Etat global d'un système réparti :

Algorithme de Chandy et Lamport

Introduction :

Obtenir une vision instantanée d'un système réparti, consistant en la collection des états des différents sites le composant (typiquement une image mémoire de chacun des sites) est difficile à obtenir sans figer chacun des systèmes.

L'absence de mémoire commune et le caractère aléatoire des délais d'acheminement des messages échangés entre les sites rendent impossible le calcul d'un état global du système dans un système réparti. Typiquement, l'image qu'un site possédera de l'état des autres sites ne pourra lui être communiquée qu'au travers de messages et ne pourra de ce fait correspondre qu'à un état du passé de ces sites: l'ordre chronologique des différentes images ainsi collectées n'est pas connu a priori. L'absence d'un état global accessible directement constitue incontestablement une caractéristique de la répartition et est source de difficultés dans le développement d'applications relatives à

- l'**interblocage** ou **verrou mortel (***deadlock*): situation dans laquelle un ensemble de processus est en situation de blocage du fait de l'existence d'un cycle dans le graphe d'allocations et de demandes des ressources à ces processus;
- le **ramasse-miettes (***garbage collecting*), opération consistant en le récupération des ressources allouées à un objet inutilisé;
- la **mise au point (***debugging*): opération incluant par exemple la consultation et/ou la modification des valeurs de variables dans différents composants *à un instant donné*.
- **Quelques définitions**
- - Un site se caractérise par son **histoire locale**, à savoir\ la suite (ordonnée) des événements (locaux, envoi ou réception de messages) qui s'y produisent. Ainsi, au site $\mathbf{S_i}$, on associe la suite $\mathbf{h_i = <e_i^1, e_i^2, ..., e_i^k, ...>}$.

  L'**histoire globale** du système de **N** sites $\mathbf{S_1}$, ..., $\mathbf{S_N}$, est $\mathbf{H = <h_1, ...,h_N>}$
- - On désigne symboliquement par $\mathbf{s_i^k}$ l'état du site $\mathbf{S_i}$ avant que l'événement $\mathbf{e_i^k}$ ne s'y produise ($\mathbf{s_i^0}$ est l'état initial du site $\mathbf{S_i}$).
- - Une **coupure C** de l'histoire globale d'un système est un N-uple dont chaque composante est un préfixe de l'histoire locale du site correspondant:

$C = <c_1, ..., c_j, ..., c_N>$

où $c_j = <e^1_j, e^2_j, ..., e_j^{m_j}>$

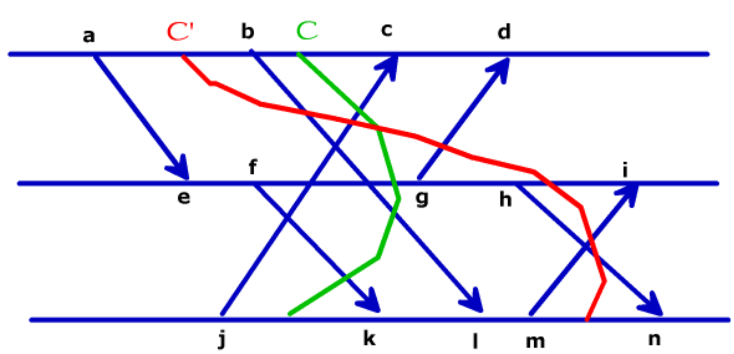
- \- Une **coupure C** est dite **cohérente (ou consistante)** si elle est fermée vis-à-vis du passé de ses éléments (tout élément dans le passé d'un événement de la coupure appartient lui-même à la coupure). En d'autres termes:

> **Soit C une coupure et e un événement dans $C = <c_1, ..., c_j, ..., c_N>$**
> **c'est-à-dire tel que il exite un numéro j de site tel que e appartient à c_j**
> **Pour tout f dans Passé(e), f appartient à C,**
> **c'est-à-dire qu'il exite un numéro k de site tel que f appartient à c_k**

- \- La donnée d'une coupure fournit état local pour chaque site **j**, à savoir **$s_j^{m_j}$**
- \- Un **état global d'un système réparti** est constitué d'une part d'un état local de chacun des sites et d'autre part d'un état de chacun des canaux de communication. La donnée d'une coupure permet d'obtenir un instantané de l'état local de chacun des N sites **$<s_1^{m_1}, ..., s_1^{m_N}>$**. Un état global cohérent (ou consistant) est un état correspondant à une coupure cohérente.

Exemple

Dans l'exemple correspondant à la figure suivante, la coupure **C** est cohérente alors que **C'** ne l'est pas.



En effet on a

- pour **C** :
  - $\mathbf{c_1 = <a, b>}$, $\mathbf{c_2 = <e, f>}$, $\mathbf{c_3 = <j>}$
  - **Passé(b) = {a, b}**
  - **Passé(f) = {a, e, f}**
  - **Passé(j) = {j}**

  Ainsi qu'on peut le constater, chacun des éléments du passé d'un évément de la coupure (il suffit de considérer sur chaque site l'événement le plus récent qui est dans la coupure) appartient à l'un des $\mathbf{c_j}$;

- pour **C'** :
  - $\mathbf{c'_1 = <a>}$, $\mathbf{c'_2 = <e, f, g, h>}$, $\mathbf{c'_3 = <j, k, l, m>}$
  - **Passé(a) = {a}**
  - **Passé(h) = {a, e, f, g, h}**
  - **Passé(m) = {a, b, e, f, j, k, l, m}**

  Pour cette coupure, on constate que **m** appartient à, la coupure et **b** appartient à, **Passé(m)** mais que **b** n'appartient pas à la coupure.

Estampillage vectorielle et caractérisation des coupures cohérentes

On considère la datation vectorielle des événements et on associe à une coupure **C** définie par les événements $\mathbf{<e_1, ..., e_j, ..., e_N>}$ ($\mathbf{e_j}$ est l'événement le plus récent du site $\mathbf{S_j}$ appartenant à la coupure). Associons à la coupure **C** l'estampille vectorielle **EV(C)** définie par $\mathbf{sup(EV(e_1), ..., EV(e_j), ..., EV(e_N))}$ c 'est-à-dire que $\mathbf{EV(C)[k]} = \mathbf{sup(EV(e_1)[k], ..., EV(e_j)[k], ..., EV(e_N)[k])}$

La coupure C est cohérente

si et seulement si

$EV(C) = <EV(e_1)[1], ...,EV(e_j)[j], ...,EV(e_N)[N]>$

Appliquons ce résultat aux deux coupures **C** et **C'** de notre exemple. Les estampilles vectorielles des différents événements sont:

sur le site 1	EV(a) = <1, 0 ,0>	EV(b) = <2, 0 ,0>	EV(c) = <3, 0, 1>	EV(d) = <4 ,3 ,1>	
sur le site 2	EV(e) = <1, 1 ,0>	EV(f) = <1, 2, 0>	EV(g) = <1, 3, 0>	EV(h) = <1, 4, 0>	EV(i) = <2, 5, 4>
sur le site 3	EV(j) = <0, 0 ,1>	EV(k) = <1, 2, 2>	EV(l) = <2, 2, 3>	EV(m) = <2, 2, 4>	EV(n) = <2, 4, 5>

On a donc:

- **EV(C) = <2, 2, 1> = sup(EV(b), EV(f), EV(j))** : la coupure est donc cohérente;
- **EV(C') = <2, 4, 4>** alors que **sup(EV(a), EV(h), EV(m)) = <1, 4, 4>** : la coupure n'est donc pas cohérente.