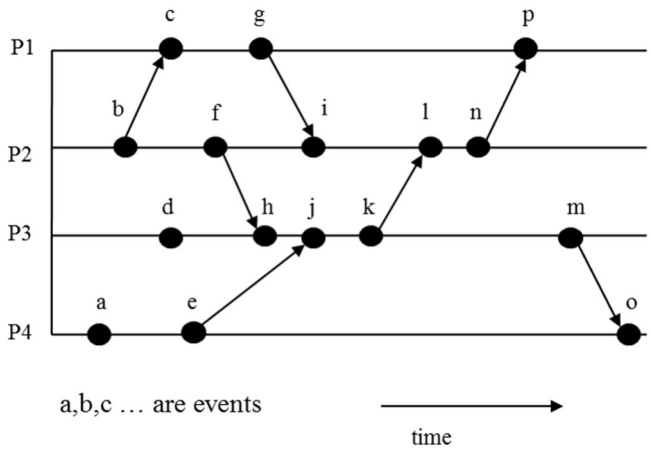
SYNCHRONISATION D'EVENEMENTS, ETATS GLOBAUX ET SNAPSHOT

On rappelle que les supports de cours sont disponibles à
http://mathieu.delalandre.free.fr/teachings/dcomputing.html

1. Horloge logique et vectorielle

1.1. Relation happens-before et propriétés

On considère un cas d'échange suivant entre quatre processus P1, P2, P3 et P4 hébergés sur des machines distantes au sein d'un système distribué. Indiquez les valeurs d'horloge logique pour ces quatre processus sur cet échange, puis celles par horloge vectorielle.



Etablissez ensuite les relations « happens-before » et concurrentes à partir des valeurs d'horloge logique puis vectorielle pour les séquences suivantes d'événement :

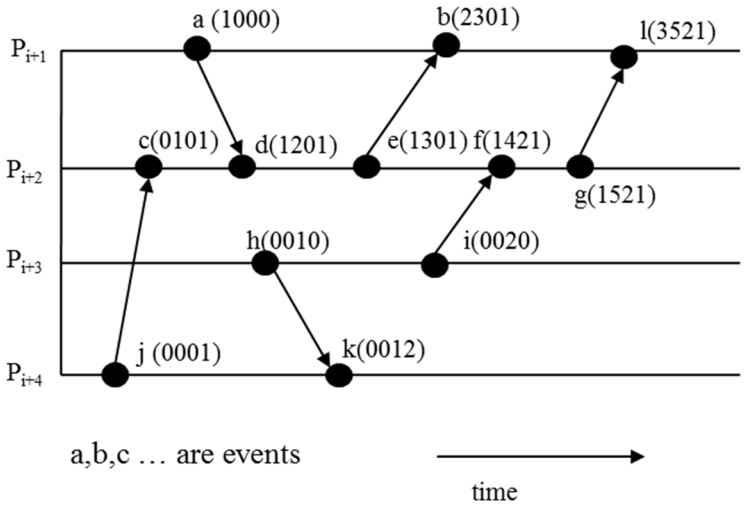
- k, l, m, n, o et p
- g, i, f, h, e, j, k et l

On considère les identifiants P1 < P2 < P3 < P4 entre les processus au sein du système (i.e. P1 est le plus prioritaire). Indiquez l'ordre des événements pour les deux séquences précédentes considérant un ordonnancement total par horloge logique puis vectorielle.

1.2. Méthode de Singhal–Kshemkalyani

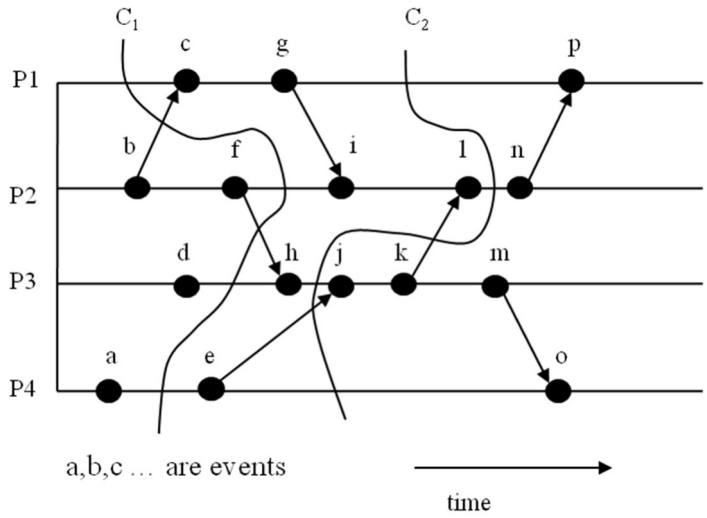
On considère un cas d'échange ci-dessous entre quatre processus P_{i+1}, P_{i+2}, P_{i+3} et P_{i+4} hébergés sur des machines distantes au sein d'un système distribué. Au sein de cet échange, les processus P_{i+1}, P_{i+2}, P_{i+3} et P_{i+4} se synchronisent à l'aide d'une horloge vectorielle via la méthode de Singhal–Kshemkalyani. Les valeurs d'horloge vectorielle sont données sur le diagramme d'échange. On considèrera le processus le P_{i+2} pour le déroulement de la méthode de Singhal–Kshemkalyani.

Etablissez sur les événements c, d, e, f, g de P_{i+2} les vecteurs LU_{i+2} et LS_{i+2}. Définissez ensuite la forme des messages diffusés par P_{i+2} aux évènements e et g.



2. Etats globaux

Soit le diagramme précédent intégrant deux coupes « cuts » C1 et C2.



A partir de ce diagramme, précisez les informations suivantes:

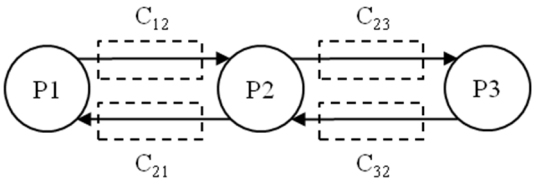
- h_1, h_2, h_3 et h_4 les histoires locales de P1, P2 P3 et P4,
- H l'histoire complète de P1, P2, P3 et P4,
- Pour les coupes « Cuts » C1 et C2: ensembles d'événements, frontières et états de consistance,

Considérant les quatre séquences suivantes d'événement, caractériser chacune de ces séquences en tant que « run » et linéarisation.

- c g f d b h
- f e h i g j
- d b c a h e j
- f g i h e j k l

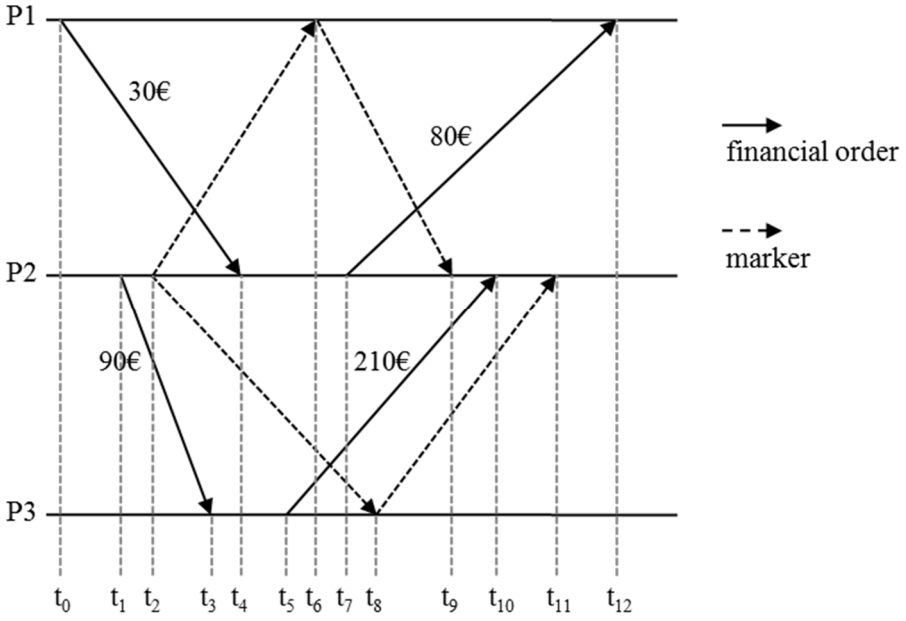
3. Algorithme du snapshot

On considère un cas de synchronisation par algorithme de snapshot entre trois processus P1, P2, P3, afin de déterminer l'état global du système. Ces trois processus sont reliés deux à deux au travers de 4 canaux de communication C21, C12, C23 et C32 tel que défini ci-dessous.



Soit un cas de transaction financière entre P1, P2, P3 sur les montants 30 €, 90 €, 210 € et 80 € tel que défini ci-dessous. A l'initialisation du système, on considère que les montants détenus par P1, P2, P3 sont respectivement de 600€, 150€ et 350€.

Déroulez l'ensemble de la synchronisation t_0-t_{12} en précisant les éléments suivants pour chacun des processus : montants détenus, états des processus et canaux, messages diffusés sur chaque canal. A l'issue de la synchronisation, vérifiez que l'état global du système est cohérent considérant l'état initial à t_0.



Précisez ensuite à quel instant t_n le snapshot prend fin, et déterminer pour les 3 processus P1, P2 et P3 les instants auxquels ils se synchronisent considérant le critère de la bande élastique. Selon vous, à la terminaison du snapshot l'état global reporté est-il synchrone avec l'état d'exécution du système.