Une approche expérimentale pour l'algorithmique parallèle sur grappes et grilles de calcul

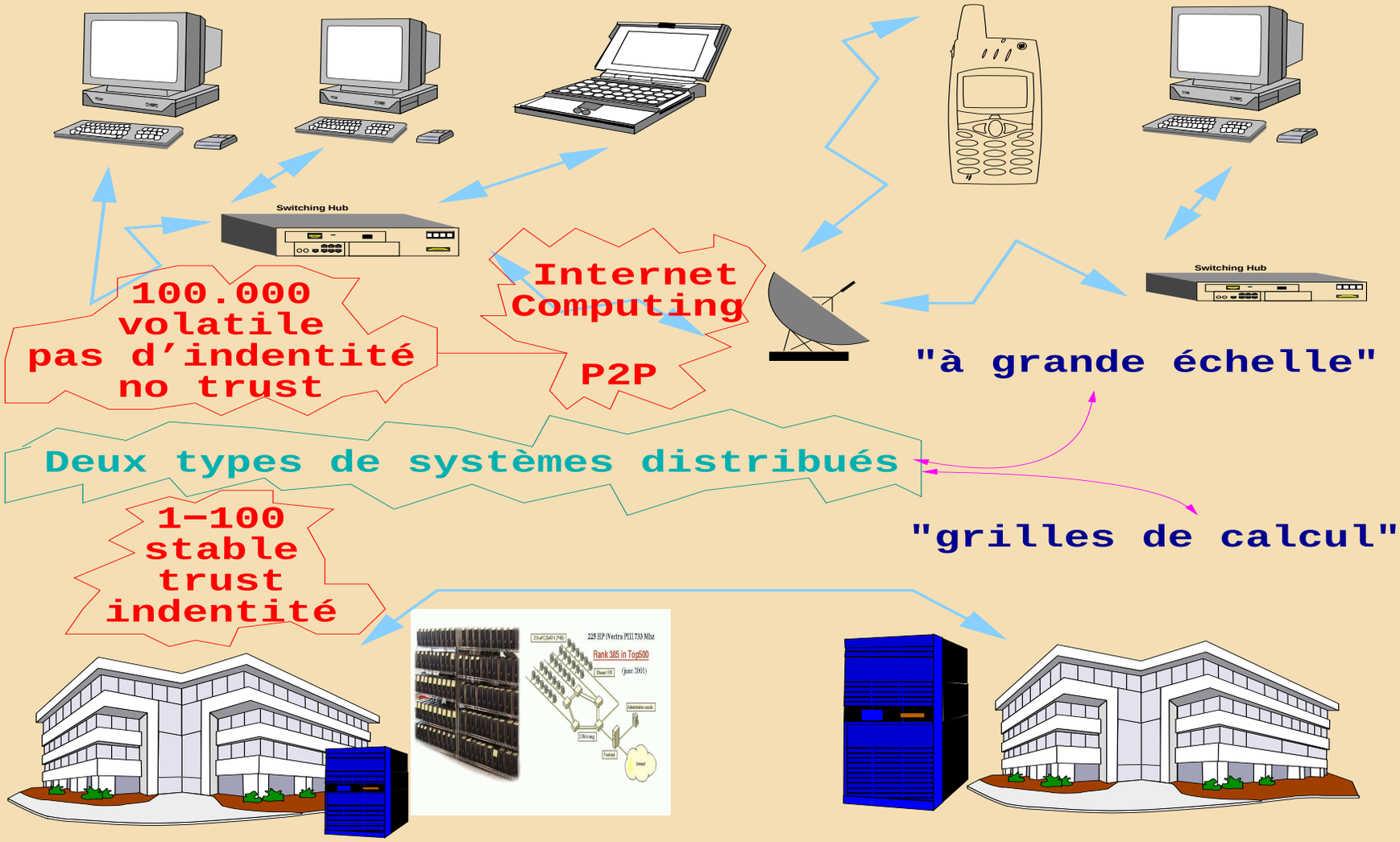
Christophe Cérin

cerin@laria.u-picardie.fr

PLAN

- Contexte, démarche et positionnement
- Archi processeurs et cache
- Algorithmes mémoire commune
- Le tri sur clusters
  - homogène et hétérogène
- I/O hautes performances
  - Représentations sur disques
  - Collecte, analyse d'événements

CLASSIFICATION SYSTÈMES DISTRIBUÉS



PROJETS SCIENTIFIQUES

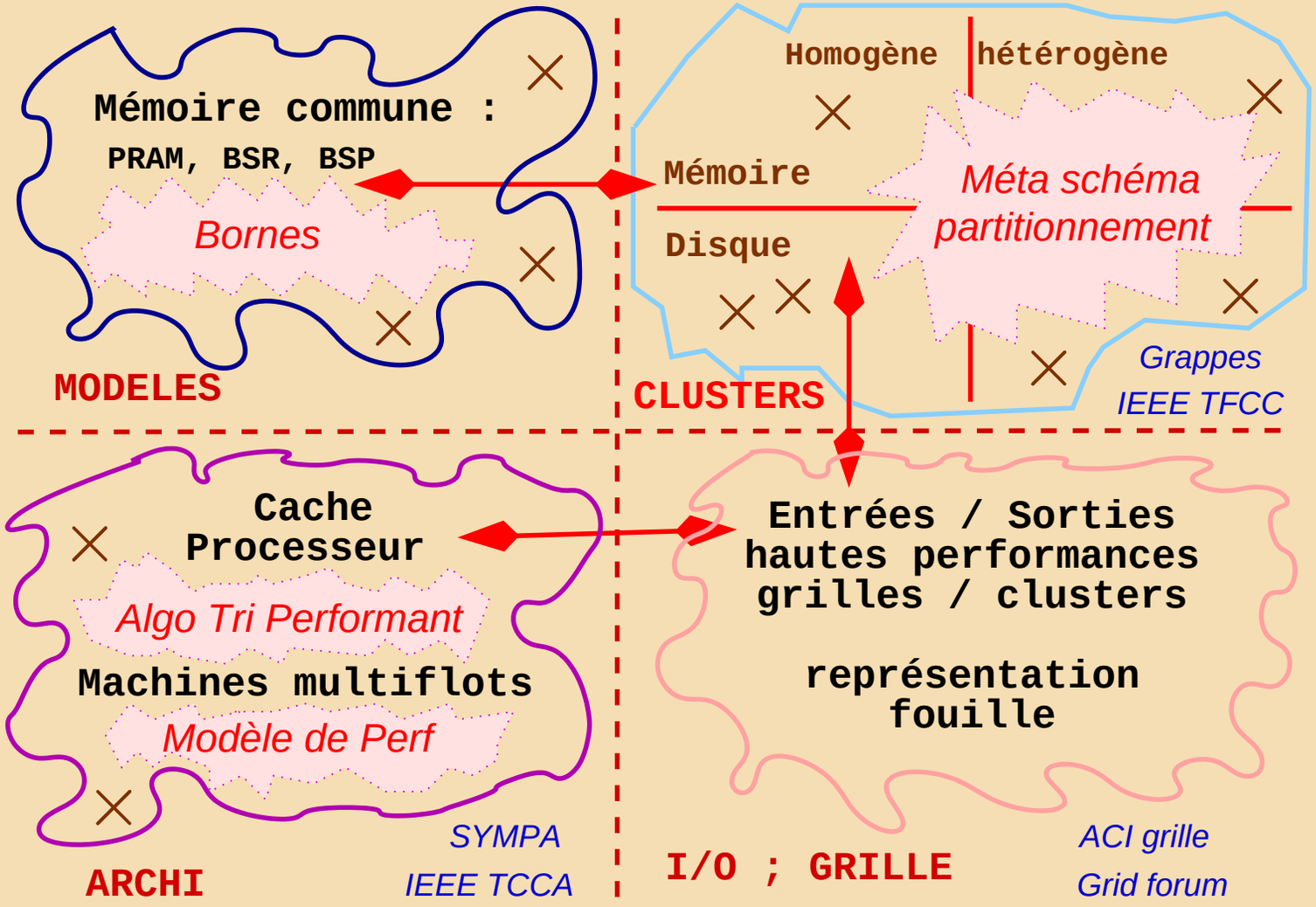
- Définir les concepts et outils
  - résoudre efficacement les problèmes avec de grandes masses de données
  - grappes et grilles
- Démarche à double sens
  - expérimentale
  - garanties algorithmiques prouvées

DÉMARCHE INTÉGRATRICE

- Établir un plan d'intégration

1) Comparaison
   - ✘ mise en évidence des conflits
2) Mise en conformité
   - ✘ résolution des conflits
3) Fusion des schémas
4) Restructuration
   - ✘ amélioration du schéma global

THÉMATIQUES SCIENTIFIQUES



ARCHITECTURE MACHINES MULTIFLOTS

1- ARCHITECTURE MACHINES MULTIFLOTS

Cray MTA :

* 128 threads / proc
* up to 8 conc. mem ref / proc
* no cache

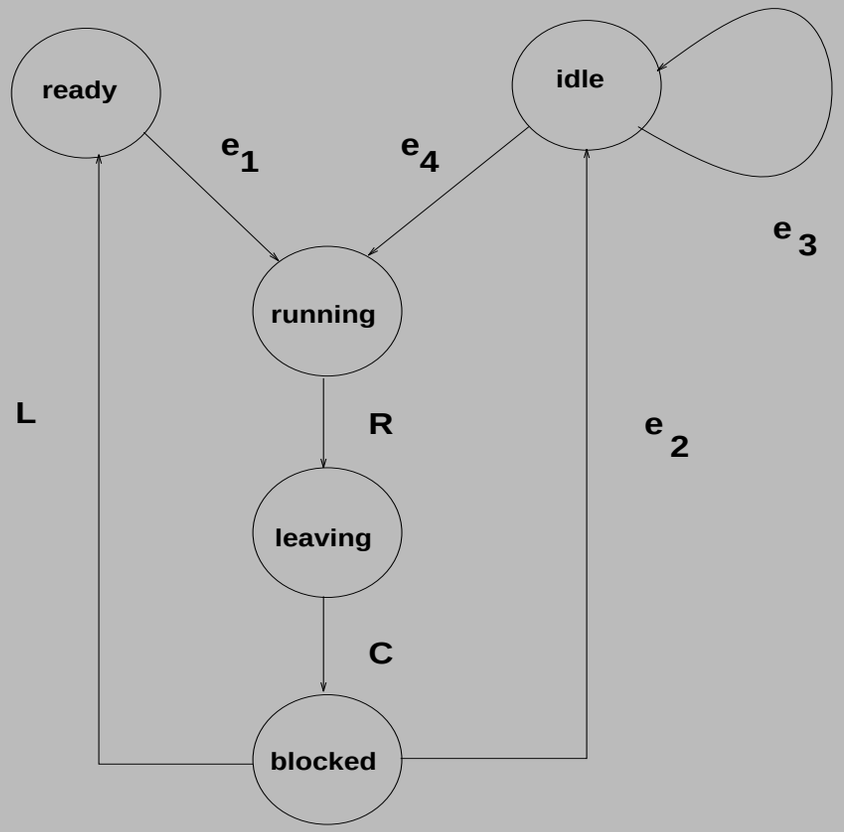
Evaluation :

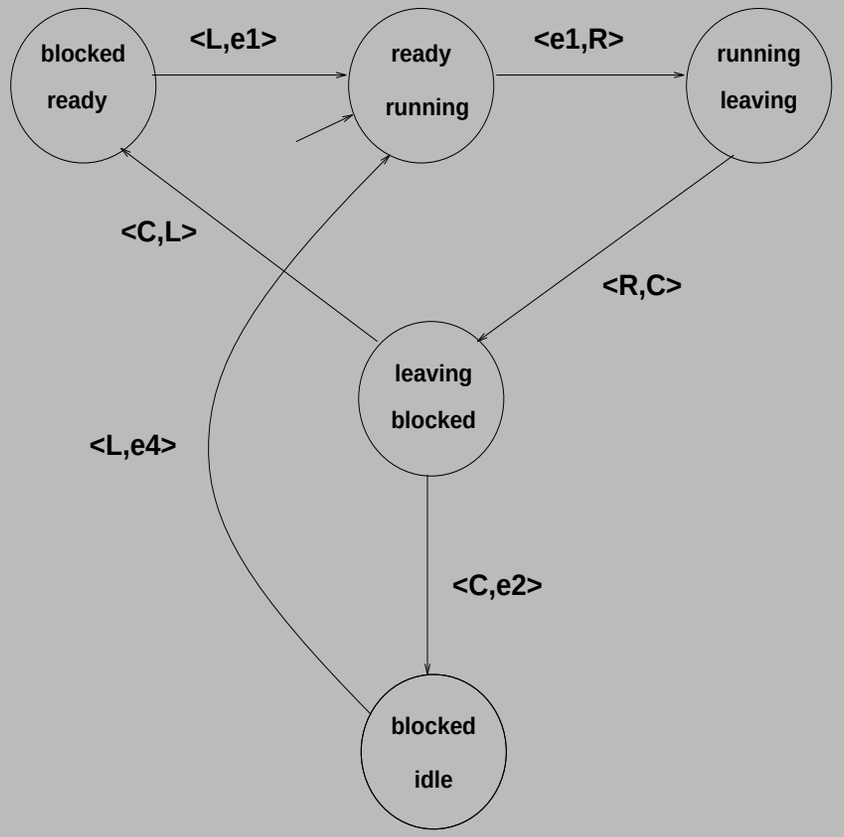
- déf de mesures : max, min de parallélisme (mot puis langage)
- Résultat : mesures calculables par induc. hauteur étoile expr. régulière
- pas de proba. (nouv. approche)

EXEMPLE

2 threads - 1 UC

Prod n-1 fois





ARCHITECTURE
IMPACT DU CACHE L1 SUR TRI

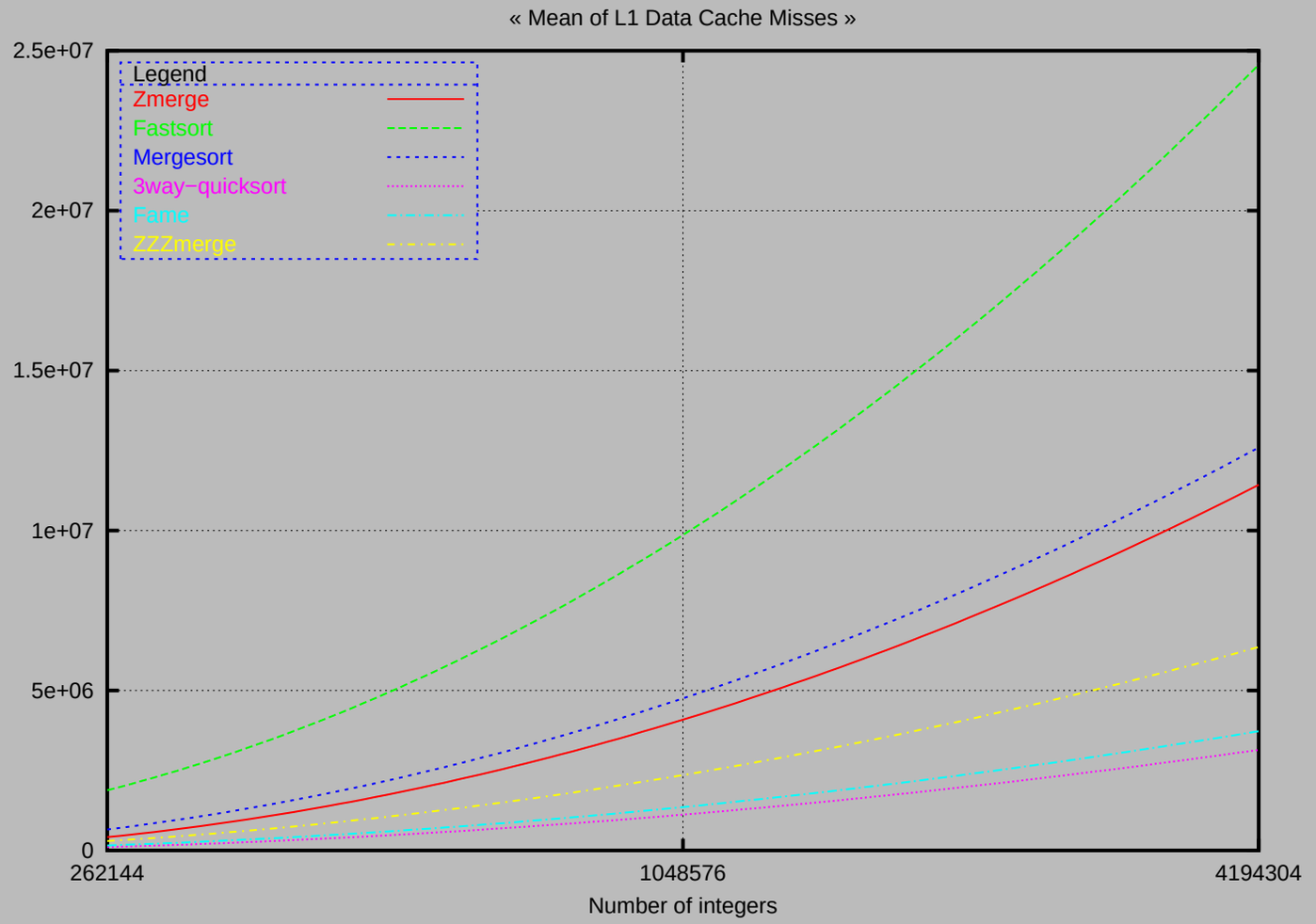
2- HIÉRARCHIE MÉMOIRE : CACHE

- **Objectifs** : impacts du cache sur les performances des tris séq.
- Dualité cache / I/O - borne en termes d'I/Os pour le tri disque
- Quelles techniques de mesure ? (simulation ou compteurs de perf.)
- Algorithmes
  - ✘ localité spaciale et temporelle
  - ✘ compteurs de perf

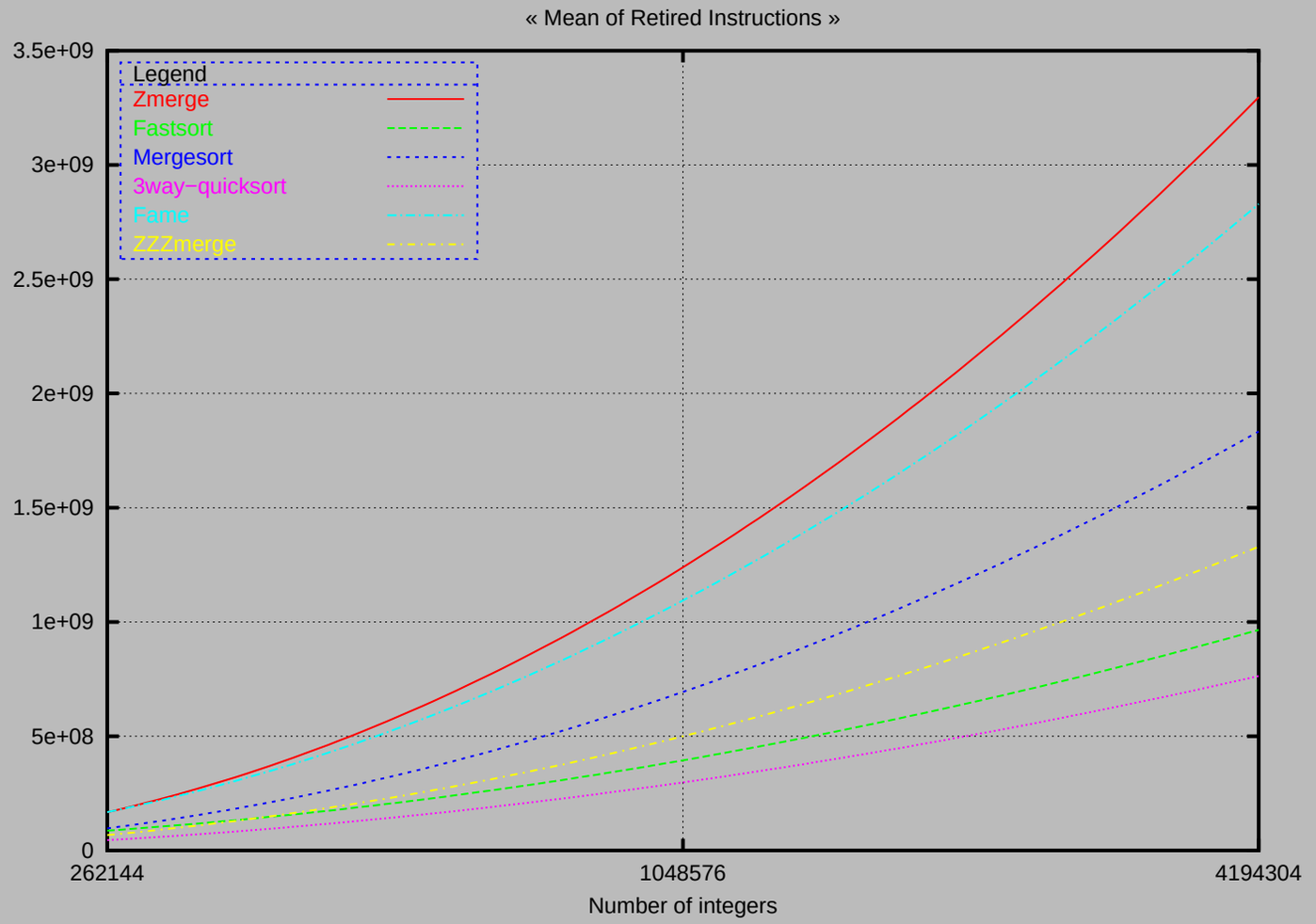
SÉLECTION DES ALGORITHMES

- FastSort (n log log n), FAME, 3-way-Quicksort, mergesort.
- Élimination de heapsort.
- Nouvel Algo. ZZZmerge
- Principe : mergesort
- RÉSULTATS
  - éval. exacte du nb de défauts ZZZmerge
  - Gains entre 9% et 15% de ZZZmerge

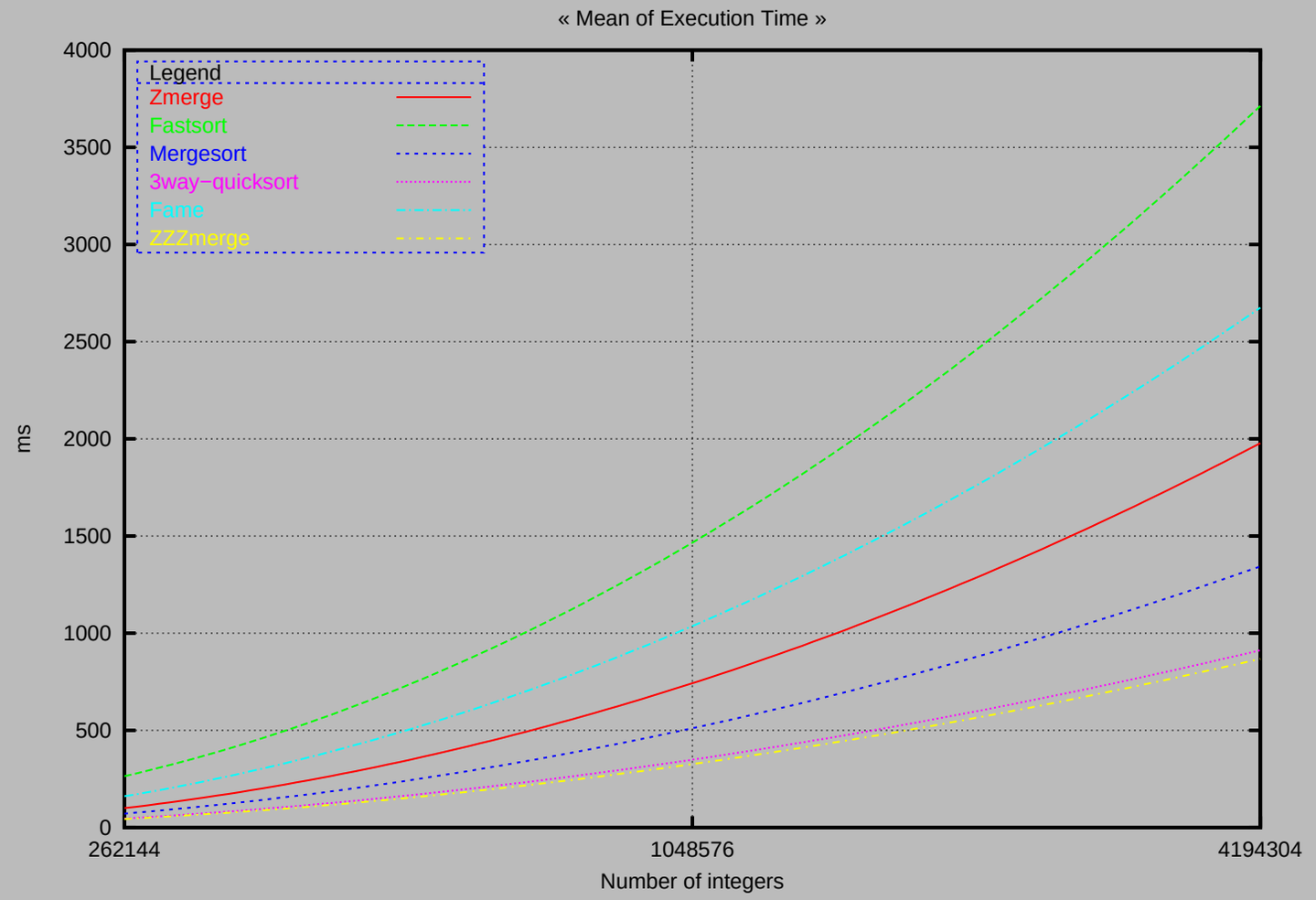
OBSERVATIONS DU NB DE DÉFAUTS L1



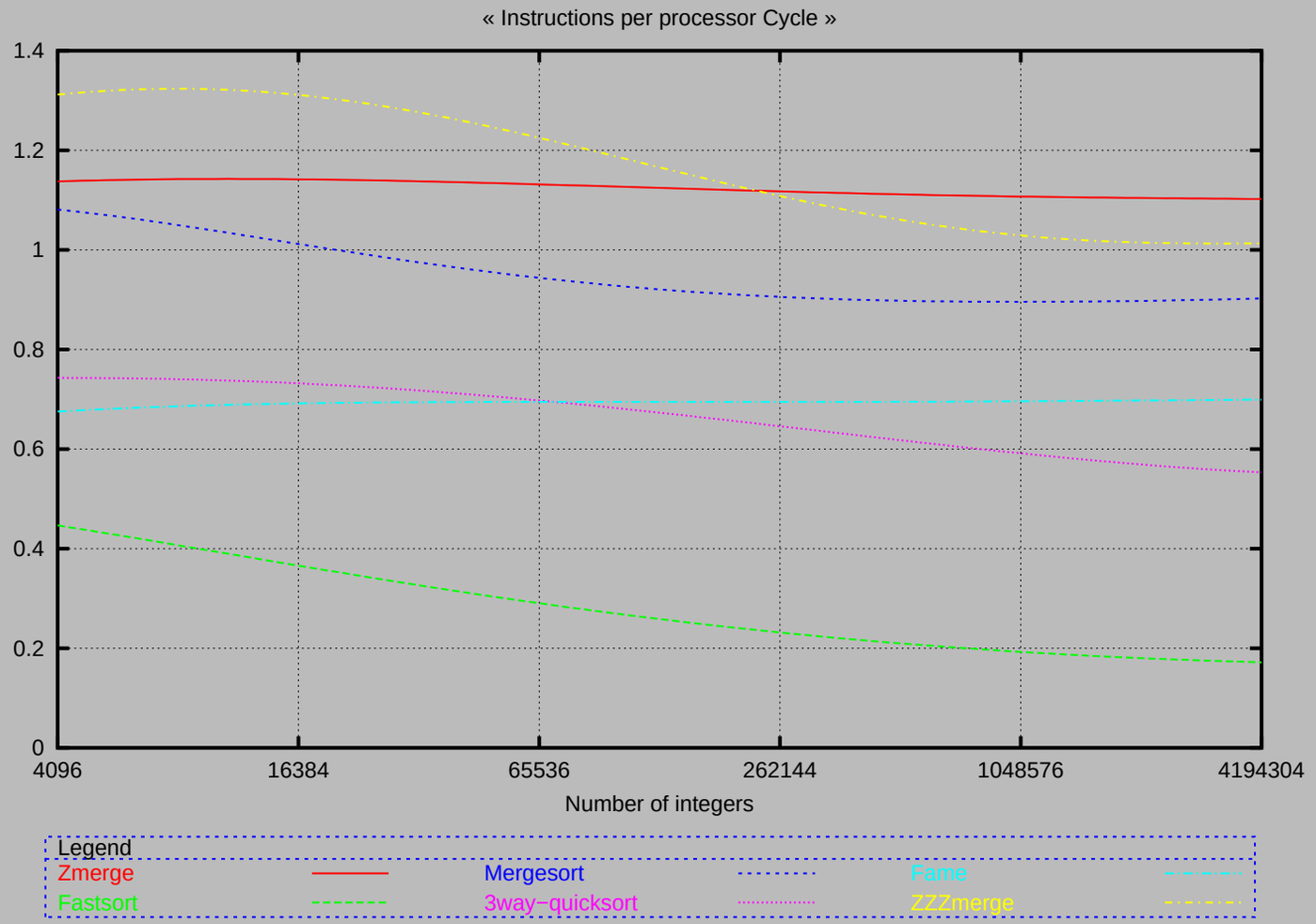
OBSERVATIONS RETIRED INSTTRUCTIONS



OBSERVATIONS TEMPS D'EXÉCUTION



OBSERVATIONS IPC



CONJECTURE

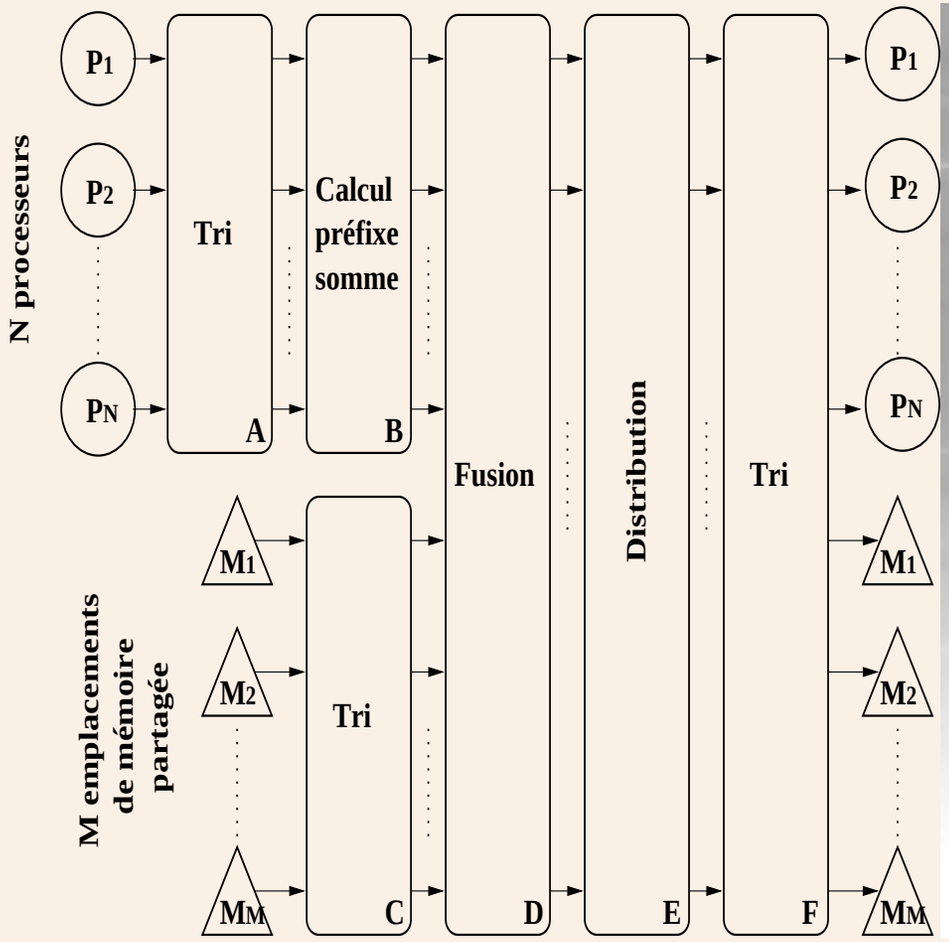
- *“**Si** X a une IPC au moins 2 fois plus grand que Y*
  mais qu’il a un nb de défauts de cache L1 et un nb d’inst. exécutées 2 fois plus important que Y,
  alors X a quand même une exécution en temps meilleure que Y.”
- **Perspectives : Paral. dans les circuits**
  - ✔ **repenser les algos et les compilateurs**
  - ✔ **tri : gain encore possible**

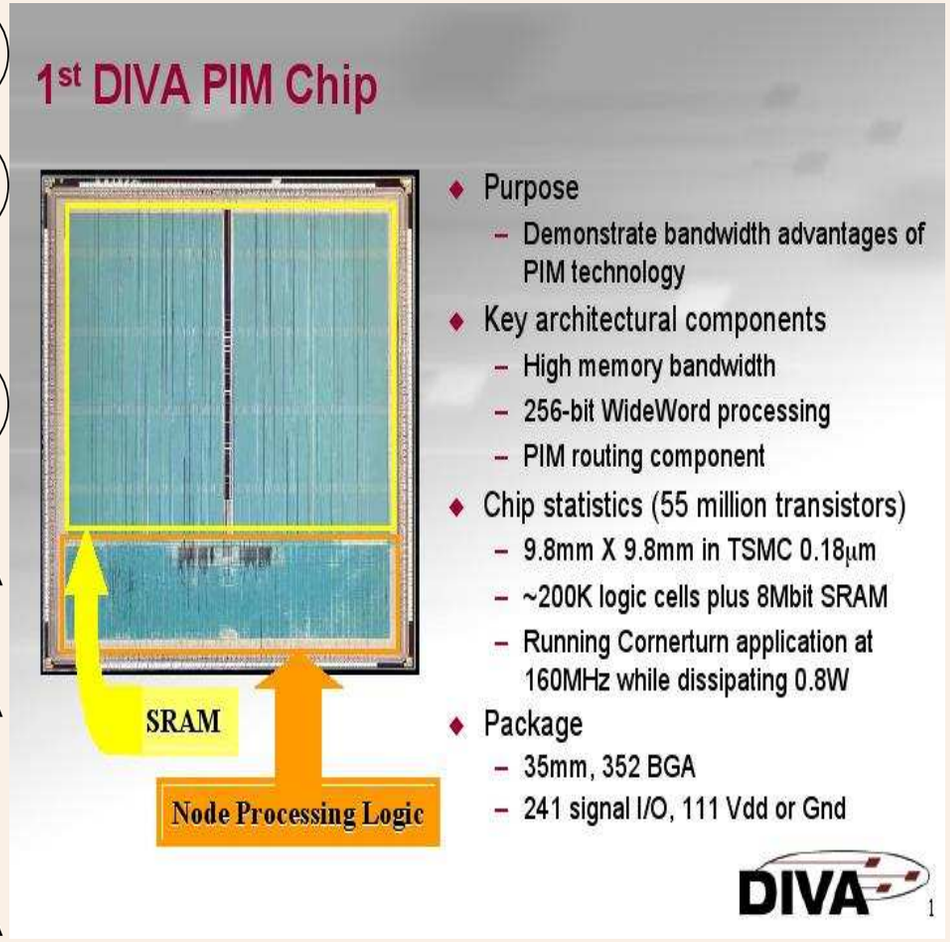
ALGORITHMIQUE
MEMOIRE COMMUNE

MODÈLE PRAM

- **PRAM :** Parenthesis matching -> éval expr. arith.
  - Atteinte de la borne + présentation propice à une impl. message passing.
- **BSR :** segment somme maximale (1-D, 2-D).
  - Atteinte borne avec moins d'inst BSR.
  - Nouvel Inst. BSR
  - Règles BSR -> BSP (compilateur).

BSR / PROCESSOR IN MEMORY (PIM)



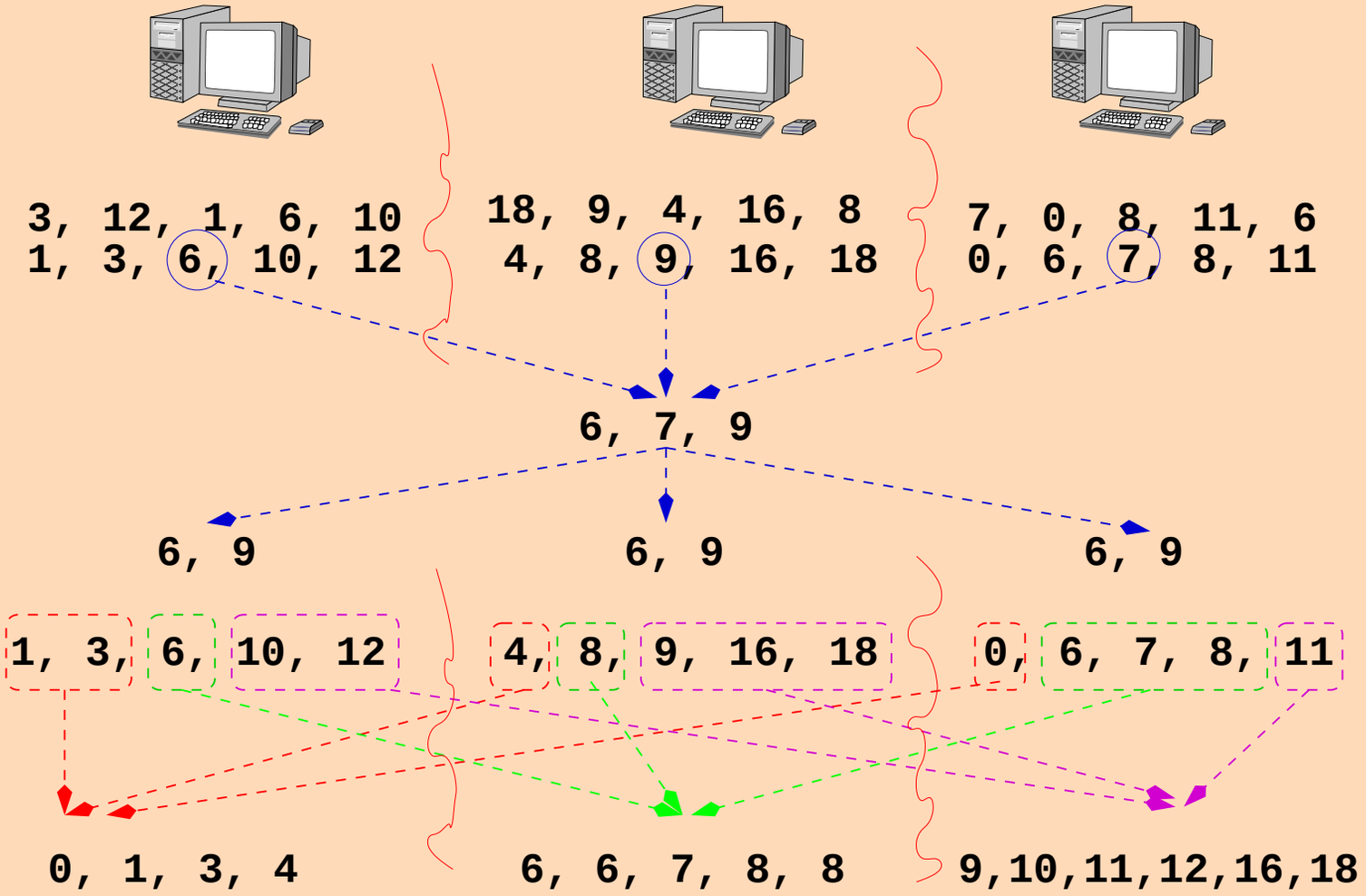


Sort: $r_j = \Sigma 1 | x_i \leq x_j$

ALGORITHMIQUE TAI SUR CLUSTERS

- Problème difficile
  - ✘ Accès mémoire imprédictible
  - ✘ Absence garanties : NowSort, NAS 2.3
  - ✘ #RAM $\subseteq$ Taille : Minute Sort
- Objectifs multiples
  - Schéma générique
    - mémoire + disque
    - Hétérogénéité (jamais étudié)
  - Garanties sur Time et Work ($\forall n$)
  - Utilisant le moins de RAM -> non dédié.
  - Reproductibilité des expériences.

MÉTA-SCHÉMA : ÉCHANTILLONNAGE

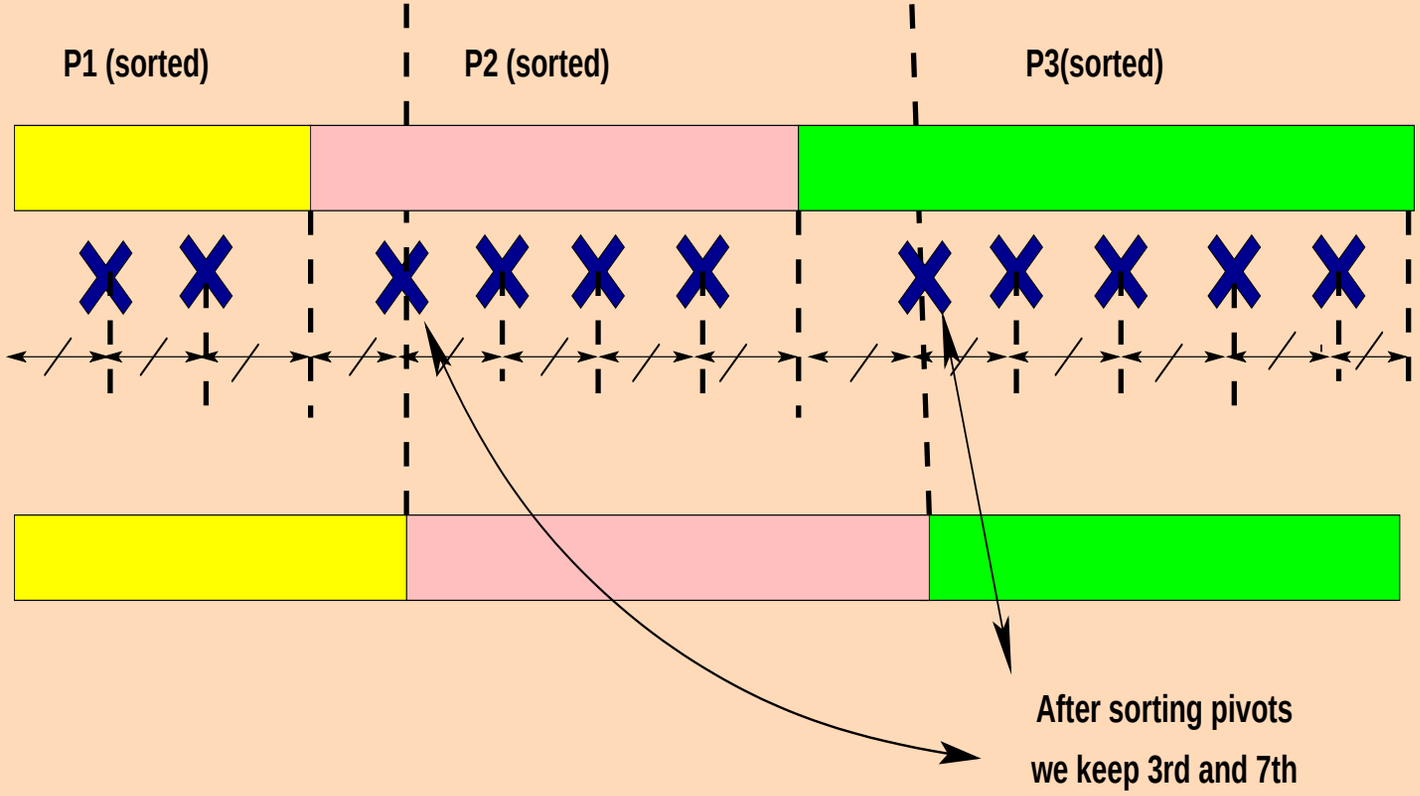


GÉNÉRALISATION DES RÉSULTATS CLASSIQUES

- **Hétérogénéité**
  - Procs à des vitesses différentes
  - Paramétrage du code
- Reproductibilité des expériences
  - 8 tests possibles ; Codes source ;
- **RÉSULTATS**
  - hétérogène ou pas,
  - mémoire ou disques
  - garanties formelles : charge, T d'exé.

LES CLÉS DU SUCCÈS (1/2)

- **Avec tri initial : aucun proc ne traite plus de 2 fois ce qu'il avait au départ.**
- **Exemple en hétérogène :**



LES CLÉS DU SUCCÈS (2/2)

- **Sans tri initial et cas homogène**

  $(p.k-1)$ **pivots partitionnent l'entrée en** $p.k$ **morceaux telle que la taille du plus grand morceau est plus petite ou égale à** n/p **avec la probabilité au moins égale à**

$$1 - 2p\left(1 - \frac{1}{2p}\right)^{p.k}$$

- **Amélioration de l'équili. pour cas hétérogène : simulation de + de processeurs** $(p \rightarrow p\prime = \sum_{i=1}^{p} perf[i])$ **et réutilisation du TH.**

RÉSULTATS TRIS DISQUES HÉTÉROGÈNES

Critère	H–PSS	H–PSRS	H–PSOP
Tri initial	Non	Oui	Non
Tri final	Oui	Merge	Oui
Nb Candidats	6p' log p'	p' (p–1)	4p'p' log p'
Nb pivots	P–1	P–1	4p' log p' –1
Equilibrage théorique	Proba	<= 2	Proba
Equilibrage mesuré	+,– 15%	+,–0.1%	+,–0.01%
Texe	1	3	2

RÉSUMÉ DE NOS RÉSULTATS

- Hétérogénéité des processeurs
- Méta-schéma (échantillonnage)
  - mémoire / disques
  - codes
  - tests : utilisations multiples

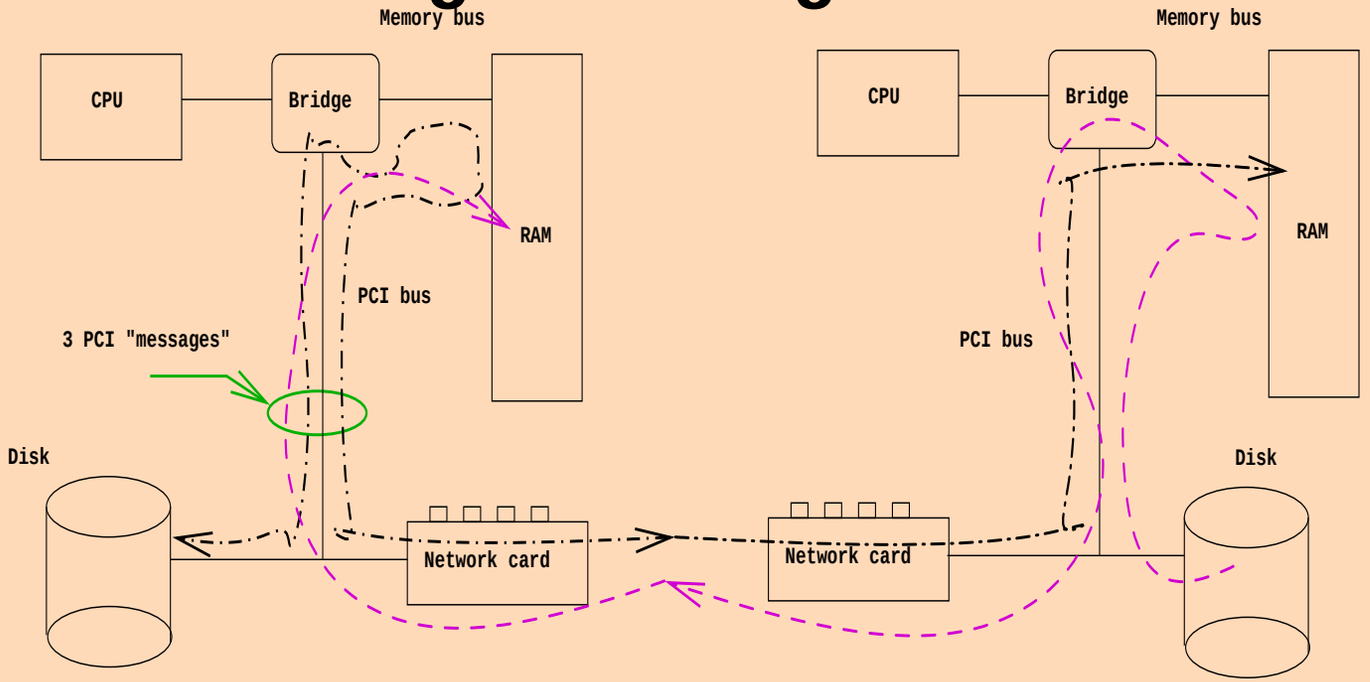
GARANTIE :

- équilibrage des charges
- temps d'exécution

TRIS DISQUES : LE FUTUR

RÉUTILISATIONS IMMÉDIATES

- Bases de données (cf jointure)
- Validation Read^2 (disk to disk efficace)
- Benchmarking hétérogène



NOS TECHNIQUES DE PARTITIONNEMENT

- **PORTÉE GÉNÉRALE DE NOS APPROCHES**
- **PARTITIONNER DES DONNÉES**
  lorsque la machine est hétérogène
- **GÉNÉRALISATIONS HÉTÉ D'ALGOS**
  - **Fouille (regroupement).**

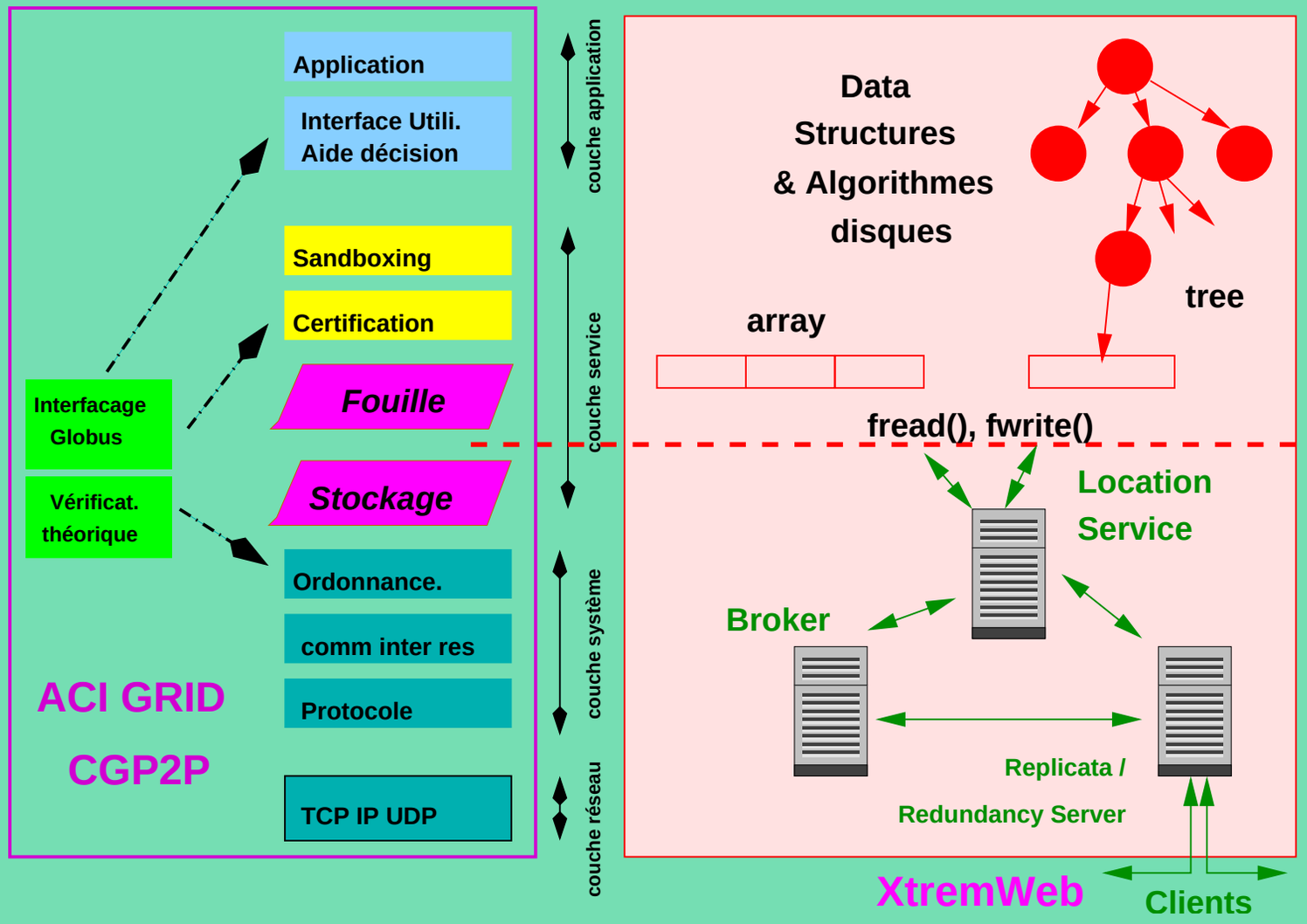
    1- prendre un sous-ensemble de données s
    2- partitionner s en p partitions de taille s/p
    3- dans chaque partition, créer s/pq clusters
  - **Partitionnement de graphes**

I/O PARALLELE
HAUTE PERFORMANCE
SUR CARTE

5- I/O PERFORMANTES

- Motivations stockage disques
  - Vidéo à la demande
  - Bases documentaires, web
  - Applications scientifiques
- Densité des disques ↗ de 60 à 80% par an
  MAIS les temps d'accès ↗ de 10% par an.
- Problématique sous-étudiée

STOCKAGE DANS "GRILLE"

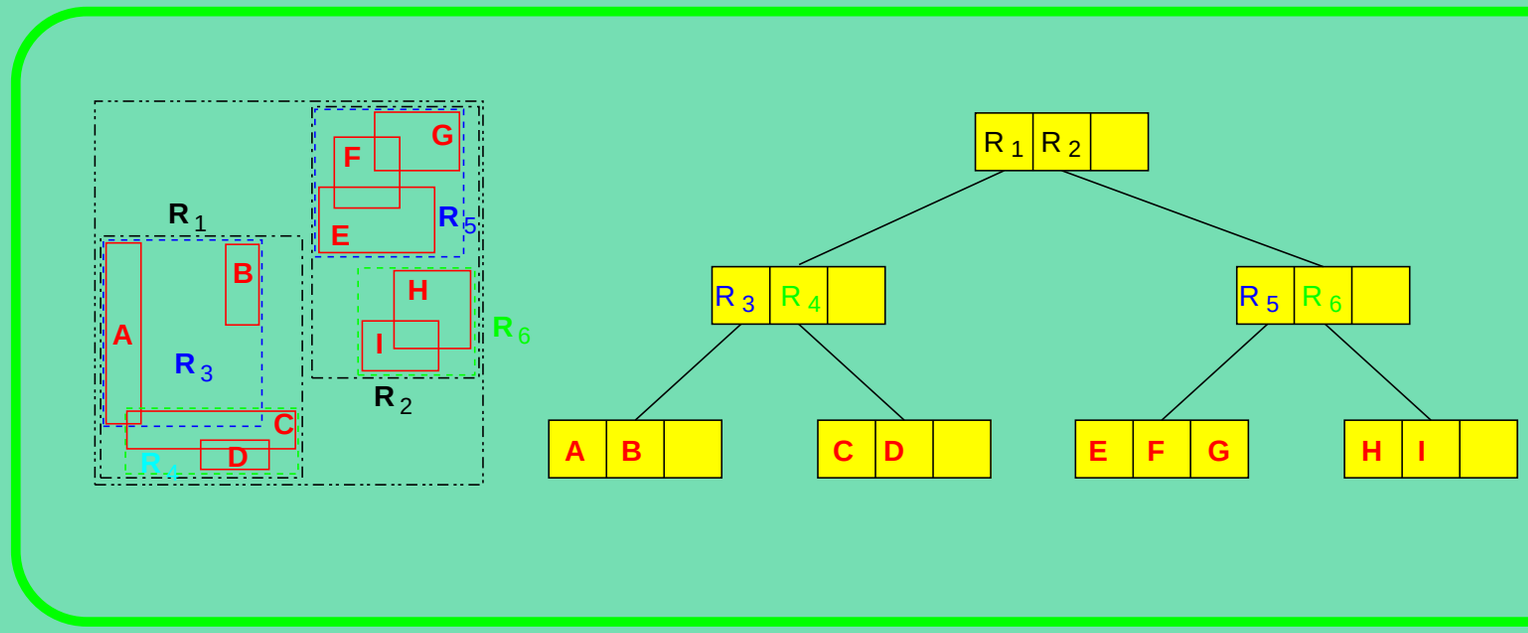


APPLICATION DÉCOUVERTE ÉVÈNEMENTS

- Placement de taches : difficile
- Nécéssité de modèles
- Il existe peu d'heuristiques
  (Pb retour comportement des grilles)
- Trace utilisée
  - ✘ 110 PC de bureau (LAL à Orsay)
  - ✘ 15 jours de mesures
  - ✘ 15 events et infos RARES
- Difficulté monter expériences
- Structures, stockage des traces

PLACER ARBRE 1To SUR LA GRILLE

...composée de 1K machines



Requêtes: Contient,Est Contenu,Est Egal.

FOUILLE, ORDONNANCEMENT/PLACEMENT

- On veut prédire avant de placer
- Fouille : produire des règles qui décrivent les relations entre différentes séquences

↓

Ex: si (BF) arrive 4 fois tandis que - (ABF) arrive 3 fois le compilateur produit la règle :
“si BF arrive alors il y a 75% de chance de voir arriver A”

FOUILLE : DÉTECTION ÉPISODES FRÉQUENTS

- **Explosion combinatoire du nombre d'états**
  Ex: m=k=15 : 71744535000000000000000 séquences !
- **ÉLAGAGE**
  - **seuil (WINEPI, MINEPI)**
  - **intégration contraintes sur les séquences d'entrées (cSPACE, SPIRIT)**

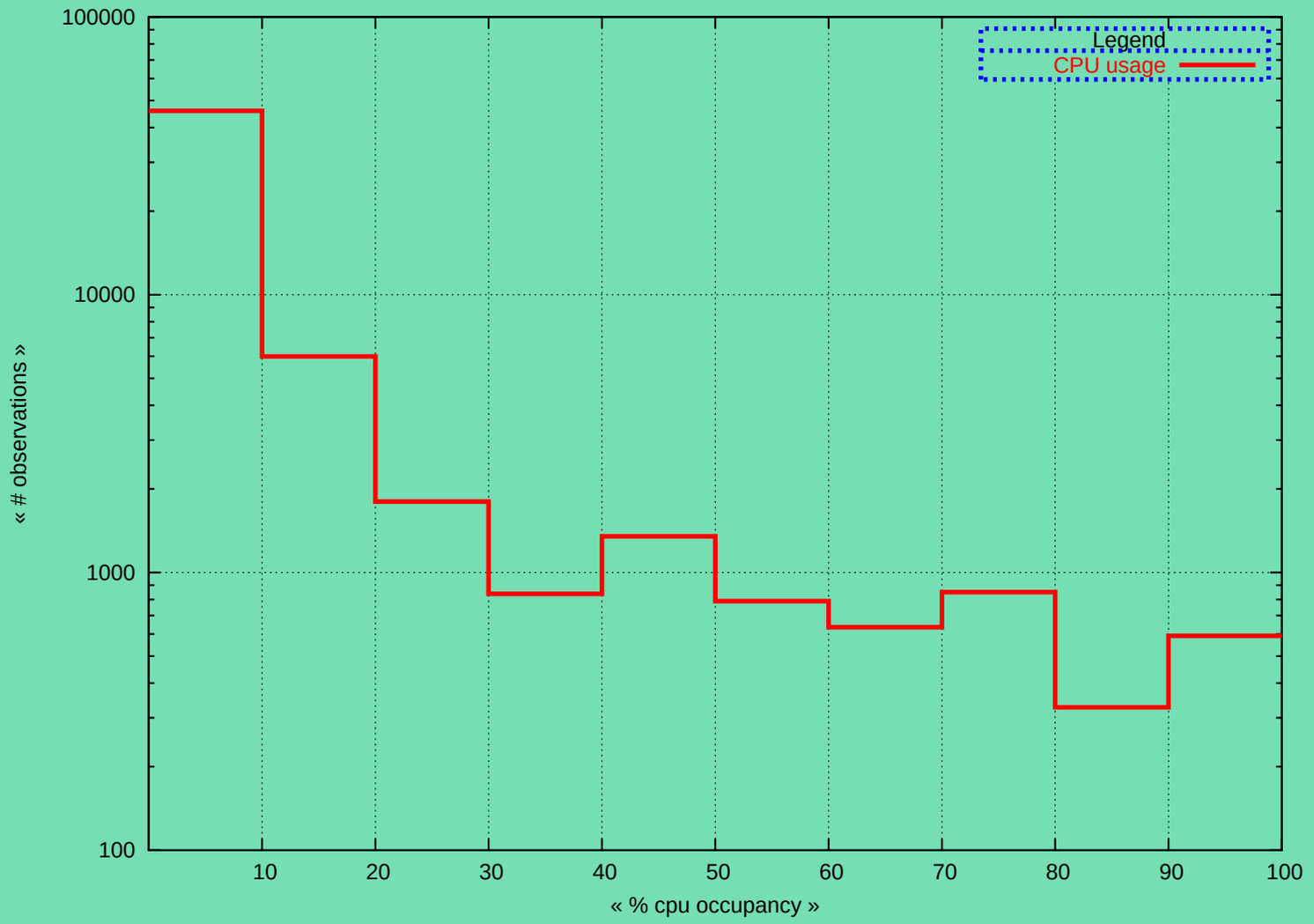
TECHNIQUES DE PATTERN MATCHING

- Versions améliorées de grep ; "approximate string matching" : insert, del, substitution.

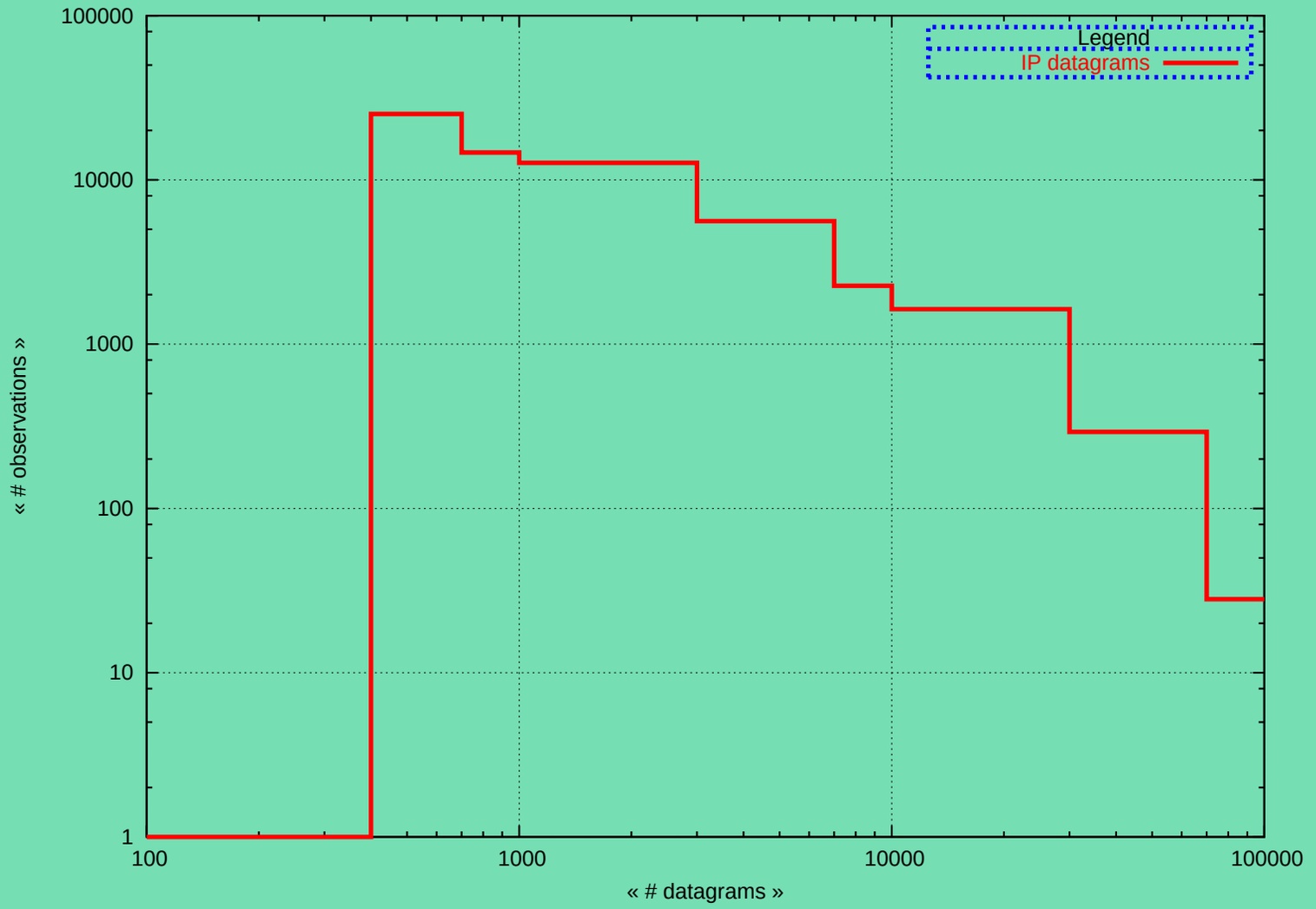
  ↓

  Candidats potentiels pour la fouille.
- nrgrep-1.1.1/ (Gonzalo Navarro at the University of Chile)
- Améliorations envisagées
  - corrections de **Bugs**

EXPLORATION DE LA CHARGE CPU



DÉNOMBREMENT DES DATAGRAMMES IP



CONCLUSIONS I/O HAUTE PERFORMANCE

- Étude statistique disponible
- Les choix des outils de fouille et de représentation des données se précisent
- Défi : fouille multi-dimensionnelle
- Problématique de la fouille est sous-étudiée dans le Grid-Forum

CONCLUSIONS GENERALES
RESULTATS - PERSPECTIVES

RÉSULTATS OBTENUS

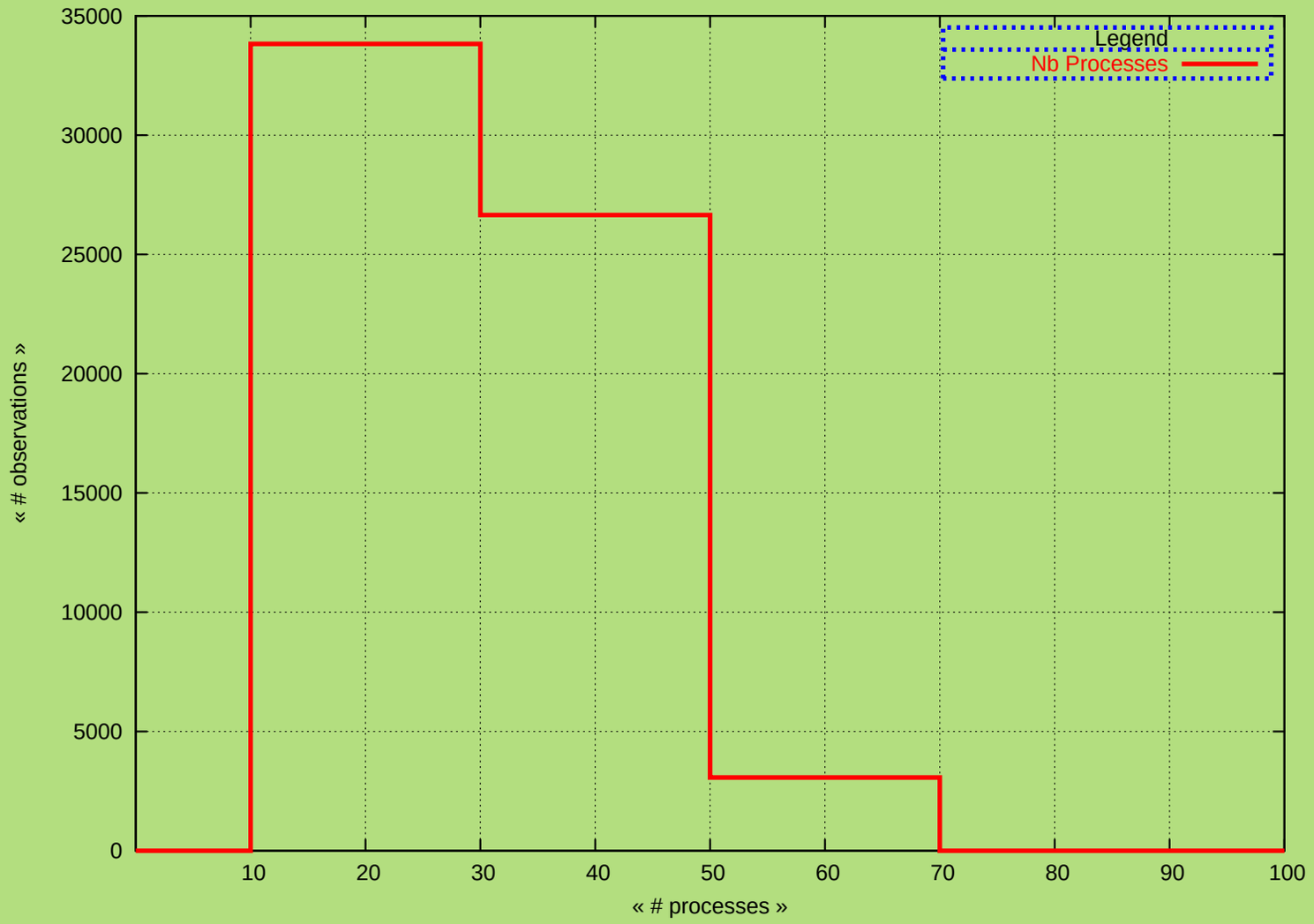
- Architecture
  - Méthode novatrice d'éval perf
  - Cache : nouvel algo de tri (9% - 15%)
- Modèles
  - PRAM, BSR : bornes
  - BSP : comparatif PUB7 / BSPLIB
- TRI SUR CLUSTERS
  - Hétérogène, disque, mémoire: GARANTIES Expérimentations - Benchmarking hété.
  - Méta-schéma de partitionnement en hétérogène réutilisable.

PERSPECTIVES

- Fouille, représentation données, stockage sur grille
  - Sous étudiés
  - Premiers résultats : statistiques
  - Choix structures données
  - Choix des algos. fouille
  - Partitionnement à large échelle
  - Positionnement dans le Grid-Forum

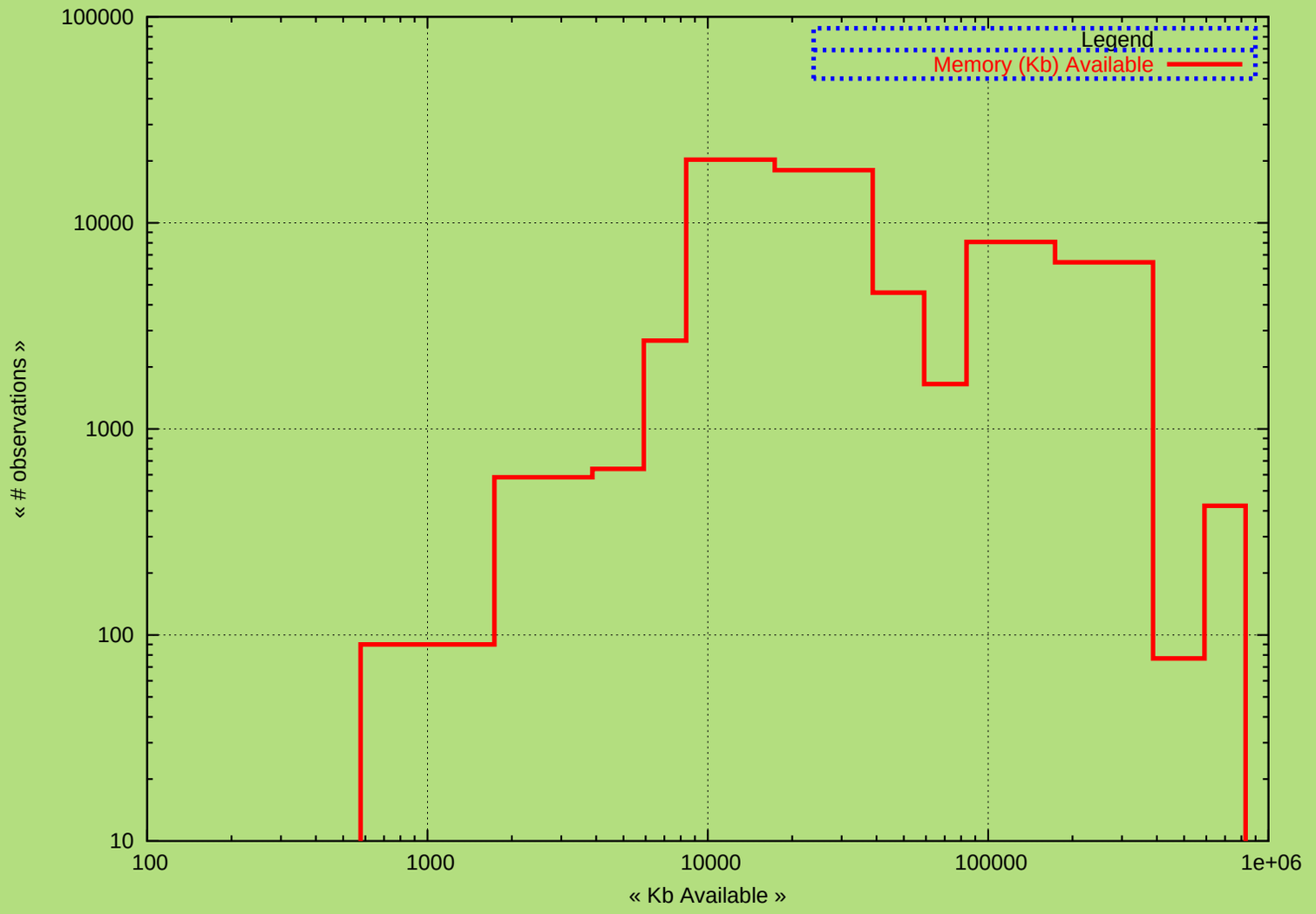
QUESTIONS
RESPONSES

NOMBRE DE PROCESSUS

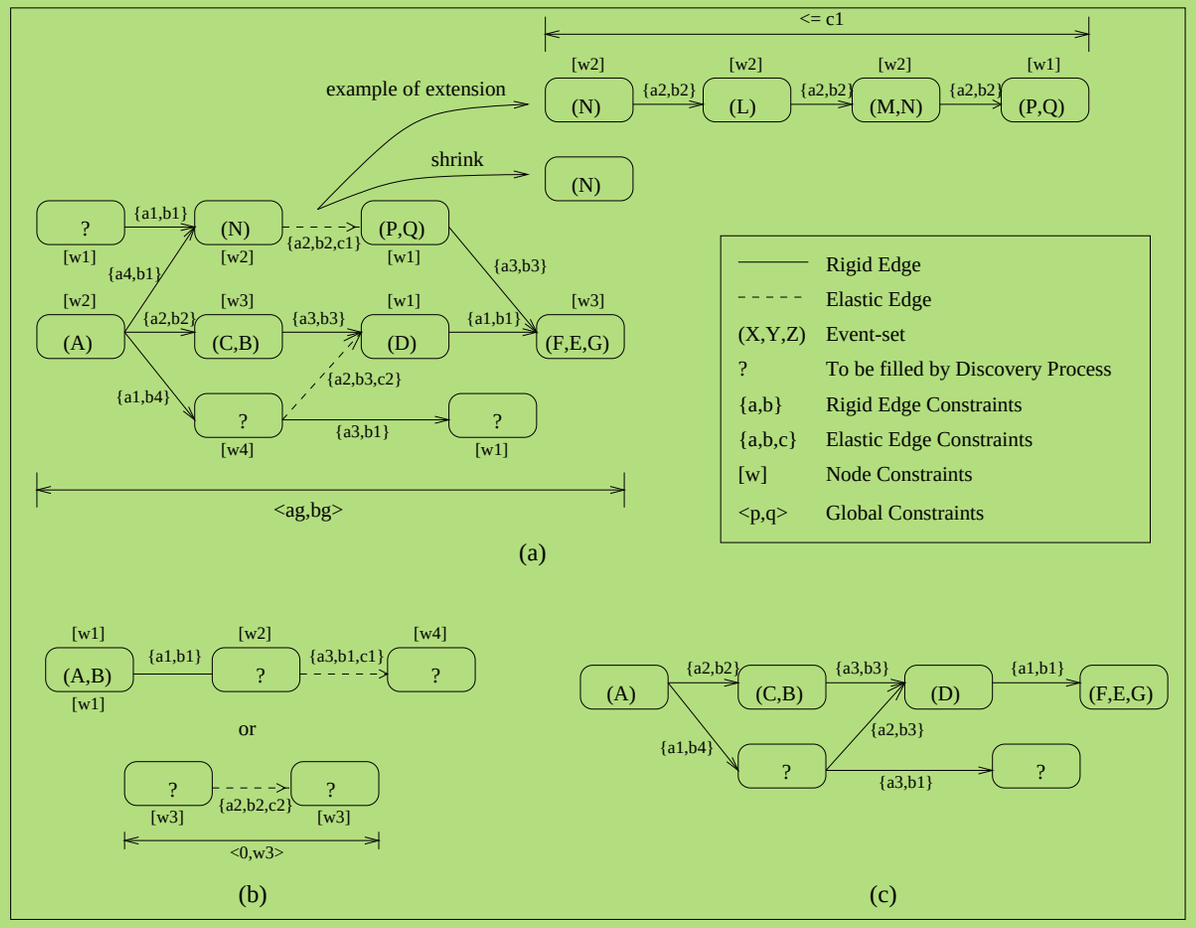


TRANSFERTS DISQUES

MÉMOIRE DISPONIBLE



RELATIONS SÉQUENTIELLES



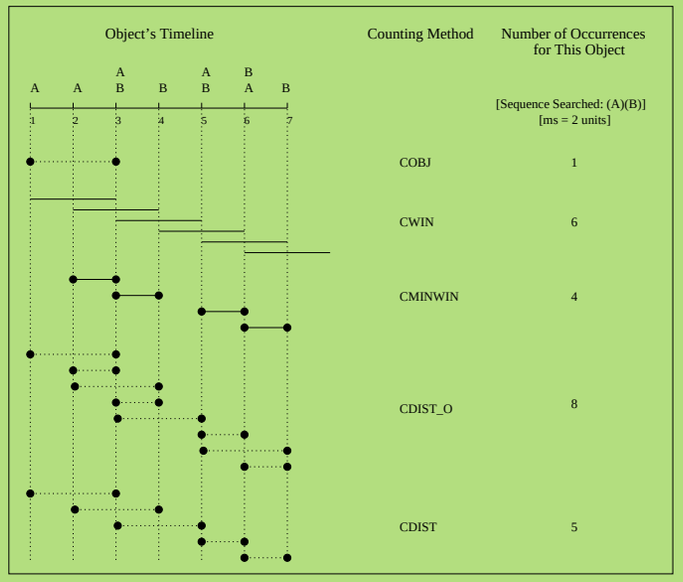
FOUILLE : CRITÈRE DE SÉLECTION

- Une séquence est intéressante si elle apparaît suffisamment de fois ;
- Différentes méthodes de comptage (>8) ;

(a)

	Count All	Count Minimal
Count Windows	CWIN	CMINWIN
Count Occurrences	CDIST_O	CDIST

(b)



FOUILLE : DÉTECTION ÉPISODES FRÉQUENTS

- **Explosion combinatoire du nombre d'états**
  Ex: m=k=15 : 71744535000000000000000 séquences !
- **ÉLAGAGE**
  - **seuil (WINEPI, MINEPI)**
  - **intégration contraintes sur les séquences d'entrées (cSPACE, SPIRIT)**

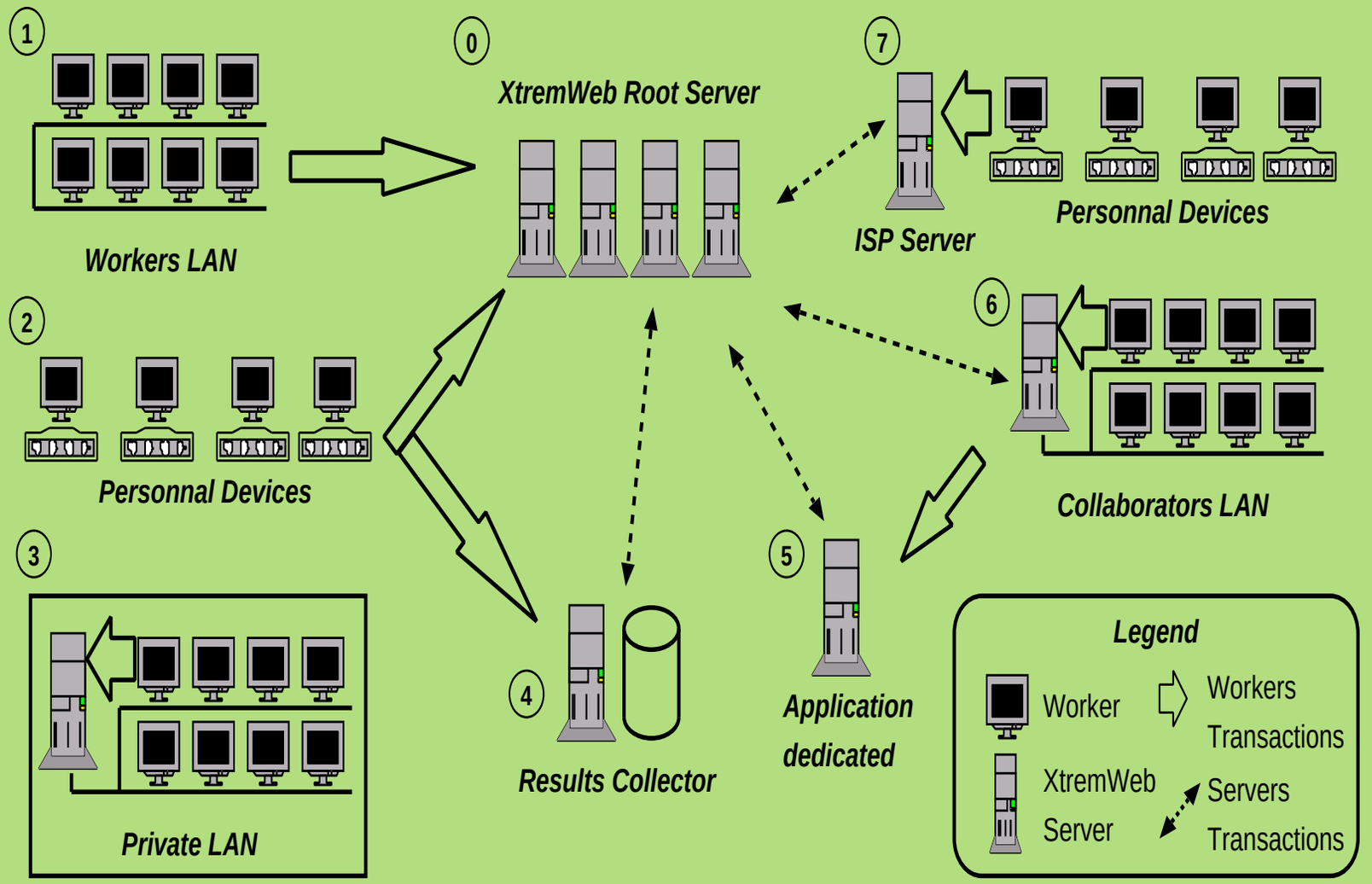
ÉLAGAGE ESPACE DE RECHERCHE

- Règle d'antimonotonie : pour qu'une séquence soit fréquente il faut que toutes les sous-séquences le soient - Utilisation du seuil
- SPIRIT [VLDB 1999] : introduit un langage souple de spécification des contraintes ;
- cSPADE [Zaki, 2001] : reconnu comme le plus performant...mais problème avec le format d'entrée ;

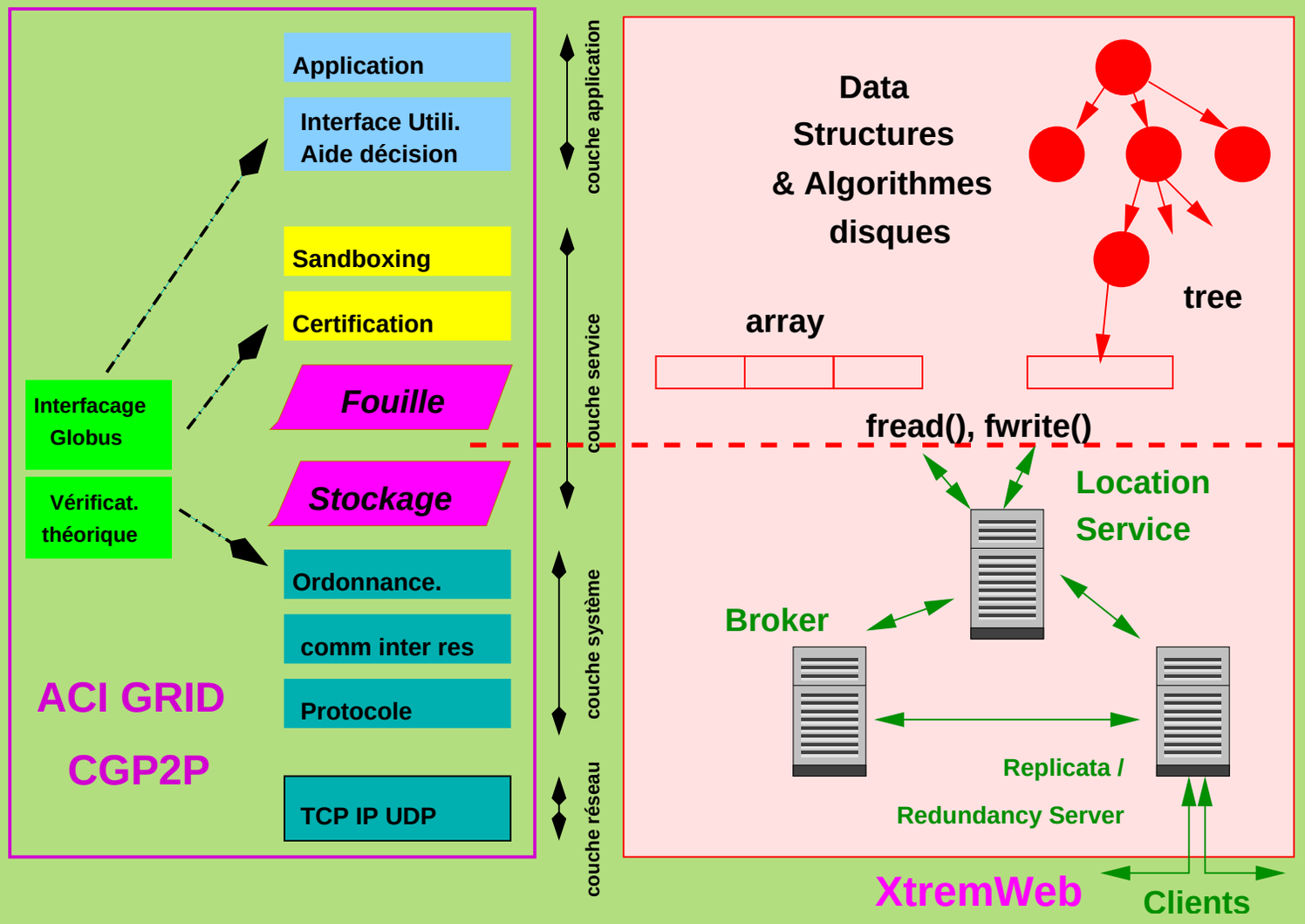
ÉLAGAGE ESPACE DE RECHERCHE

- Exprimer des contraintes au niveau des motifs ;
- SPIRIT [VLDB 1999] : introduit un langage souple de spécification des contraintes ;
- cSPADE [Zaki, 2001] : reconnu comme le plus performant...mais problème avec le format d'entrée ;

INTERNET COMP : XtremWeb

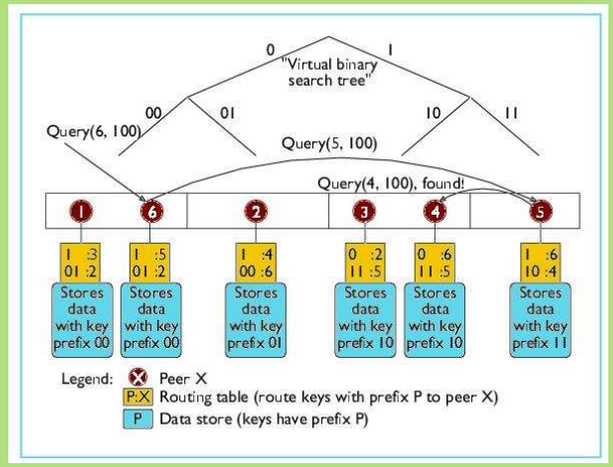


STOCKAGE DANS "GRILLE"

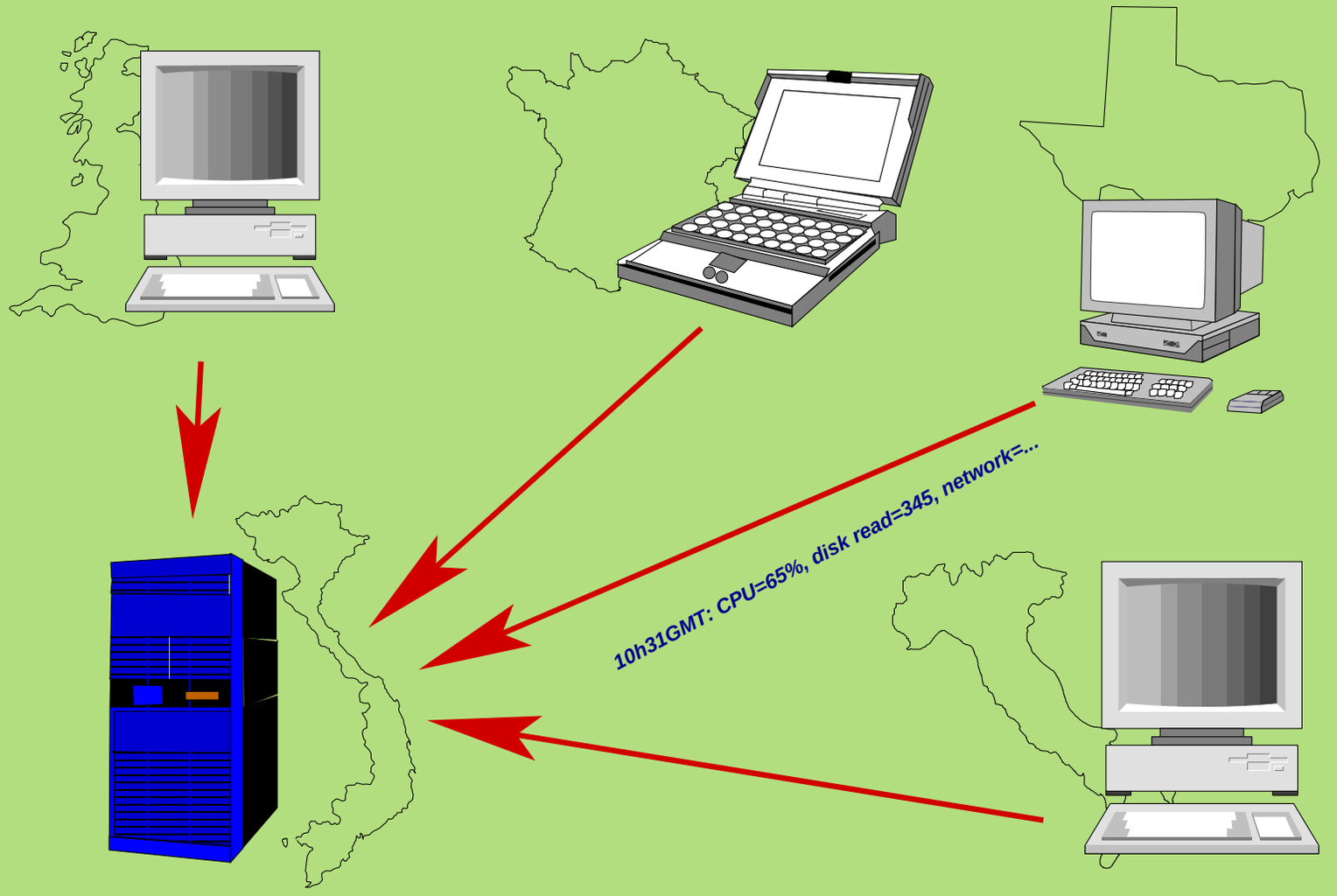


DATA ACCESS: GRIDELLA

- http://www.p-grid.org
- Architecture : pas de contrôle global.



FOUILLE : CONTEXTE ARCHITECTURAL



TEMPS GLOBAL

- **Network Time Protocol (NTP) synchronise le temps d'un client ;**
- **Organisation hiérarchique ;**

```
[root@m24 etc]# crontab -e
5 * * * * /usr/local/bin/ntpdate ntp.obspm.fr
```

  Serveurs secondaire de temps de l'Observatoire de Paris.

- **NTP assure une précision de 10ms ;**
- **http://www.ntp.org/**