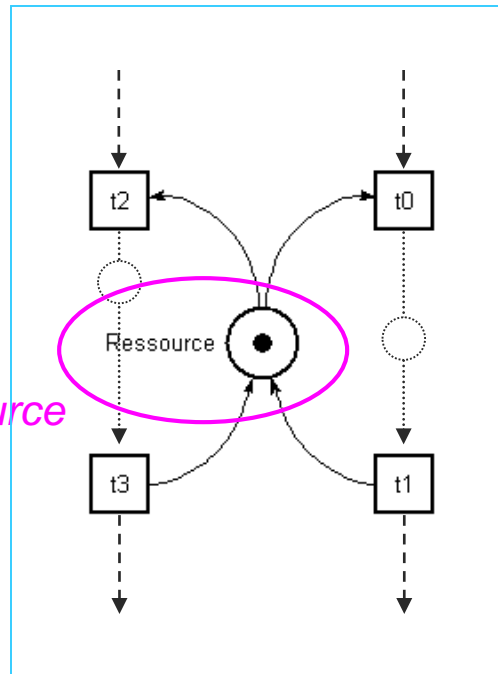
Exemples de structures – Stock à capacité limitée

- Stock : 5 positions libres (P1) ou occupées (P2)

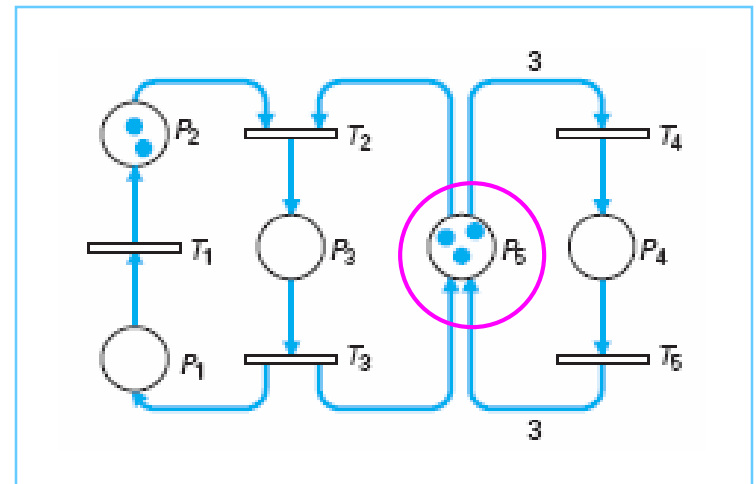
Exemple : une machine permet l'usinage de 4 produits en même temps.



Exemples de structures – Gestion de ressources partagées



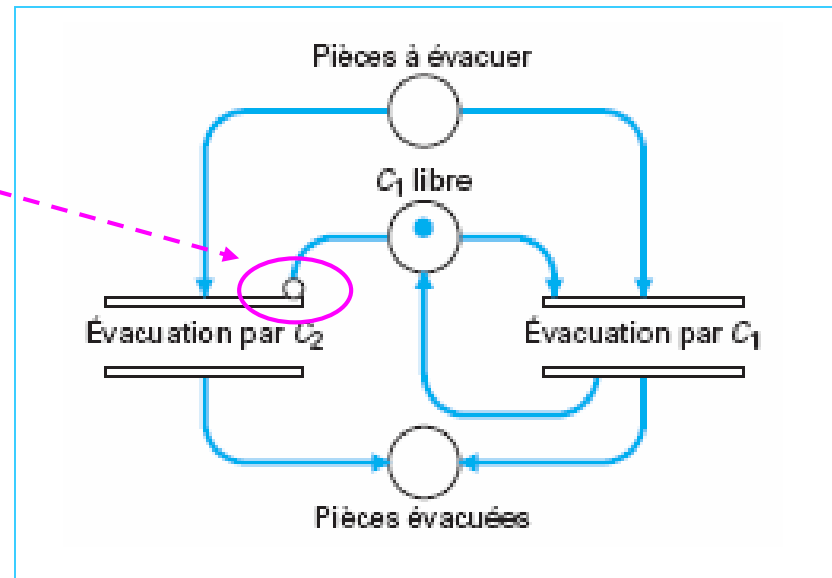
Gestion de la ressource



*Rmq : structure identique à
l'exclusion mutuelle
si ressource=1*

Exemples de structures – Arc inhibiteur

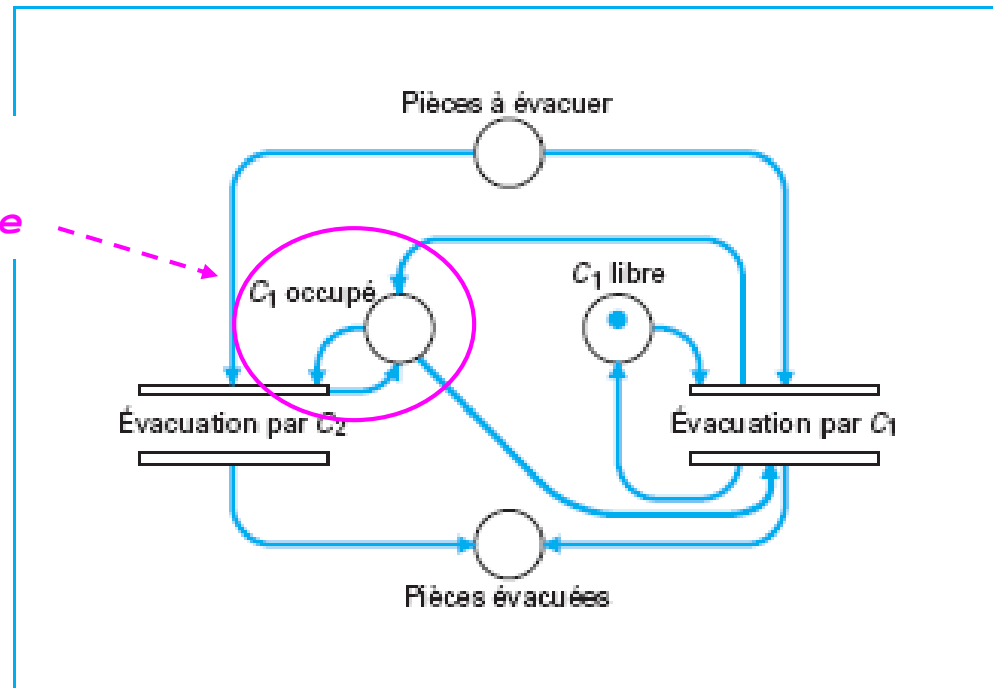
*Évacuation_par_C2 n'est tirable
que si C1_libre est VIDE*



- Exemple d'utilisation : ici, le système donne la **priorité** à l'évacuation par C₁

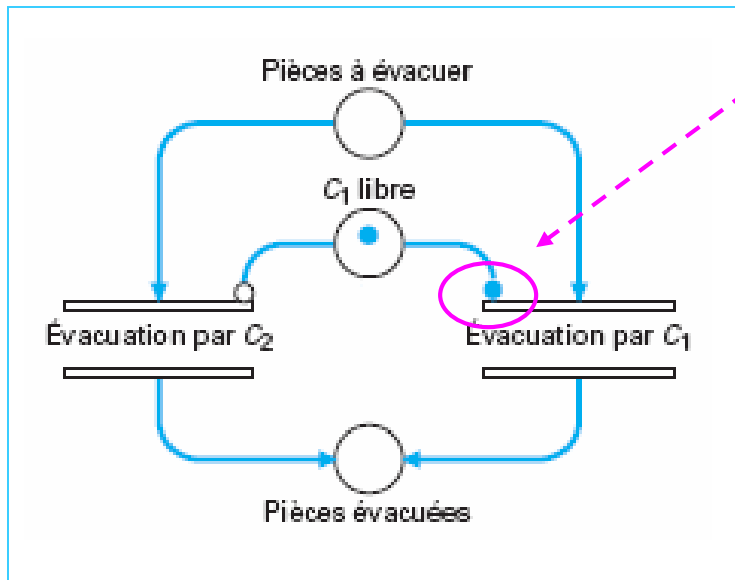
Exemples de structures – Place complémentaire

*C1_occupé à toujours un
marquage opposé à C1_libre*

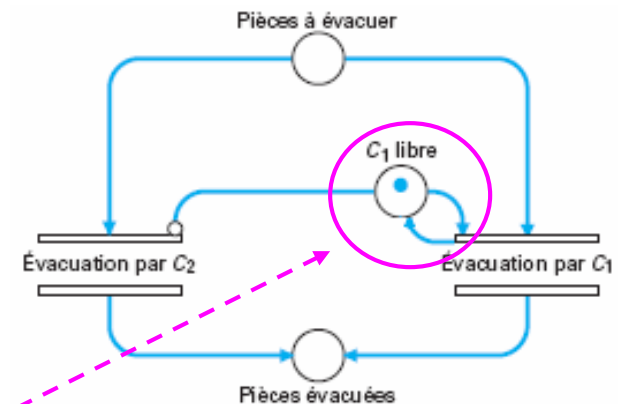


- Cette structure permet la modélisation d'un arc inhibiteur en RdP ordinaires (parfois nécessaire pour permettre l'analyse).

Exemples de structures – Arc test



*Évacuation par C1 n'est tirable
que si C1_libre est pleine, mais
SANS CONSOMMATION DU JETON !*



- *Equivalence en RdP ordinaire : la place est démarquée et immédiatement remarquée par le tir de la transition.*
- *Rmq : ce RdP est inutile car C1_libre ne peut jamais se démarquer..*



Automatismes et SED

Modélisation

-

Exemples

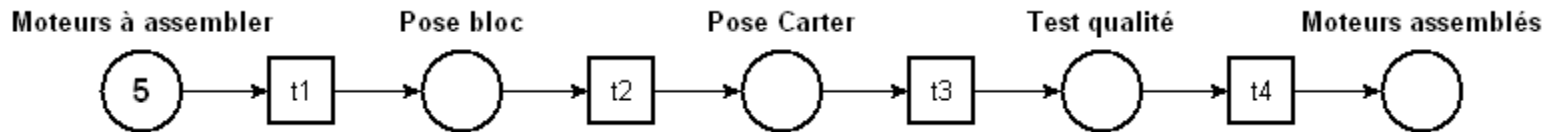
Assemblage d'un moteur

- L'assemblage d'un moteur est constitué de 3 opérations séquentielles, dans 3 ateliers différents :
 - *pose du bloc*
 - *pose du carter*
 - *test qualité*
- Dans chaque atelier un certain nombre d'ouvriers peuvent travailler en parallèle :
 - *Atelier pose du bloc = 3*
 - *Atelier pose du carter = 3*
 - *Atelier test qualité = 2*
- Chaque moteur est posé sur une palette spéciale; il y en a 5 au départ.

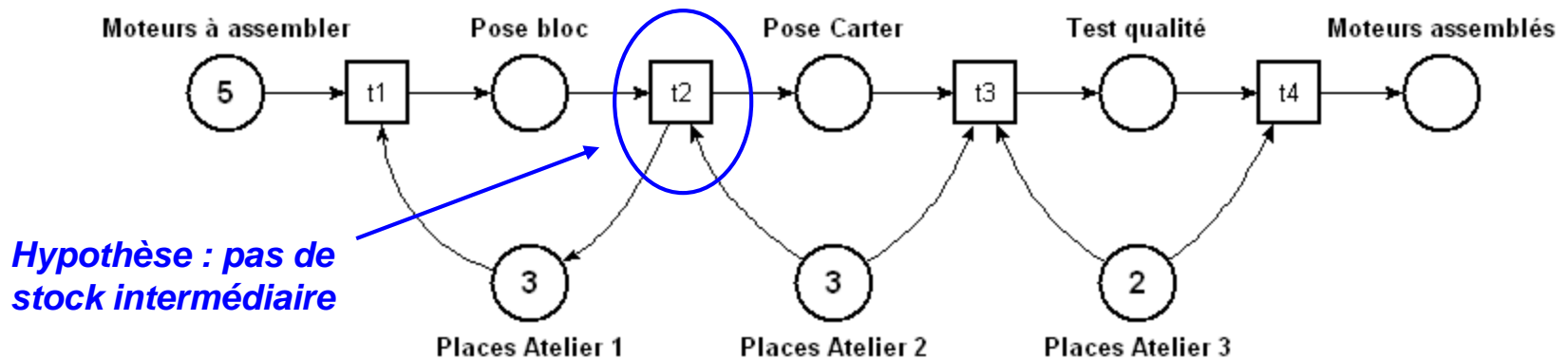
⇒ *Faire la modélisation de ce système*

Assemblage d'un moteur

- Modèle sans prise en compte des ressources :



- Modèle avec prise en compte des ressources :

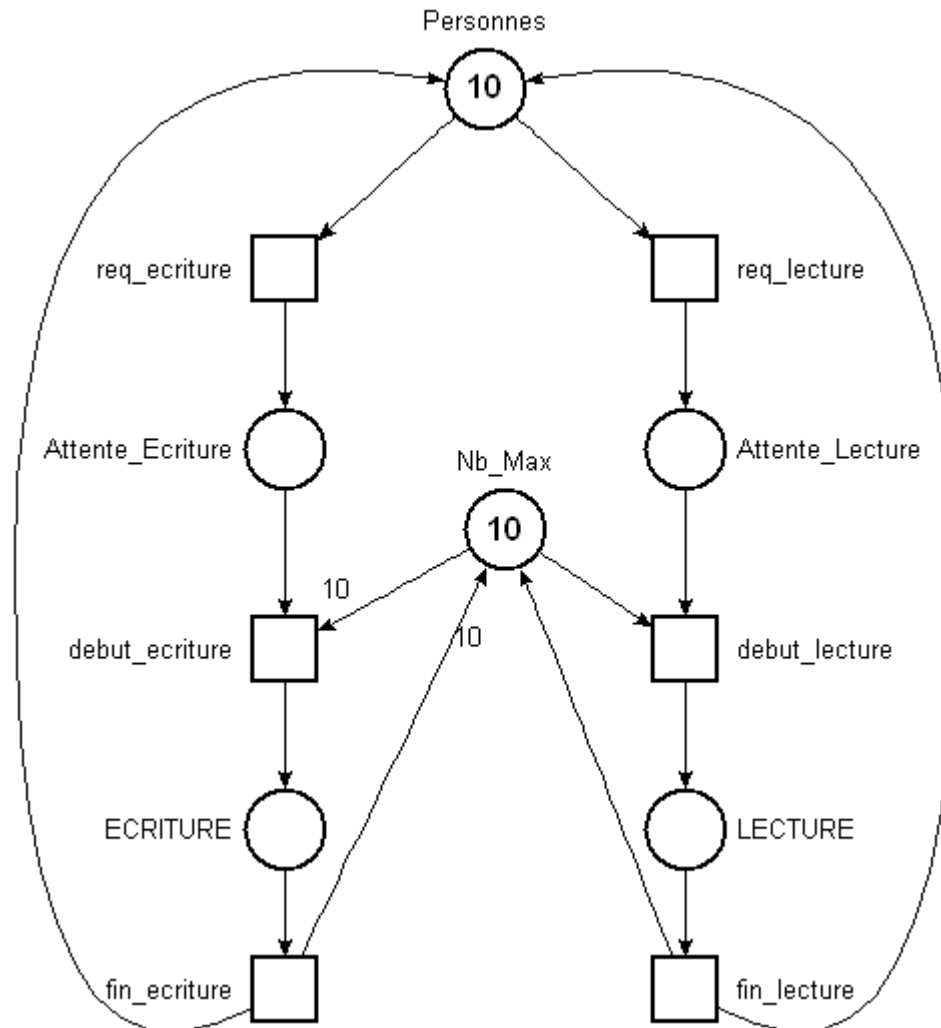


Problème des Lecteurs - écrivains

- On veut modéliser l'accès à une base de données, avec les contraintes suivantes :
 - *K "processus" qui tentent d'accéder à la BD soit pour lire, soit pour écrire.*
 - *Pas d'écritures simultanées sur les données.*
 - *Pas de lecture et écriture simultanées : risque d'incohérence des données lues.*
 - *Par contre, lectures possibles en parallèle.*
- Faire la modélisation de ce système

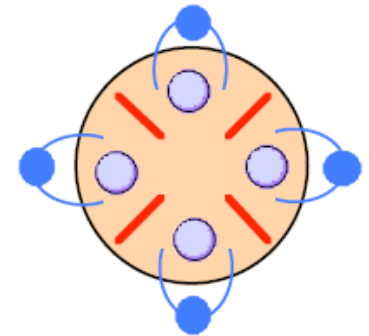
Problème des Lecteurs - écrivains

$K=10$

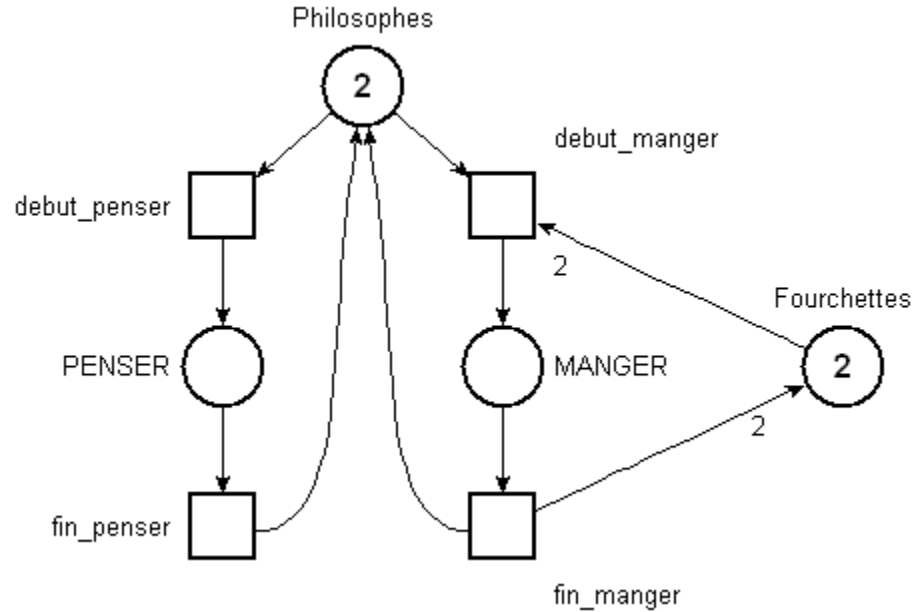
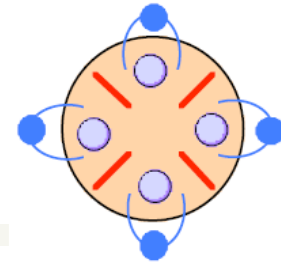


Le dîner des philosophes

- On veut modéliser le classique pb du dîner des philosophes.
- Contraintes :
 - *Table ronde.*
 - *Chaque philosophe est devant une assiette,*
 - *et il y a une fourchette entre chaque assiette.*
 - *Actions d'un philosophe : soit manger, soit penser.*
 - *Pour manger, un philosophe a besoin de ses 2 fourchettes.*
- Faire la modélisation de ce système, dans le cas où il n'y a que deux philosophes :
 - 1) Modéliser le comportement d'un seul philosophe;
 - 2) Modéliser le comportement des deux philosophes.



Le dîner des philosophes



- **ATTENTION : Ceci est une simplification du système global :**
 - On considère les fourchettes comme un stock global.
 - C'est possible ici car tous les philosophes ont accès à toutes les fourchettes.
 - Mais si par exemple 4 philosophes et 4 fourchettes : 2 philosophes voisins ne peuvent pas manger, même s'il reste des fourchettes libres.

[

]

FIN