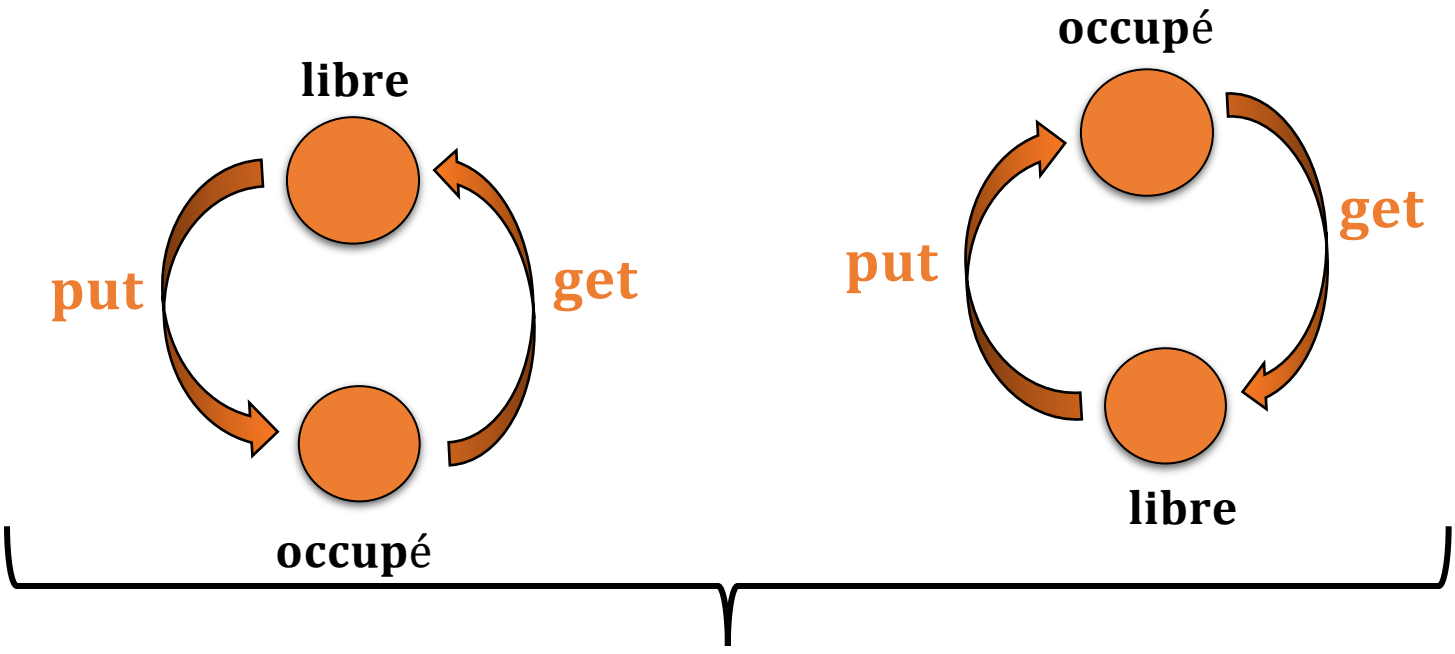
CM3 Réseaux de Petri

Lundi 27·09·2021

Le cours précédent : synchronisation

Si on représente nos systèmes comme des automates, il existe cette notion de synchronisation :

- Si on a des actions asynchrones, chaque composante peut l'exécuter à tous moments·
- Action synchrone : les 2 composantes qui sont doive se synchroniser peuvent pas agir indépendamment, mais qu'ensemble·

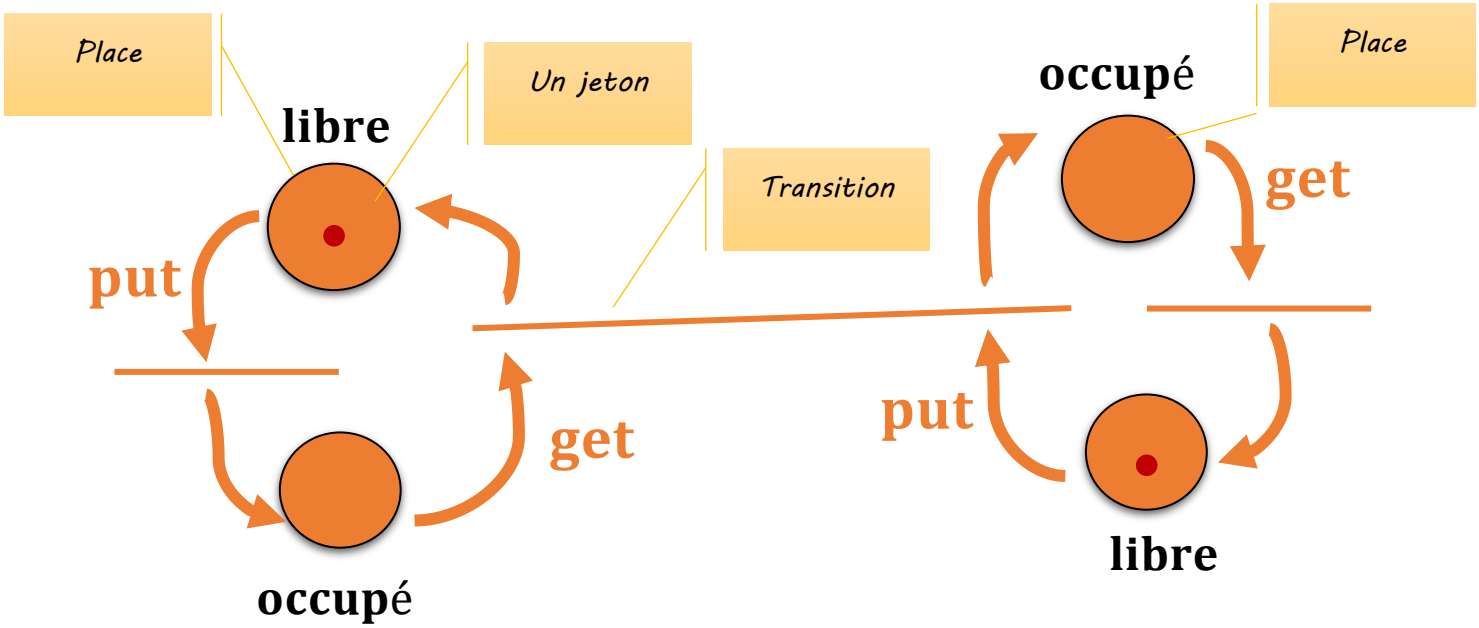


On a vu que si on veux faire de ce system un system syncroniser qui implemente un buffer a 2 places, faut syncroniser des action...

Le cours d'ajd : Réseaux de Petri

Dans le cours d'ajd on va introduire un autre type de modèle qui permet de modéliser plus simplement la synchronisation : **les réseaux de Petri**.

On va donner la syntaxe et la sémantique de cela en termes de systèmes de transitions.



Notion de transition

On imagine qu'on dépose un jeton dans « libre » (on parle de « marquage du réseaux »). Une certaine transition est exécutable que si tous les places qui ont un arc vers cette transition contienne un jeton.

Pour l'exécution d'une transition on va donc consommer un jeton de tous les places qui sont en amont de la transition, et on va mettre un jeton a la sortie.

On a donc les notions suivantes :

- **Place**
- **Transition**
- **Sémantique qui dit comment s'exécute une transition** (passage) .

Dans notre réseau, il y a dans chaque Place au max 1 jeton,
car dans les automates il y a un jeton dans une machine séquentielle, et donc si on compose en parallèle, les jetons qui circules dans chaque machine pour faire certaines actions doive s'entendre, on ne peut pas circuler tous seul tous le temp· C'est ainsi qu'on va concevoir des réseaux de patri à partir de machine d'états·

Mais, de manière générale, dans les réseaux de pétri j'ai pas la contrainte qui fait que dans une Place il y a au max 1 jeton (le nombre de jetons et donc un entier (int) et pas un Boolean comme dans l'exemple précédente·

La syntaxe (grammaire)

Un réseau de Petri est d'abord donné par un graphe (un formalisme qu'on peut visualiser, jusqu'à un certain point, évidemment)·

RdP

Réseau de Petri

$R = (P, T, \Delta)$

Ensemble de transition

Ensemble de places

Une relation « delta »

Dune certain manière, c'est un graphe biparti

2 types d'arcs : dans un sens ou dans un autre

$$\begin{cases} P \text{ est un ensemble de places} \\ T \text{ est un en ensemble de transition} \\ \Delta \subseteq P \times T \ \cup T \times P \end{cases}$$

La sémantique

Photo de la situation actuelle : combien de jetons il y a dans chaque une des places ?

Marquage

Marquage de R

$$Marquage\ de\ R\ :\quad M : P \longrightarrow \mathbb{N}$$

C'est une fonction qui associe à chaque Place un entier naturel (le nb de jetons).

Ça c'est un marquage.

Définition qui me fait passer d'un marquage vers un autre

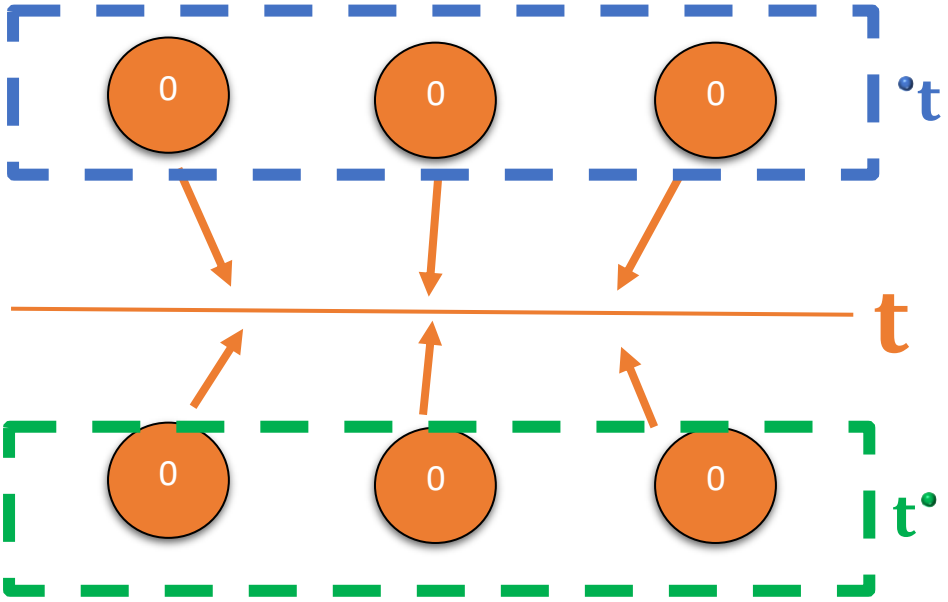
Transition exécutable : tous les places qui sont en amont dune certain transition au moins un jeton.

$$Pour\ t \in T,\qquad {}^{\bullet}t = \{p \in P | (p,t) \in \Delta\}$$

Je vais appeler ${}^{\bullet}t$ l'ensemble des places appartenant a p, tel que $(p,t) \in \Delta$

$$t^{\bullet} = \{p \in P | (t,p) \in \Delta\}$$

${}^{\bullet}t$ = l'ensemble des places qui ont des arcs vers t



Graphe des marquages

Soit M et M' deux marquages

Soit $t \in T$

$M \triangleright M'$

Formalisation de ce qu'on a vu pas formellement : la condition est que t doit être exécutable à partir de M

- Décrémenter de 1 $^\bullet t$
- Incrémenter de 1 $t^\bullet$

Relation de translation

ssi

$(1) \forall p \in {}^\bullet t, \quad M(p) \geq 1$

(2) Soit $M'' + q$, $\forall p \in P$

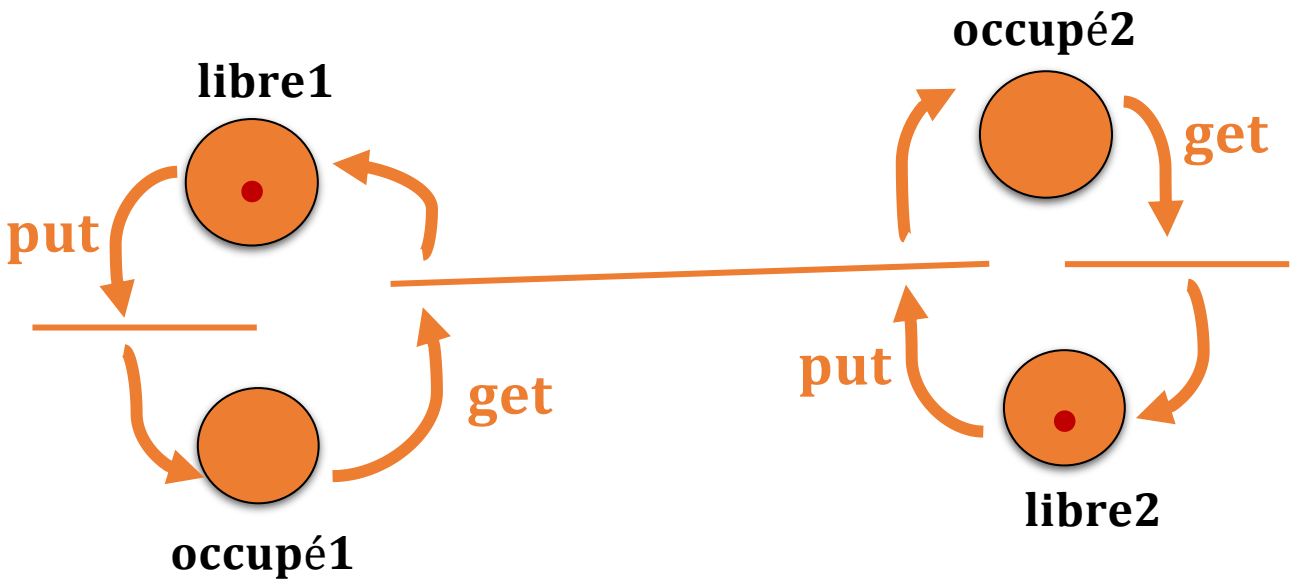
On va définir un marquage intermédiaire qui est obtenu lorsque on décrémente d'abord

$$M''(p) = \begin{cases} M(p) \text{ si } p \notin {}^\bullet t \\ M(p) - 1 \text{ si } p \in {}^\bullet t \end{cases}$$

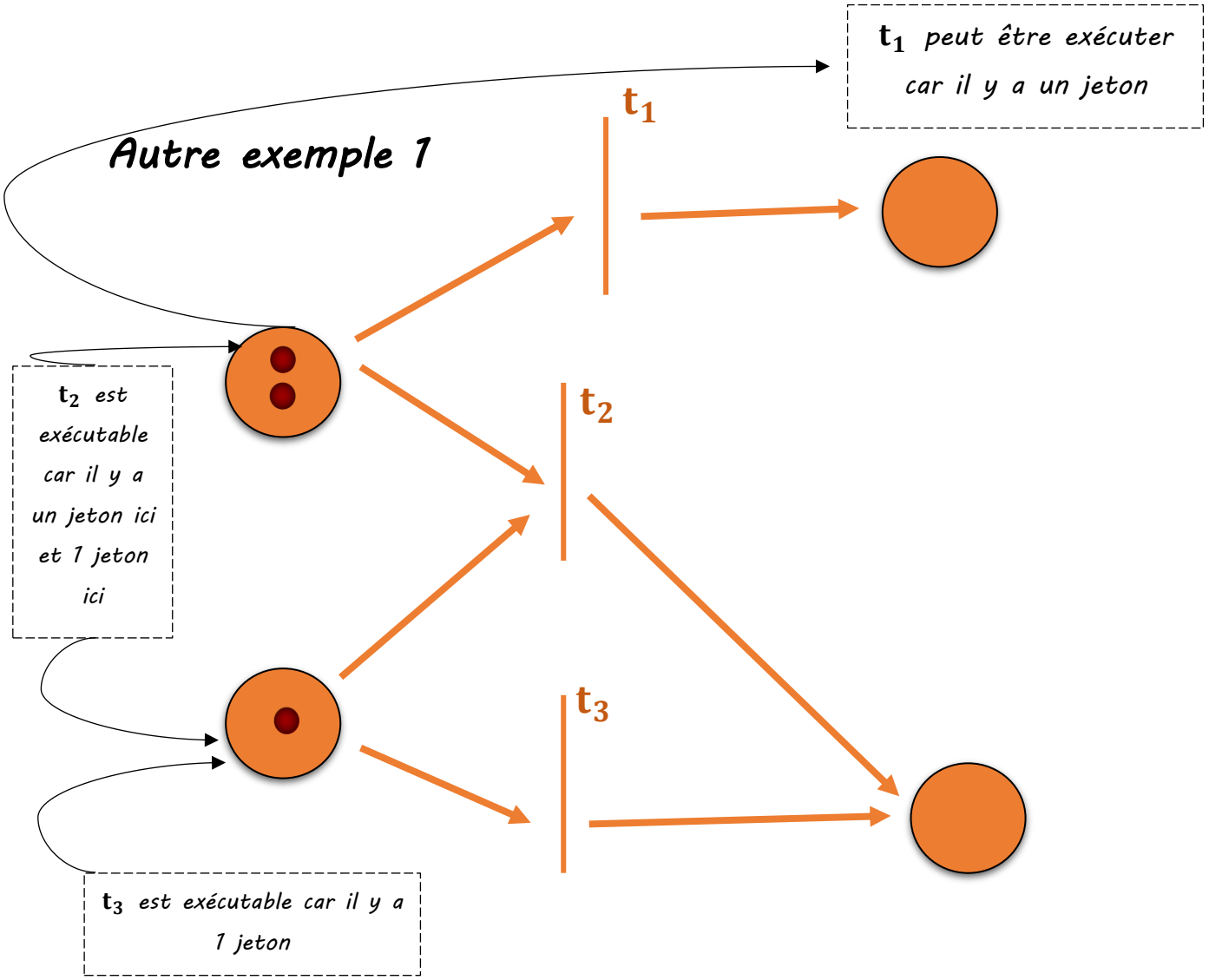
Alors

$$\forall p \in P \; M'(p) = \begin{cases} M''(p) \text{ si } p \notin t^\bullet \\ M''(p) + 1 \text{ si } p \in t^\bullet \end{cases}$$

On a défini les marquages et comment on va bouger d'un marquage a l'autre.

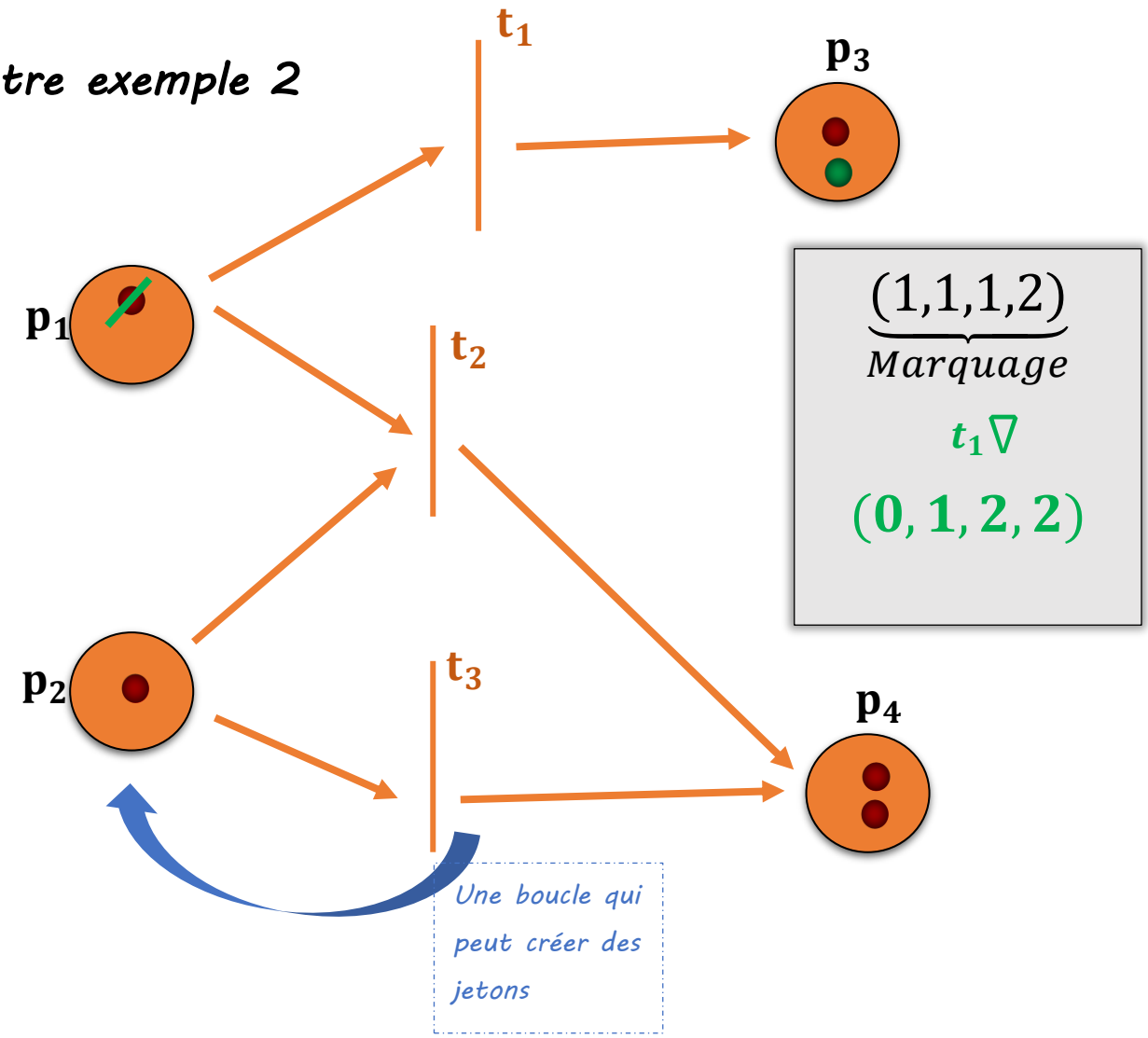


Lorsque on va exécuter, on va retrouver le même système vu la semaine dernier.



- Si j'exécute t_1 : t_2 , t_3 toujours exécutable. t_1 aussi exécutable a nouveau, mais alors il y a que t_3 qui est exécutable. Sinon, on peut exécuter t_2 et alors on prend 1 jeton de chaque place et la place en bas à droite reçois donc 1 jeton.
- Si j'exécute t_1 et on tire la transition t_3 alors t_2 deviens inexécutable.

Autre exemple 2



$\underbrace{(1,1,1,2)}_{Marquage}$

$t_1 \nabla$

$(0,1,2,2)$

.

Si t_3 s'exécute, il y a un jeton de plus a p_4 et il reste tjr un jeton a p_2

Donc, dans ce cas ce n'est PAS borné

Marquage accessible

Pour un marquage initial M_0,

M *est accessible* (à *partir de* M_0)

Ssi $M_0 \triangleright^* M$

Not (notation) : $M \triangleright M'$

pour $\exists\, t \in T$, $M \overset{t}{\triangleright} M'$

Def : $\triangleright^i M'$ *pour* $i \geq 0$

Relation « i fois »

De la composition de relations

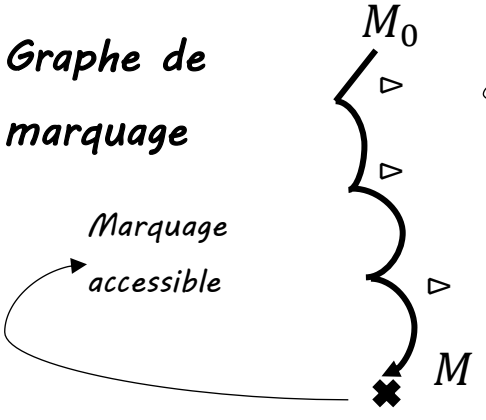
$$\begin{cases} \triangleright^0 = \pm\alpha \\ \triangleright^{i+1} = \triangleright^i \circ \triangleright \end{cases}$$

$M \triangleright^i M'$ *ssi il existe un chemin de lng i entre M et M' dans le graphe de marquage.*

$$\triangleright^* = \cup \triangleright^i \quad i \geq 0$$

$M \triangleright^* M'$ *ssi il existe un chemin entre entre M et M' dans le g·d·m (graphe de marquage)*

Graphe de marquage



Un réseau R est borné (à partir d'un marquage initial ssi …)

$$R \text{ est borné (à partir de } M_0)$$

$$\text{Ssi } \exists k \in \mathbb{N} \mid \underbrace{\forall M \quad M_0 \rhd^* M}_{M \text{ est accessible}} \quad , \quad \underbrace{\forall p \in P \quad M(p) \leq k}_{\substack{\text{Jamais plus de k jetons} \\ \text{dans la place}}}$$

Marquage

Notion qui me dit combien il y a de jetons dans chaque place du réseau.

k-borné

Pour k donné, k-borné
quel que soit M accessible (∀ M accessible)

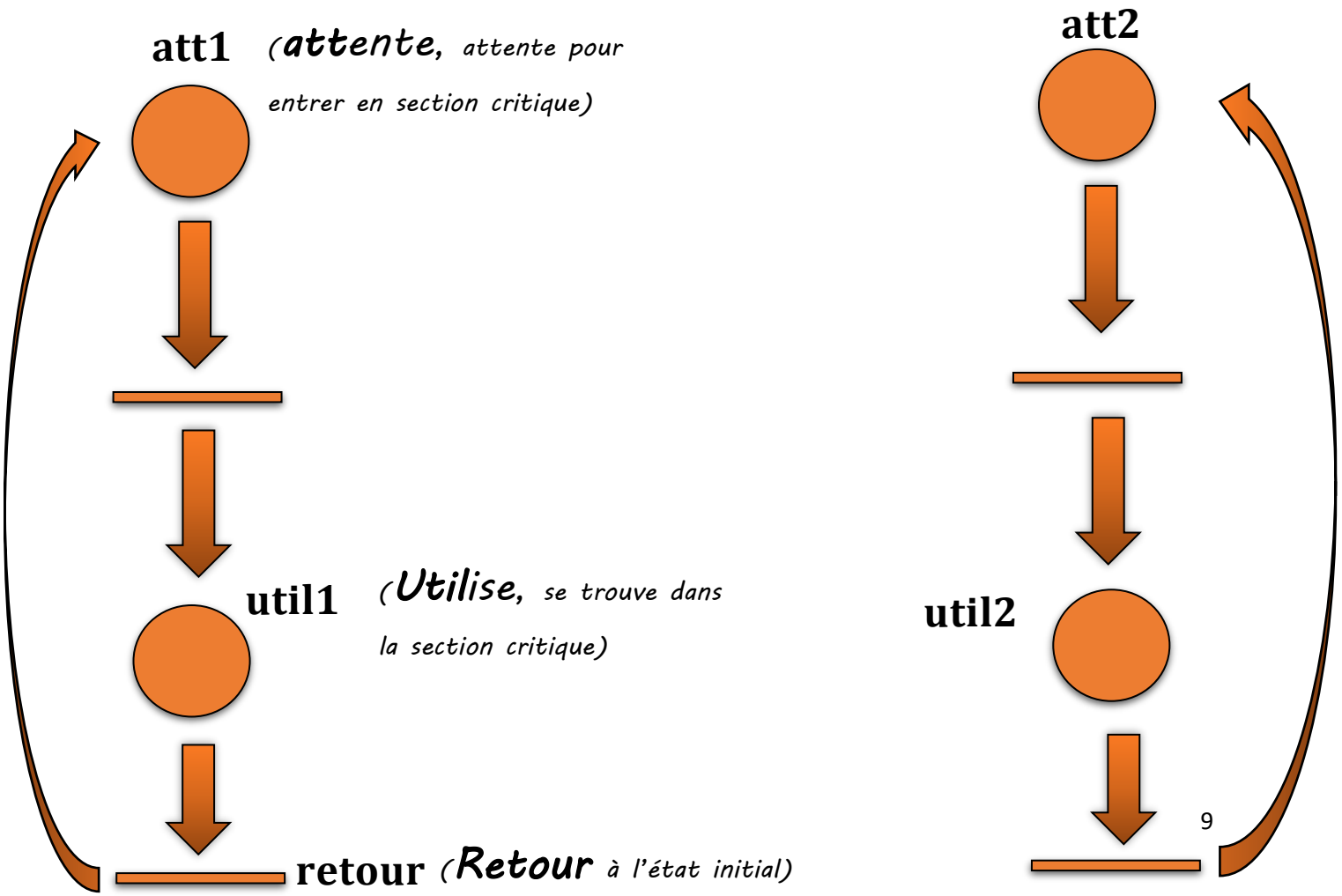
$$M(p) \leq k \quad \forall p \in P$$

Dans notre 1er exemple : 1-borné (chaque place peut être vu comme un Boolean)

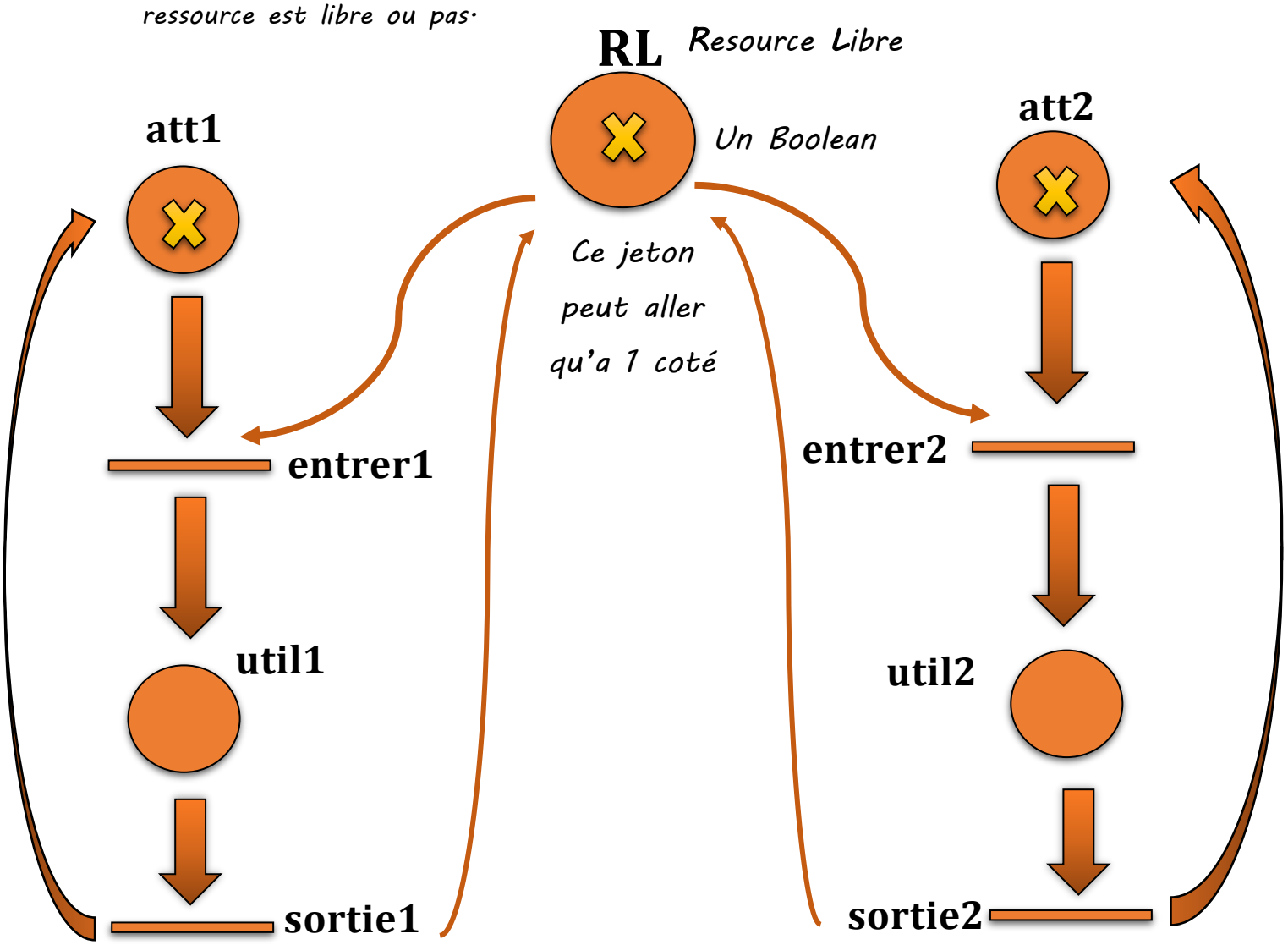
Lecteurs - Rédacteurs

On a vu la semaine dernière en TD le pbm de l'exclusion mutuel· Mtn, on va voir le pbm des lecteurs-rédacteurs :

On a 1 variable, 1 lecteur et 1 rédacteur· Tous les 2 on besoin d'accéder à la ressource· Comment on va modéliser ça ? On veut régler l'accès a la ressource, pour laquelle les 2 veules accéder en exclusion mutuelle· Faut pas qu'ils puisse accéder au même moment·

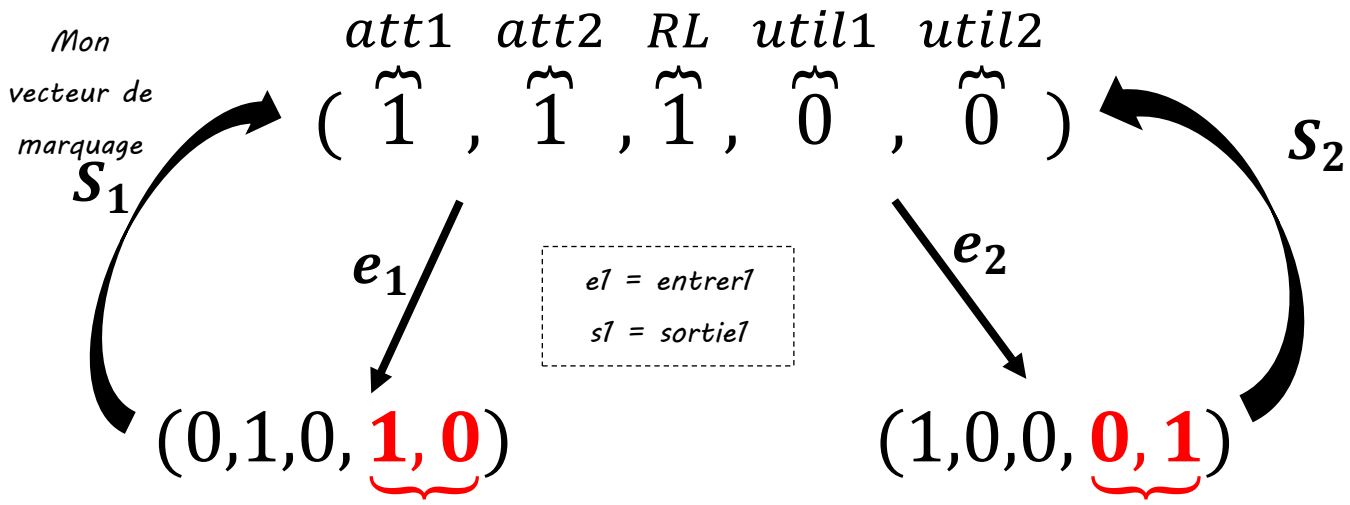


Il ne faut pas que le lecteur et le rédacteur puisse accéder a la section critique au même moment, mais il est préférable que un processus ne devra pas demander l'autorisation de l'autre processus. Donc, il nous faut un mécanisme qui puisse donner l'autorisation : une information qui nous dit si la ressource est libre ou pas.



Faut garantir que ça soit automatique : que soit att1 va gagner, soit att2.

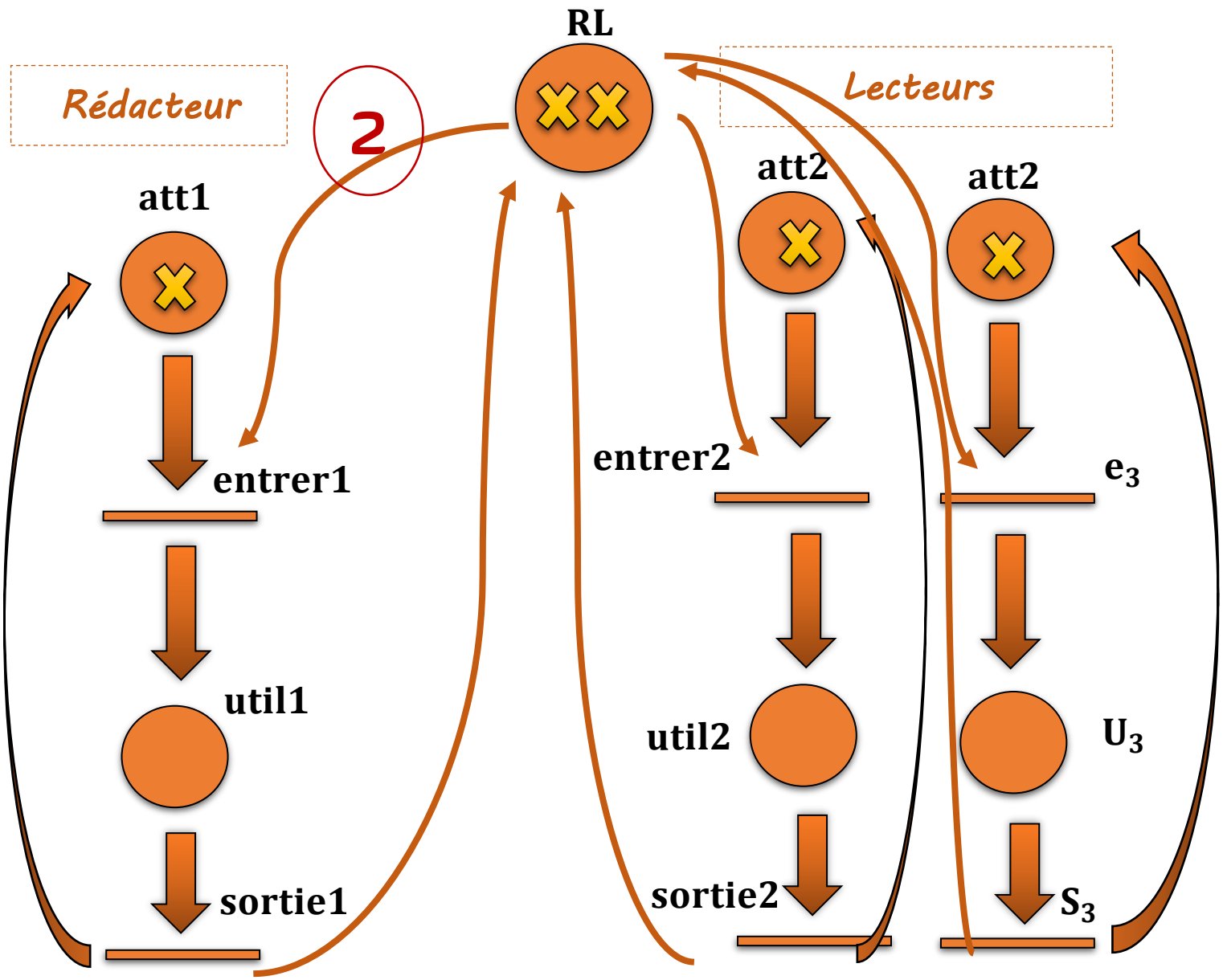
On peut construire un graphe de marquage pour voir que l'exclusion mutuelle est respectée.



C'est ça le graphe de marquage.
On constate que (u1,u2) sont jamais tous les deux a 1 !

Exercice

Mtn on a 2 lecteurs et 1 rédacteur. Le rédacteur, lorsqu'il écrit, il veut être seul. Cependant, les lecteurs peuvent lire ensemble (pendant ce temp l'écrivain ne peut pas écrire). Exclusion mutuelle, évidemment.



Le rédacteur a besoin de prendre 2 jetons de RL pour entrer en section critique.