ENREGISTREMENT D'UNE TRACE

• Objectifs :

• Reconstruire la séquence des messages échangés pour faire une analyse post mortem ou pour préparer la réexécution d'une situation à analyser.
Il faut donc conserver la dépendance causale entre les événements dépendants et indiquer les événements causalement indépendants.

• On date chaque événement e du système avec une méthode de datation causale (l'horloge causale). Soit D(e) la date ainsi fournie.
Elle permettra de reconstituer la trace si et seulement si :

$$a \rightarrow b \Leftrightarrow D(a) < D(b)$$
$$a \,||\, b \Leftrightarrow \text{non } (D(a) < D(b)) \text{ et non } (D(b) < D(a))$$

C'est la condition forte des horloges car elle inclut : $D(a) < D(b) \Rightarrow a \rightarrow b$

Rappel : condition pour que deux événements a et b soient concurrents, ou encore causalement indépendants :

$$a \,||\, b \Leftrightarrow \text{non } (a \rightarrow b) \text{ et non } (b \rightarrow a)$$

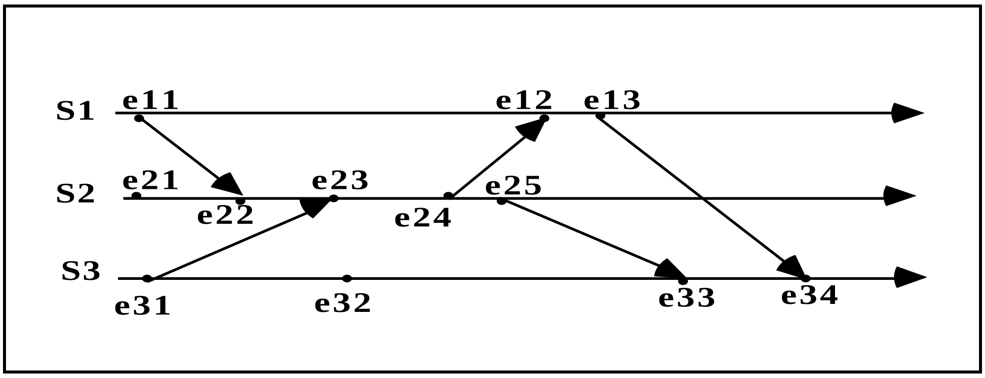
UNE METHODE DE DATATION CAUSALE

les historiques

Rappel : passé d'un événement e

• Le passé (ou historique) d'un événement e, c'est par définition :

hist(e) = e $\cup$ ensemble des événements e' tels que e' $\rightarrow$ e



hist(e33) = {e11 e25 e24 e23 e22 e21 e31 e32 e33}

Idée : utiliser le passé de e pour la datation car le passé permet de représenter la dépendance causale :

$$a \rightarrow b \Leftrightarrow a \in \text{hist}(b)$$

$$a \,||\, b \Leftrightarrow (a \notin \text{hist}(b)) \text{ et } (b \notin \text{hist}(a))$$

Gros inconvénient : la taille de hist(e)

Remède : on observe que, pour définir hist(e), un événement par site suffit.

HORLOGES VECTORIELLES

(Fidge, Mattern 1988)

• **Projection de hist(e) sur Si :**

$$\text{hist}_i(e) = \{\ a \in \text{hist}(e) \mid a \in S_i\ \}$$

• **Propriété :** $e_{i,k} \in \text{hist}_i(e) \Rightarrow$ **pour tout** $j < k$: $e_{i,j} \in \text{hist}_i(e)$

Si on indice les événements de $\text{hist}_i(e)$ et si un événement avec un indice k appartient à $\text{hist}_i(e)$, alors tous les événements d'indice inférieur à k font aussi partie de $\text{hist}_i(e)$.

• **Alors un seul entier suffit pour représenter $\text{hist}_i(e)$, c'est le nombre d'événements de $\text{hist}_i(e)$.**

Comme $\text{hist}(e) = \cup\, \text{hist}_i(e)$**, on représente tout hist(e) avec un vecteur V(e)**

$$1 \le i \le n : V(e)[i] = k \text{ tel que } e_{i,k} \in \text{hist}_i(e) \text{ et } e_{i,k+1} \notin \text{hist}_i(e)$$

V(e)[i] = nombre d'événements de Si "connus de e"

(i. e. connus sur le site de e immédiatement après l'occurrence de e)
(nombre d'événements de l'historique de e qui sont localisés sur Si)

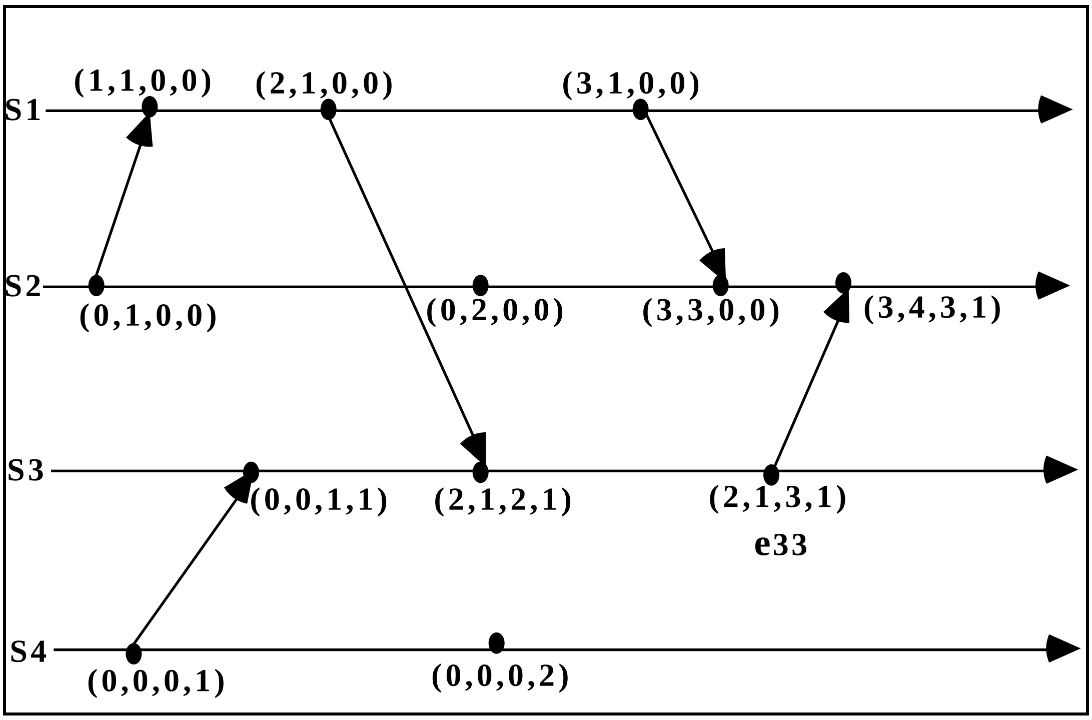
REALISATION DES HORLOGES VECTORIELLES

On associe une horloge vectorielle V_i à chaque site Si

- **Initialement Vi = (0,...,0)**
- **A chaque événement à dater local à Si, on fait $V_i[i] := V_i[i] + 1$**
- **Chaque message m porte une estampille Vm (Vm = Vi de l'émetteur)**
- **A la réception de (m, Vm) par un site Si, on enrichit l'historique connu par Si avec l'historique transporté par m :**

  $V_i[i] := V_i[i] + 1$

  $V_i[j] := \max(V_i[j], Vm[j])$ **pour tous** $j = 1,...n, j \neq i$



- **Autre façon de connaître l'"heure" de e33:**

  l'histoire de e comprend

  2 événements sur S1

  1 événement sur S2

  3 événements sur S3

  1 seul événement sur S4

PROPRIETES DES HORLOGES VECTORIELLES

• Relation d'ordre partiel sur les horloges vectorielles :

$V \leq V'$ défini par : quel que soit i, $V[i] \leq V'[i]$

$V < V'$ défini par $V \leq V'$ et $V \neq V'$

$V \,||\, V'$ défini par $\neg (V < V')$ et $\neg (V' < V)$

Les horloges vectorielles représentent <u>exactement</u> la dépendance causale : pour tout a, b

$$a \rightarrow b \Leftrightarrow V(a) < V(b)$$
$$a \,||\, b \Leftrightarrow V(a) \,||\, V(b)$$

• Les horloges vectorielles sont "denses" :

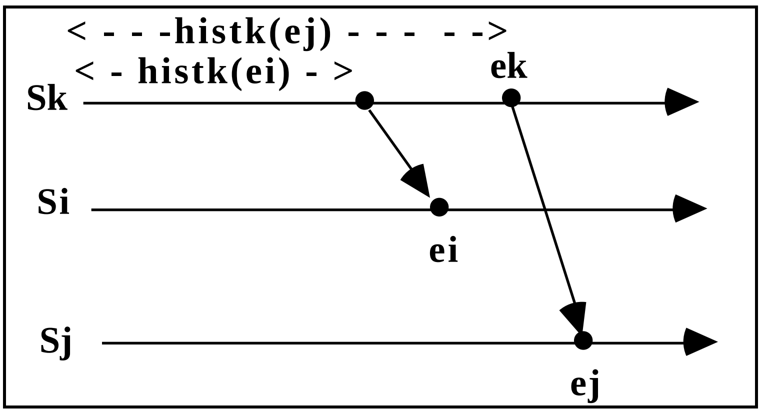
Soit $ei \in Si$, $ej \in Sj$,

si $V(ei)[k] < V(ej)[k]$, pour $k \neq j$, alors il existe ek sur Sk tel que

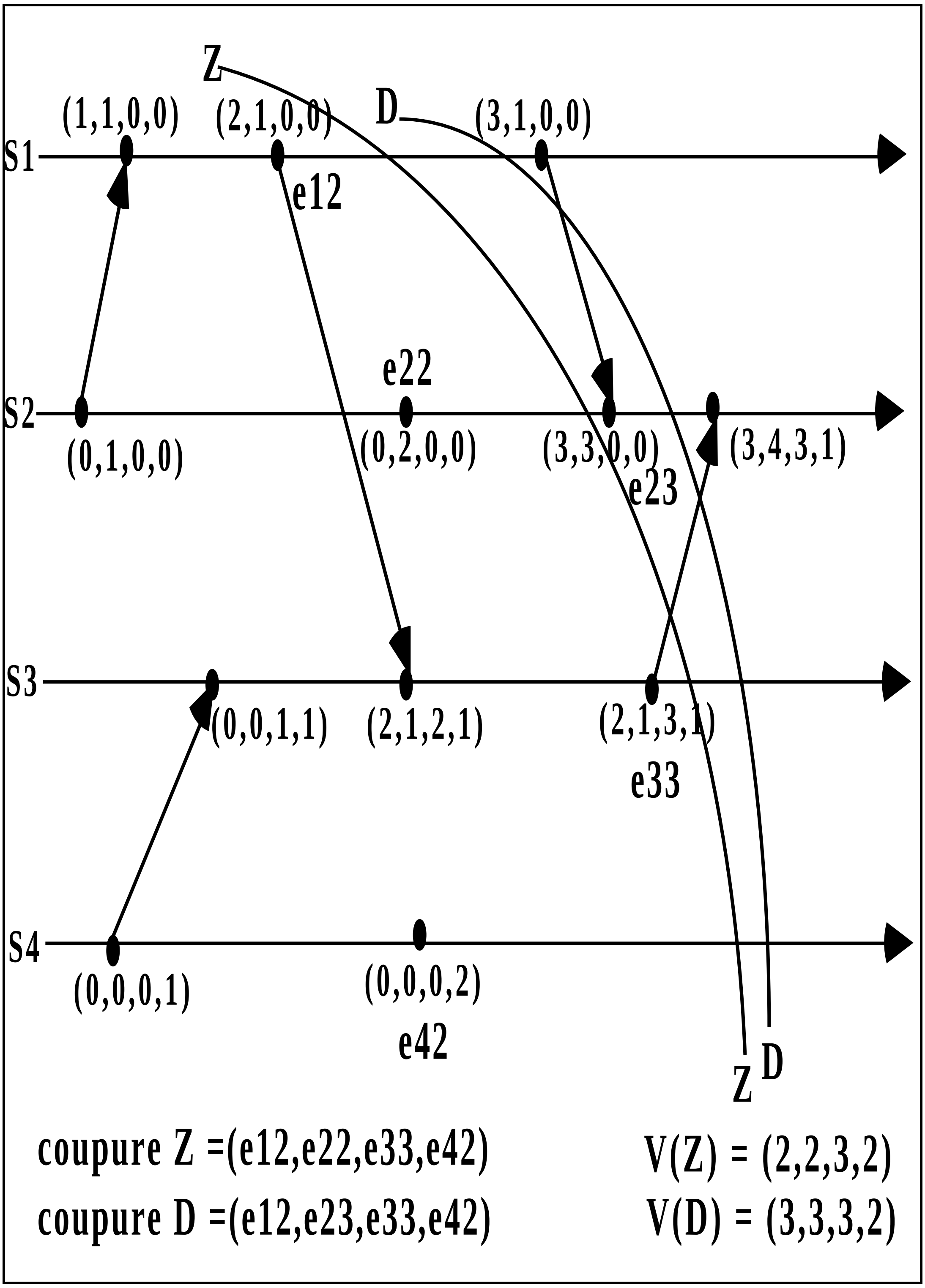
$\neg(ek \rightarrow ei)$ et $(ek \rightarrow ej)$

En effet $V(ei)[k] = |hist_k(ei)|$ **= nombre d'événements de hist(ei) sur Sk**

$V(ej)[k] = |hist_k(ej)|$ **= nombre d'événements de hist(ei) sur Sk**



HORLOGES VECTORIELLES ET COUPURES COHERENTES



• **Date d'une coupure C = (c1,c2,...,cn)**

V(C) = sup (V(c1),...,V(cn))

soit pour tout i, V(C)[i] = sup (V(c1)[i],...,V(cn)[i])

La coupure est cohérente si et seulement si

V(C) = (V(c1)[1],...,V(cn)[n])

V(Z) = (2, 2, 3, 2) = (V(e12)[1], V(e22)[2], V(e33)[3], V(42)[4])

V(D) = (3, 3, 3, 2)

(V(e12)[1], V(e23)[2], V(e33)[3], V(42)[4]) = (2, 3, 3, 2)

• Z est cohérente, D ne l'est pas

DIFFUSION AVEC ORDRE CAUSAL

(Birman, Schiper, Stephenson 1990)

• Utilisation des horloges vectorielles pour garantir la causalité

• Si un message arrive trop tôt pour la causalité, on le fait attendre

• On ne compte que les diffusions :

1) avant diffusion de m, le site Si exécute $V_i[i] := V_i[i] + 1$

2) estampille de m par Si : Vm = Vi

3) à la réception sur Sj de (m, Vm) diffusé par Si, on attend que :

a) toutes les diffusions précédentes de Si soient arrivées sur Sj

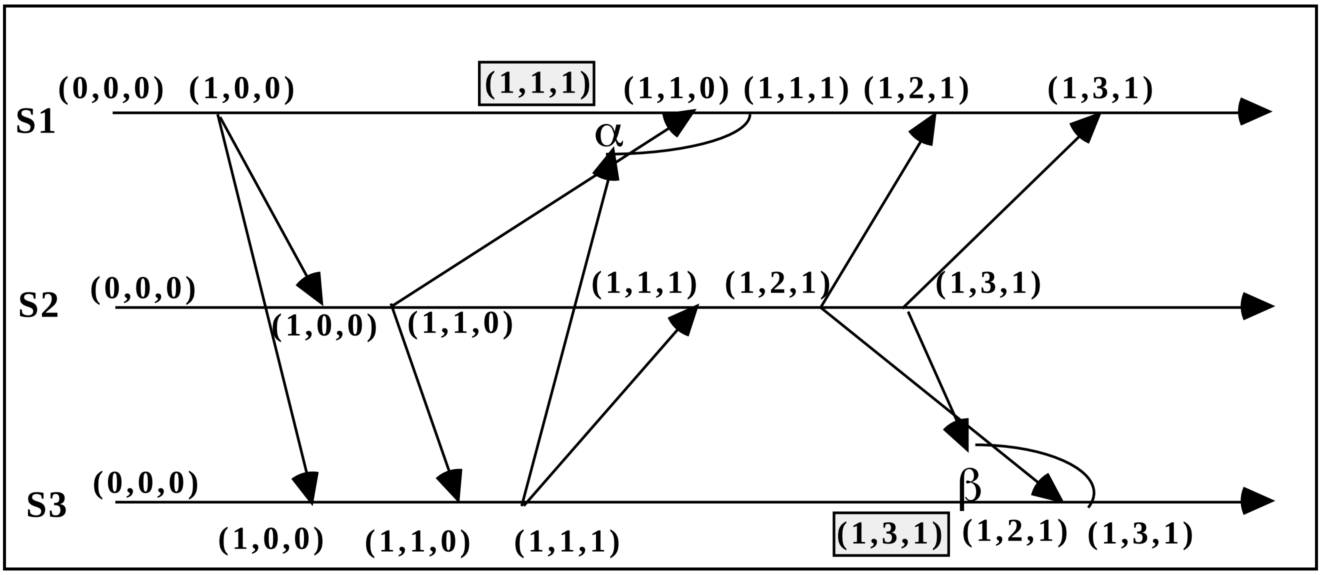
soit Vj[i] = Vm[i] - 1

b) toutes les diffusions antérieures à m et reçues sur Si aient été aussi reçues par Sj

soit pour tous k ≠ i, Vj[k] ≥ Vm[k]

4) après remise de m, on enregistre l'historique connu grâce à m

soit Vj := max (Vj, Vm)



en α : V1 = (1,0,0) et Vm = (1,1,1) soit V1[3] = Vm[3] - 1; <u>V1[2] < Vm[2]</u>

en β : V3 = (1,1,1) et Vm = (1,3,1) soit <u>V3[2] ≠ Vm[2] - 1</u>; V3[1] ≥ Vm[1]