

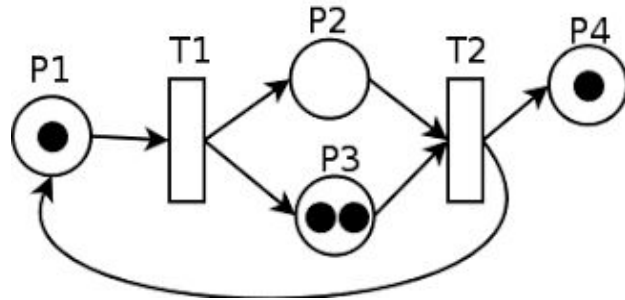
Projet de recherche

Recapitulons !

Sommaire

- A propos de cette semaine:
 - Qu'est ce qu'un réseau de Petri ?
 - Qu'est ce qu'une table de hachage distribuée (DHT) ?
 - Quelques algorithmes en lien avec les DHT
 - La logique temporelle, c'est quoi ?
- Objectifs

Réseau de Petri



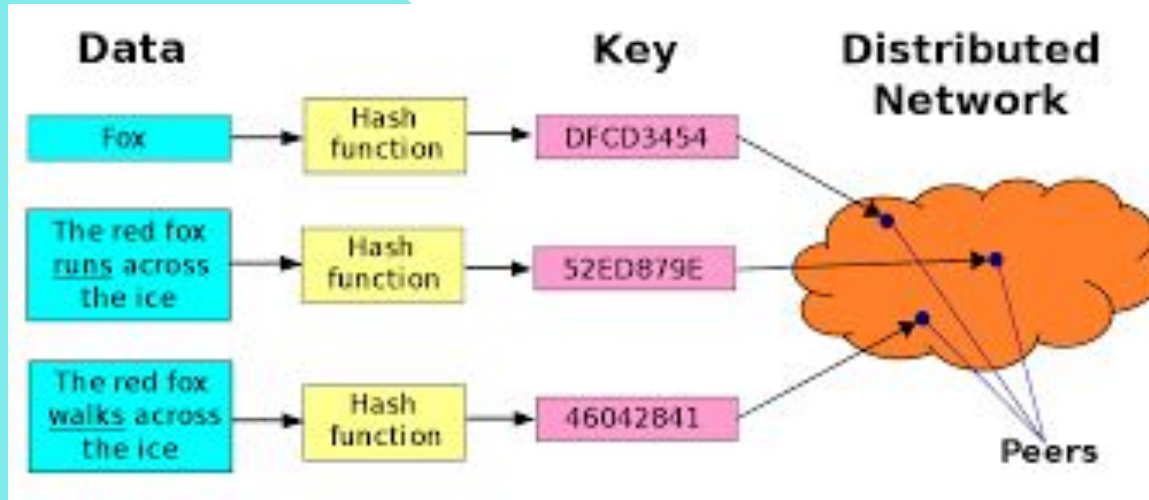
Graphe biparti:

Famille de places et de transitions

Arcs relie uniquement place/transition ou transition/place

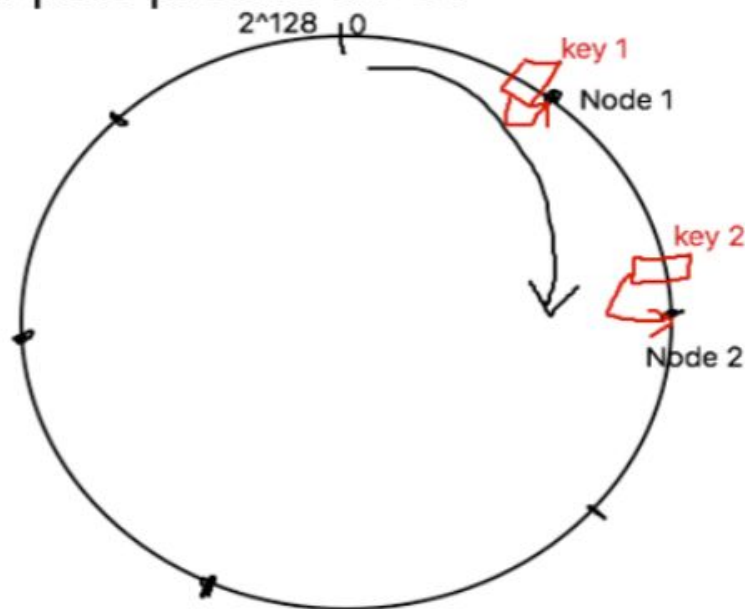
Marquage donne l'état du système

Table de hachage distribuée



Routage dans un Réseau de Pastry

-On associe à chaque valeur une clé $c1$ à 164 bits, cette valeur est stockée sur un nœud dont la clé $c2$ est la plus proche de $c1$



Ajout d'un noeud dans un réseau de Pastry

- Le noeud X veut joindre le réseau -> une table de hachage associe une clé de 128 bits à X (soit 32 symboles hexadécimaux)
- X localise un noeud proche (proche par rapport à sa clé) -> le noeud A
- Le noeud A diffuse la requête de X (de joindre le réseau) au noeud Z le plus proche de X que A ait dans sa table de routage.
- Les noeuds A,Z et ceux dans le chemin entre A et Z envoient leur table de routage à X
- X utilise ces tables pour initialiser sa propre table de routage.



Longueur 15: chaque signe a 16 valeurs possibles
- le signe correspondant a la clé du noeud

Comment atteindre un noeud en connaissant sa clé ?

- Soient 2 noeuds A et B tel que A veuille envoyer une requête au noeud B
- La requête va utiliser un algorithme "Greedy": A envoie la requête vers le noeud X1 dont la clé se rapproche le plus de B d'après sa table de routage
- Le noeud X1 l'envoie vers le noeud X2 dont la clé se rapproche le plus de B d'après sa table de routage ... jusqu'à atteindre B.
- Si la requête atteint un noeud Xn qui n'est pas B et n'a pas de voisin dont la clé se rapproche de B, alors B n'est pas sur le réseau

Content Addressable Network

- Réseau de recouvrement de type table de hachage basé sur des coordonnées spatiales.
- Chaque CAN a un unique nom DNS, chaque nœud est responsable de sa zone dans l'espace.
- L'espace est partagé en rectangles on parle de topologie hypercubique.

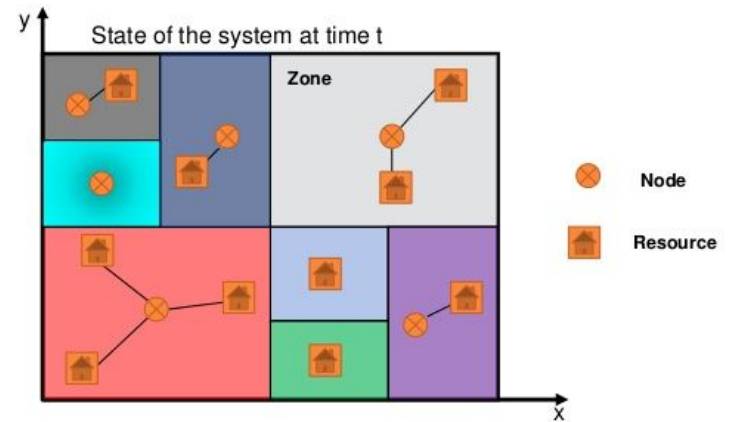


Fig:2 dimensional space with a key mapped to a point (x,y)

Algorithme de jointure:

Composé de 3 étapes:

- Amorçage (bootstrapping):

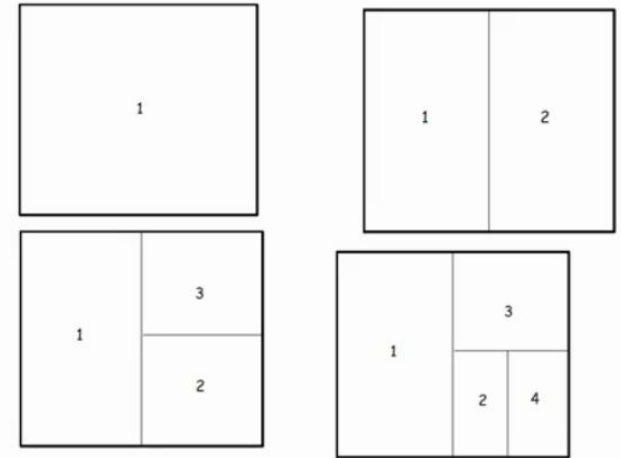
Trouver @IP d'un noeud c du CAN, puis déterminer une liste de noeuds actifs à l'aide de c

- Rechercher zone:

c choisi aléatoirement un noeud b ainsi qu'un point p appartenant au CAN qui sera son identifiant.

c envoie une requête d'adhésion à P
Si P accepte alors zone scindée en deux.

CAN- IDEA- Example



- Rejoindre le routage:

Les noeuds sont mis à jours avec les nouvelles adresses.

Logique temporelle

- Logique Temporelle Linéaire (LTL)

Logique temporelle qui exprime des propriétés sur les exécutions des systèmes, contrairement aux réseaux de Petri, il n'y a pas de relations de transitions explicites ou d'états initiaux.

- Logique du Temps Arborescent (CTL) :

Logique temporelle où les modèles ont une structure arborescente symbolisant les différents chemins possibles car le futur est indéterminé.

Objectifs pour la prochaine séance

- S'informer sur plus d'algorithmes de DHT, pas seulement Pastry et CAN, aussi Chord ou Kademlia par ex
- Réfléchir à comment modéliser des DHT avec les réseaux de Petri
- S'informer sur la logique temporelle linéaire et la logique du temps arborescent

Sources:

http://www.morere.eu/IMG/pdf/cours_petri2.pdf<http://homepages.laas.fr/francois/RdP/rdp.pdf>

<https://www.emse.fr/~augusto/enseignement/icm/gis1/UP2-2-RdP-slides.pdf>

https://fr.wikipedia.org/wiki/R%C3%A9seau_de_Petri<http://www.specialautom.net/automatisme/cours%20version%20revue%20pour%20le%20site.pdf>

<http://eric.univ-lyon2.fr/~jdarmont/docs/old/exam0.pdf>

http://www.site.uottawa.ca/~nrahmani/CEG4566_H13/notes_cours/Annexe_1.pdf

https://fr.wikipedia.org/wiki/Table_de_hachage_distribu%C3%A9

https://fr.wikipedia.org/wiki/Table_de_hachage

<https://www.ietf.org/proceedings/65/slides/plenaryt-2.pdf>

<http://cedric.cnam.fr/~sailhanf/enseignements/p2p/dht.pdf>

<https://www.irif.fr/~jch/software/bittorrent/dht-20130214.pdf>

https://www.researchgate.net/publication/30498632_Routing_in_Content_Addressable_Networks_Algorithms_and_Performance