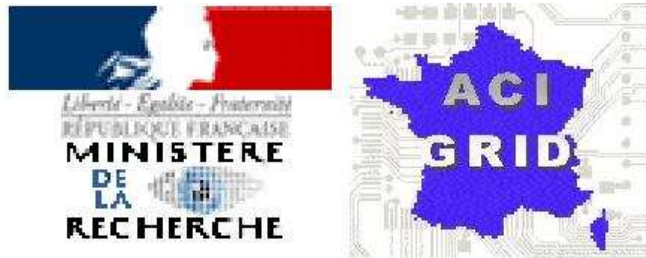


# CiGri

## Aspects infrastructures logicielles



**Action Concertée Incitative [ACI]**  
**Globalisation des Ressources Informatiques**  
**et des Données [GRID]**



Laboratoire  
Informatique et  
Distribution



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



Institut National  
Polytechnique  
de Grenoble



INSTITUT NATIONAL  
DE RECHERCHE EN  
INFORMATIQUE ET  
EN AUTOMATIQUE



UNIVERSITÉ  
JOSEPH FOURIER  
SCIENCES, TECHNOLOGIE, MÉDECINE

# Sommaire

- ▶ Objectif
- ▶ Contexte
- ▶ Approche
- ▶ Points abordés
  - ▶ Intergiciel pour grille
  - ▶ *Méta*-ordonnancement/coopération
  - ▶ Interopérabilité
- ▶ Calendrier
- ▶ Conclusion
- ▶ Première expérience

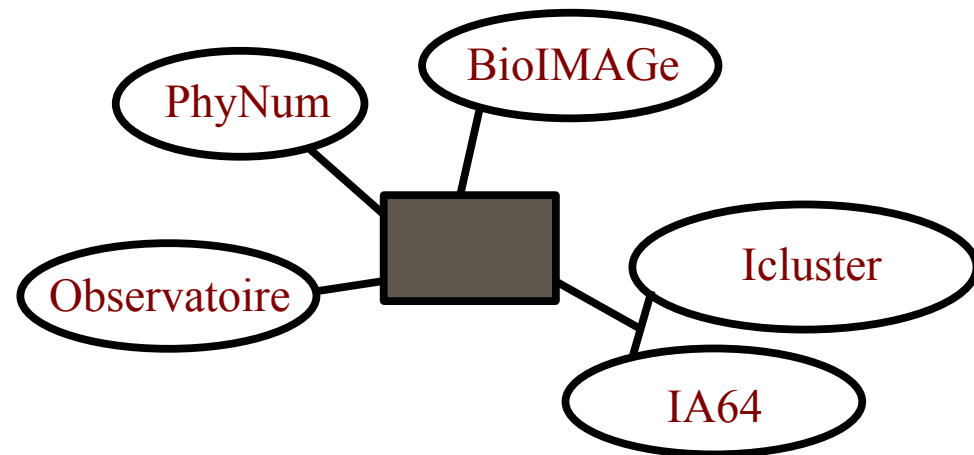
# Objectifs

Une grille opérationnelle pour un domaine applicatif identifié dont les ressources disponibles sont mutualisées

# Contexte

## ▶ **Matériels : Hétérogènes**

- ▶ Grappes des participants à la communauté Ciment
  - BioIMAGE (48 processeur Athlons )
  - PhyNum
  - Icluster ID (200 processeurs PIII 733Mhz, 256 Mo)
  - Grappe200PC (200 processeurs IA64), disponibilité début 2003
  - Autres (MIRAGE, SCIOG, CECIC) ?
- ▶ - **Environ 500 processeurs de puissances et d'architectures différentes**



# Contexte (suite)

- ▶ **Logiciels : Moins hétérogènes**
  - ▶ Système
    - ▶ GNU/Linux(distribution **CLIC**)
    - ▶ Système de gestion de ressources : **PBS**
    - ▶ + systèmes expérimentaux ??
- ▶ Administration : (Ingé. Système)
  - ▶ Locale aux différents sites et peu disponible pour des expérimentations
- ▶ Les utilisateurs *conventionnels* des grappes ne veulent pas être perturbés.

# Contexte (suite)

## ▶ Applications

- ▶ De type Monte-Carlo
- ▶ Nombre de travaux soumis -> grand ?
- ▶ Pas de communications entre tâches
- ▶ Taille mémoire ?
- ▶ Besoin d'entrée/sortie ?
- ▶ Portabilité ?
- ▶ Fixer précisément le *contours* des applications.
- ▶ Un très grand nombre de petit job (10k par soumission)
- ▶ Les utilisateurs valident leurs applications sur les différentes plate-formes

# Notion de Grille légère

## ▶ Constat

- ▶ Difficultés de la mise-en-oeuvre des grandes grilles (Globus/DataGrid/RntI E-toile) (taille de code / nb d'intervenants / maturité du domaine)
- ▶ Les grilles opérationnelles => **2/3 grappes** (simplification des problèmes : d'administrations / d'homogénéités)

# Approche

- ▶ **Restriction de la problématique du calcul sur grille.**
- ▶ Le système doit être
  - ▶ *Simple*
  - ▶ Concis (<20000 lignes de code)
  - ▶ Robuste
  - ▶ Évolutif / Interopérable (modulaire/standard)
- ▶ Contenter les utilisateurs
  - ▶ Bonne tolérance aux pannes
  - ▶ Transparent / simple
  - ▶ Équitable
  - ▶ Pas plus complexe que l'utilisation d'une grappe
- ▶ Contenter les administrateurs
  - ▶ Peu de perturbations
  - ▶ Une nouvelle fonction **l'ingénieur d'administrateur de grille ?**



# Aspects logiciels abordés

- ▶ Système de gestion de ressources pour Grille (intergiciel)
- ▶ *Méta-ordonnancement* (ordonnancement sur grille)
- ▶ Interopérabilité
- ▶ Adaptation des applications
- ▶ Interface utilisateur

## Simplification

- Sécurité (ssh)
- ▶ Système de fichier

---

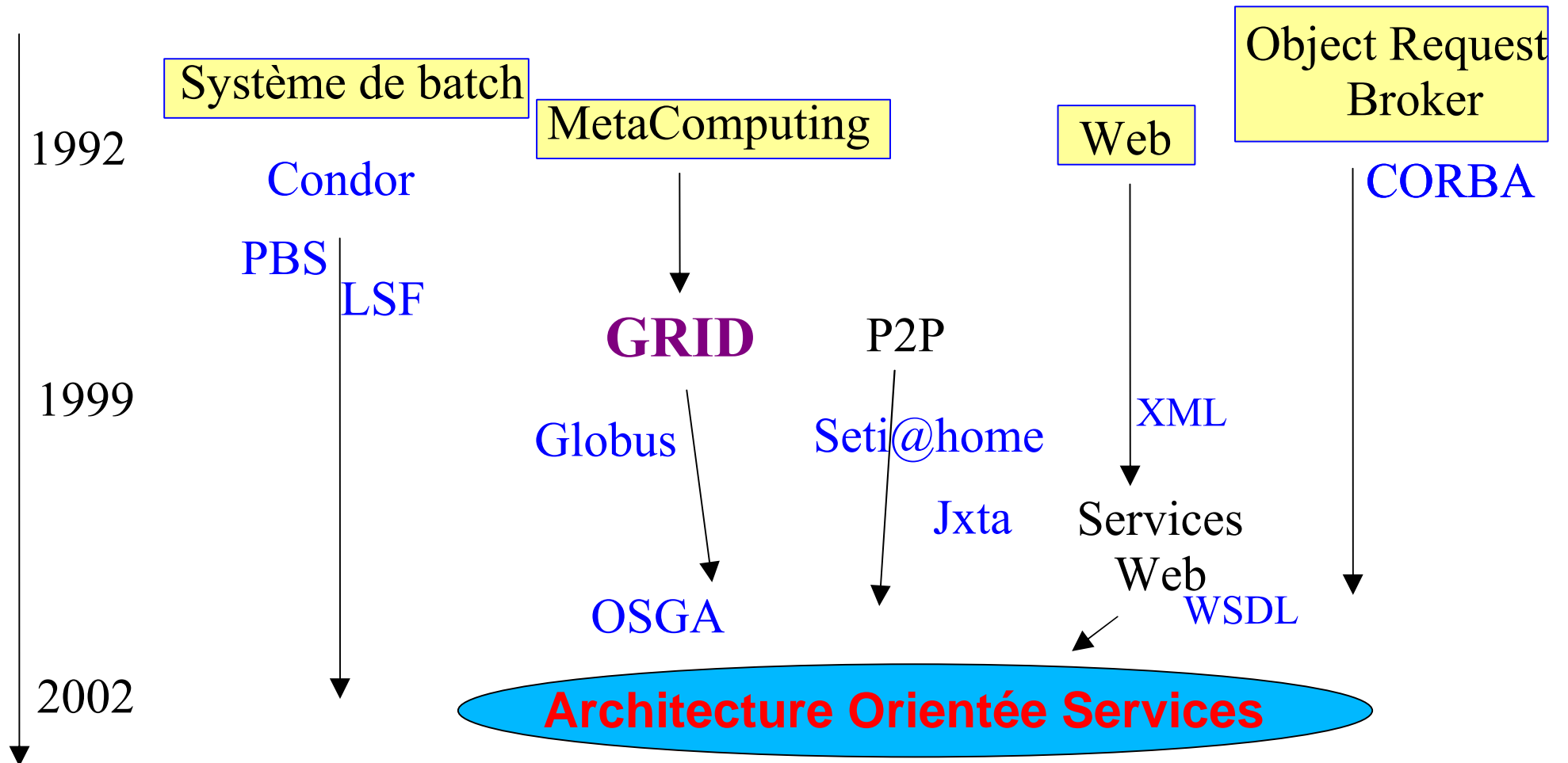
# Systemes de gestion de ressources pour Grille

---

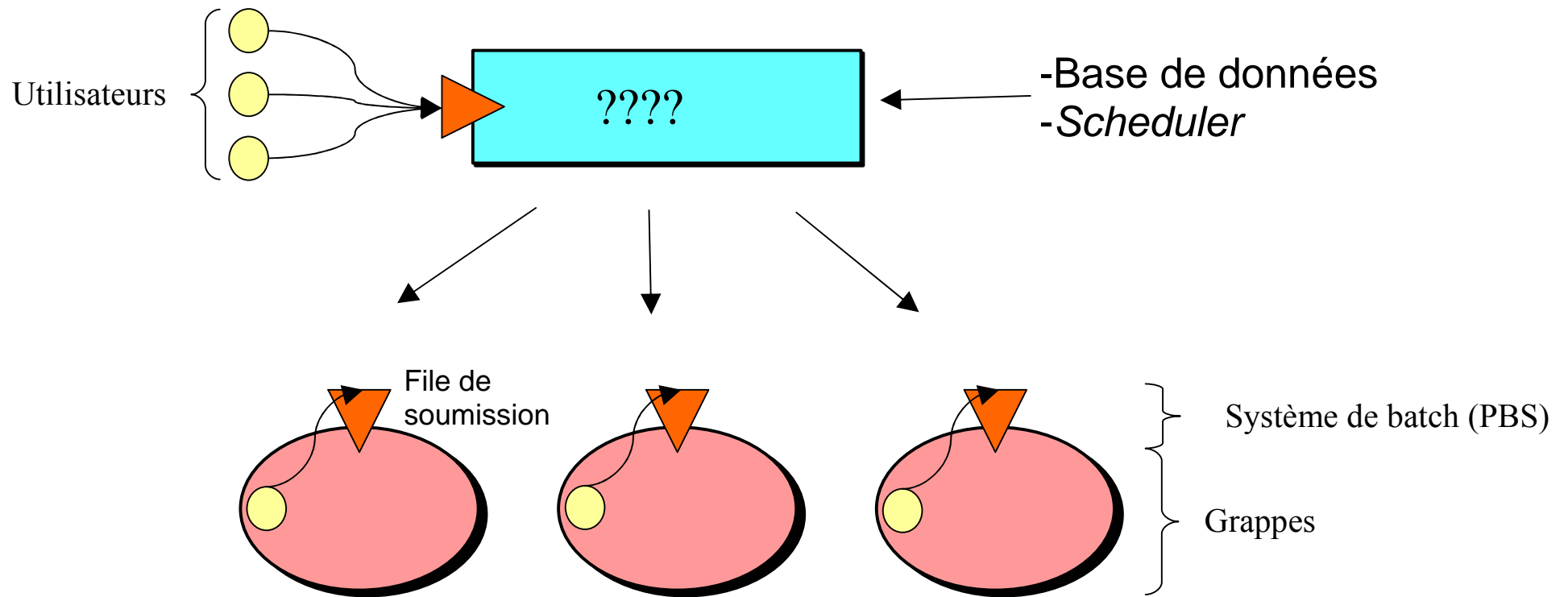
# Les intergiciels pour Grille

- ▶ Pléthore de systèmes >100
- ▶ Pas de **vision unique** de la gestion et de l'exploitation d'une Grille
- ▶ Une petite sélection qui nous intéresse :
  - Globus, Condor-G (MOP), PBS, CCS, Nimrod/G, Unicore
  - Diet (ACI-GRID GASP?)**
  - XtremWeb (ACI-GRID CGP2P)**

# Intergiciels pour Grille (un historique)



# Quelle architecture générale ?

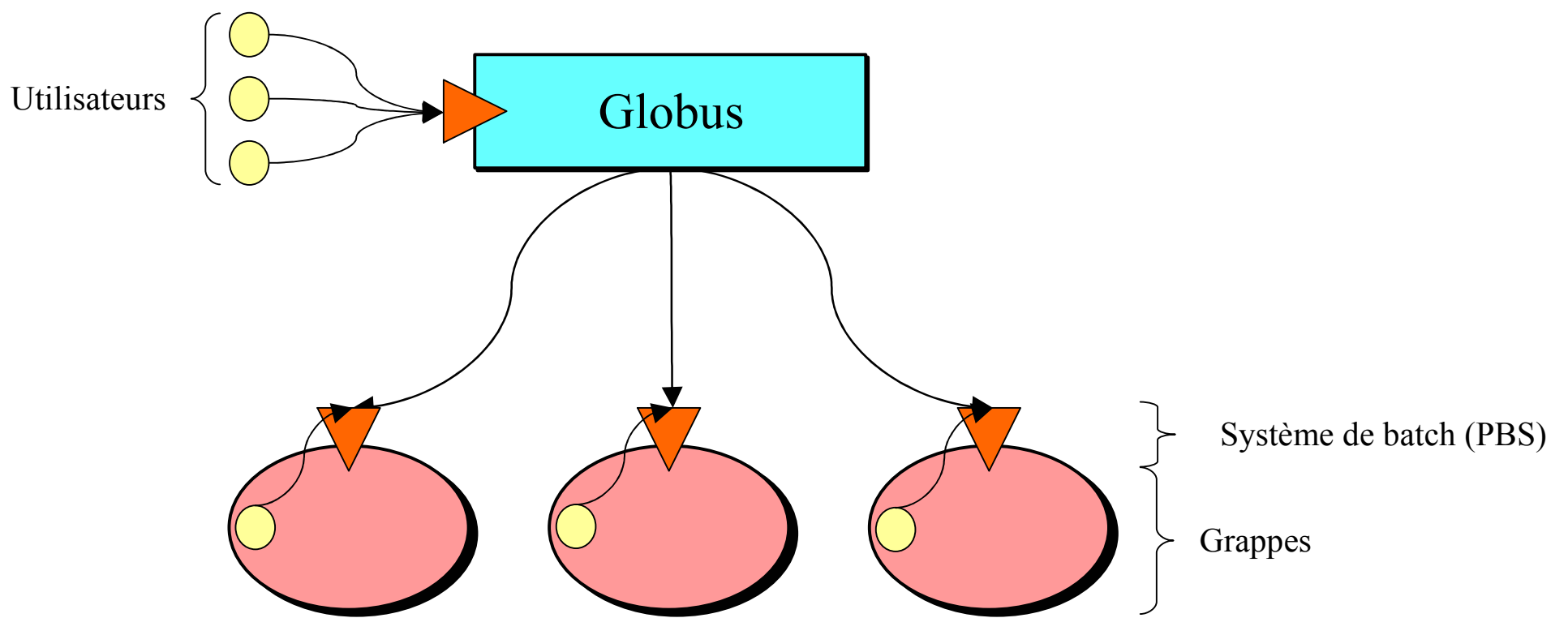


► Quelle organisation fonctionnelle ?

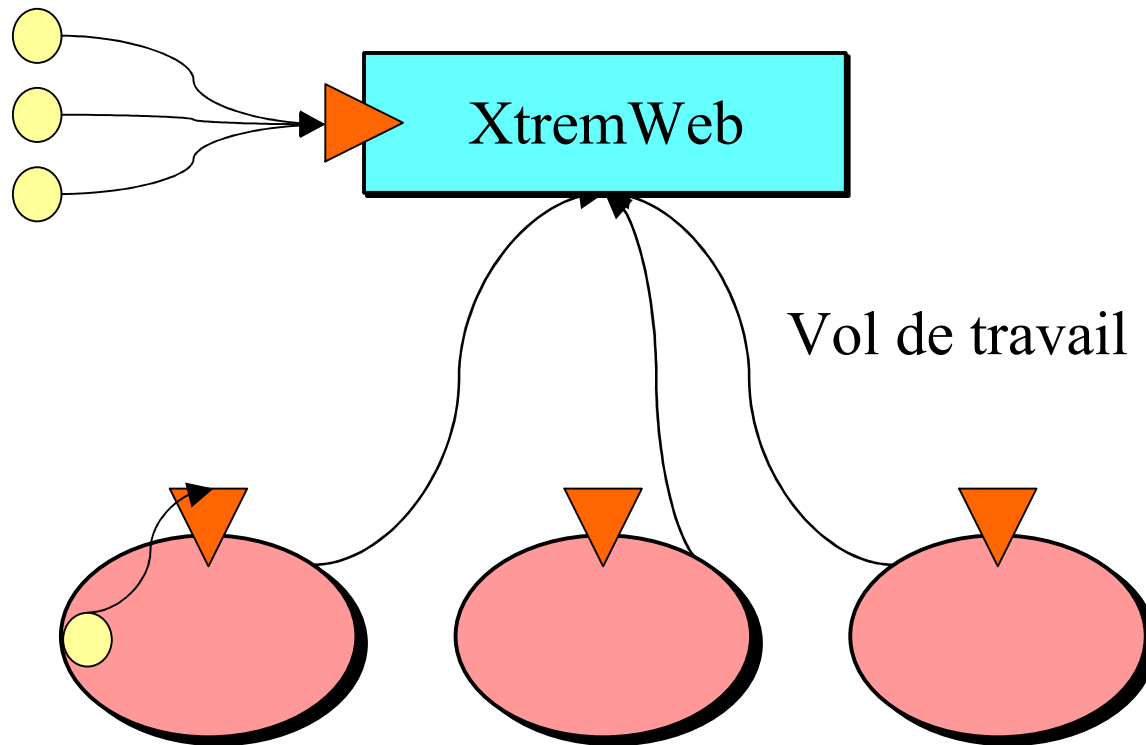
# Une classification

- ▶ Suivant **l'organisation fonctionnelle d'exploitation** des ressources de la grille (comme grappe de grappe)
  - ▶ Centralisé
  - ▶ Centralisé (sur ressources inutilisées / best effort)
  - ▶ Décentralisé (ou agent utilisateur)
  - ▶ Hiérarchique
  - ▶ Coopératif (sur ressources dédiées)
  - ▶ Coopératif (sur ressources inutilisées / best effort)

# Centralisée

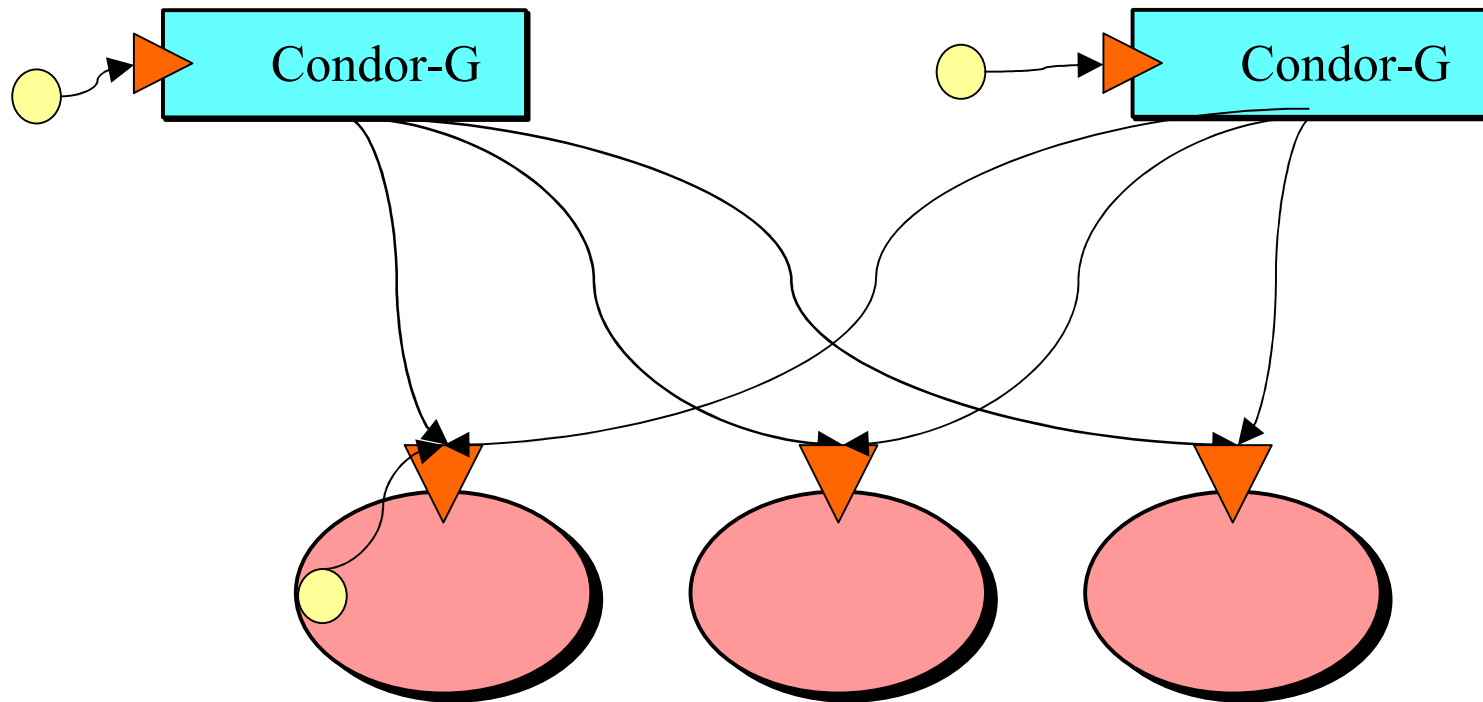


# Centralisée ressources inutilisées (*Global Computing*)

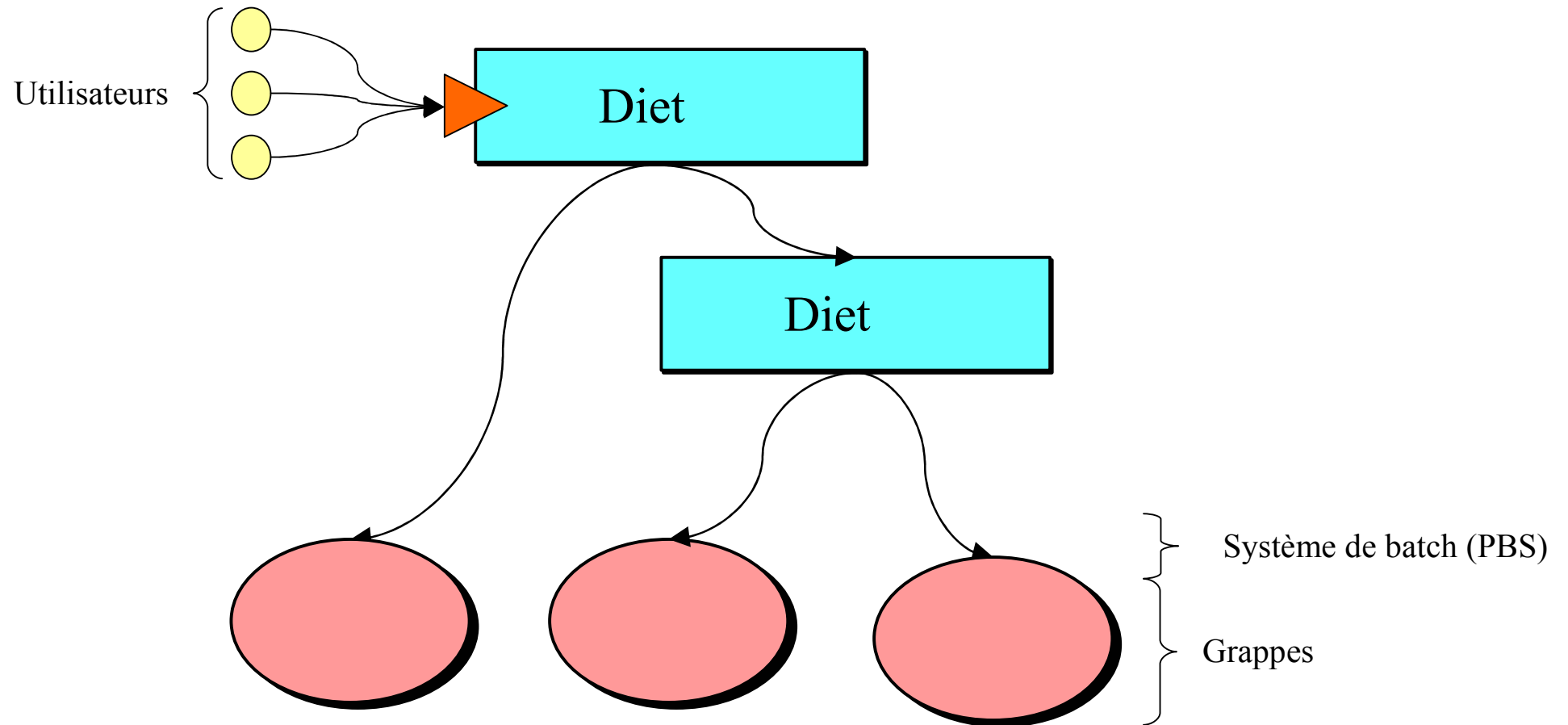




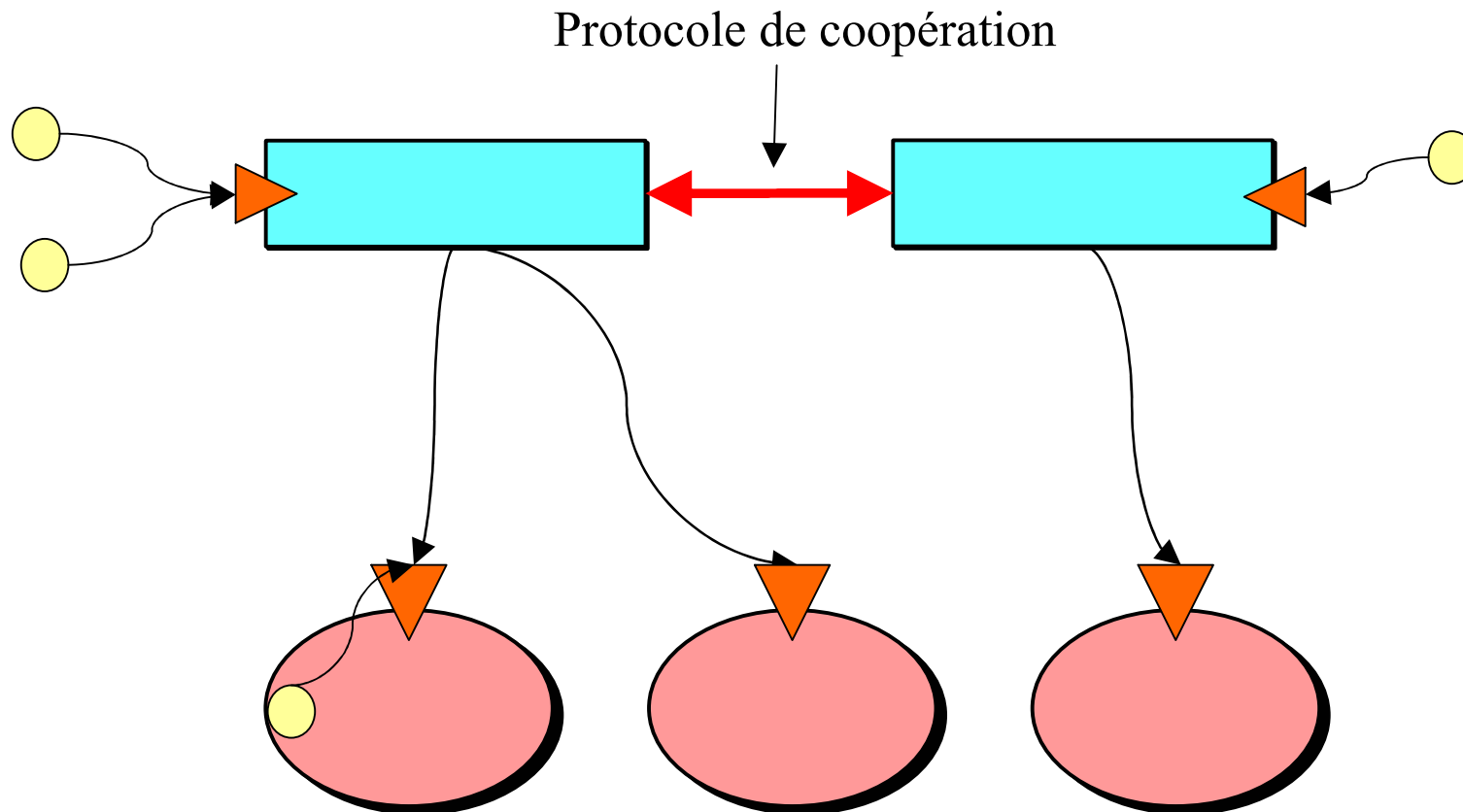
# Décentralisé (agent utilisateur)



# Hiérarchique

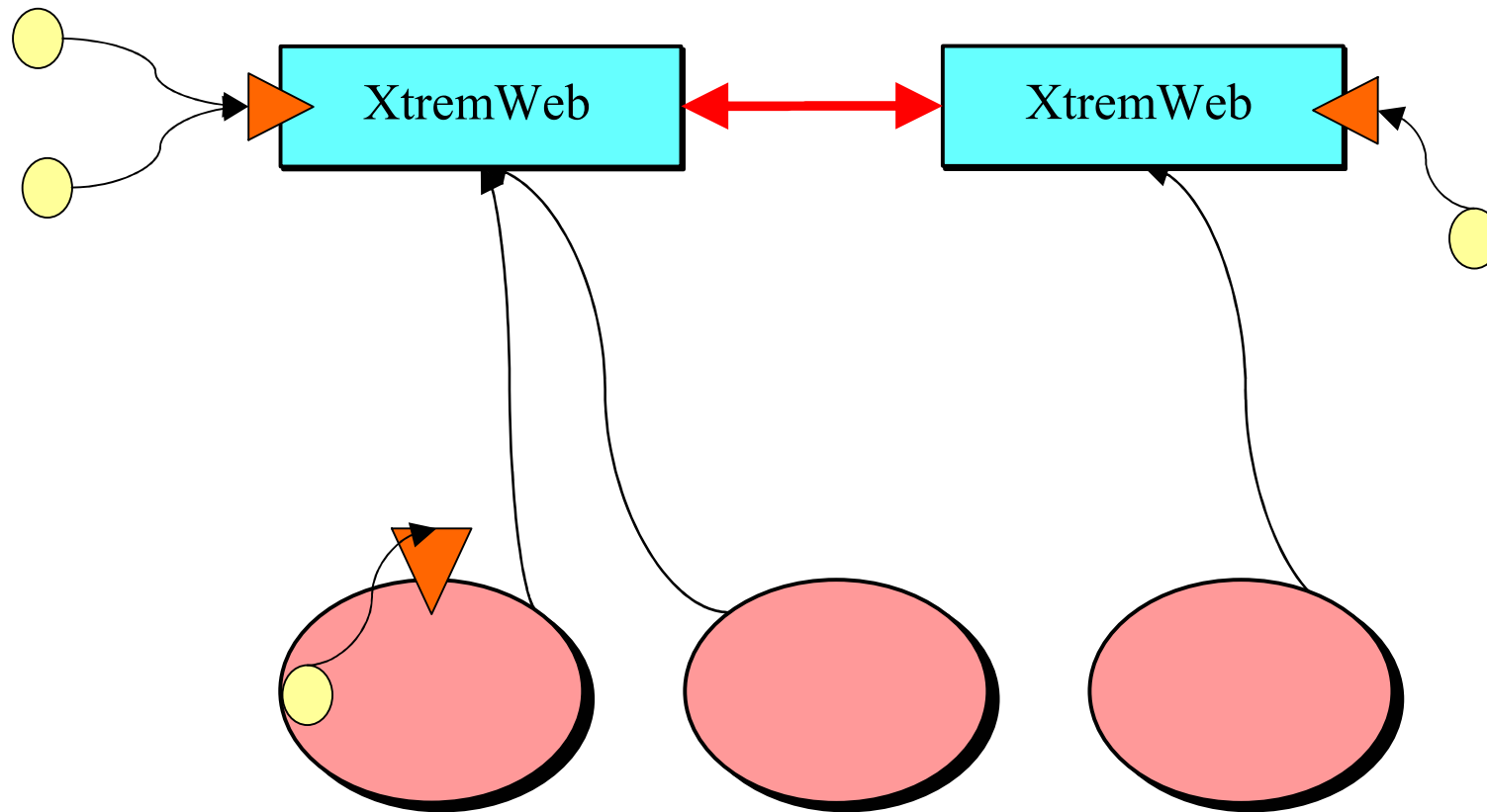


# Coopératif ressources dédiées



- ▶ Correspond à la vision de la Grille *Ciment*

# Coopératif ressources inutilisées



# Synthèse

- ▶ Beaucoup de systèmes
- ▶ Suivi des efforts autour du GridForum et OSGA (définition de standards en cours)
  - ▶ Language de batch (RSL, xRSL)
  - ▶ Protocoles pour la négociations (SNAP)
- ▶ Faire une sélection
  - ▶ Un état de l'art plus poussé est nécessaire

---

*Mé*ta-Ordonnancement et  
Coopération/Négociation

---

# Vision de la gestion de ressources

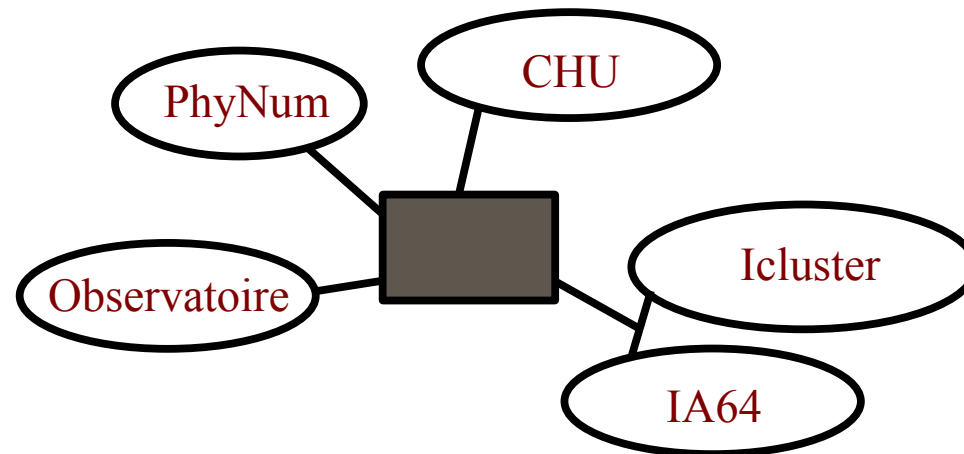
Deux niveaux :

Inter-communautés (site) et sur chaque grappe.

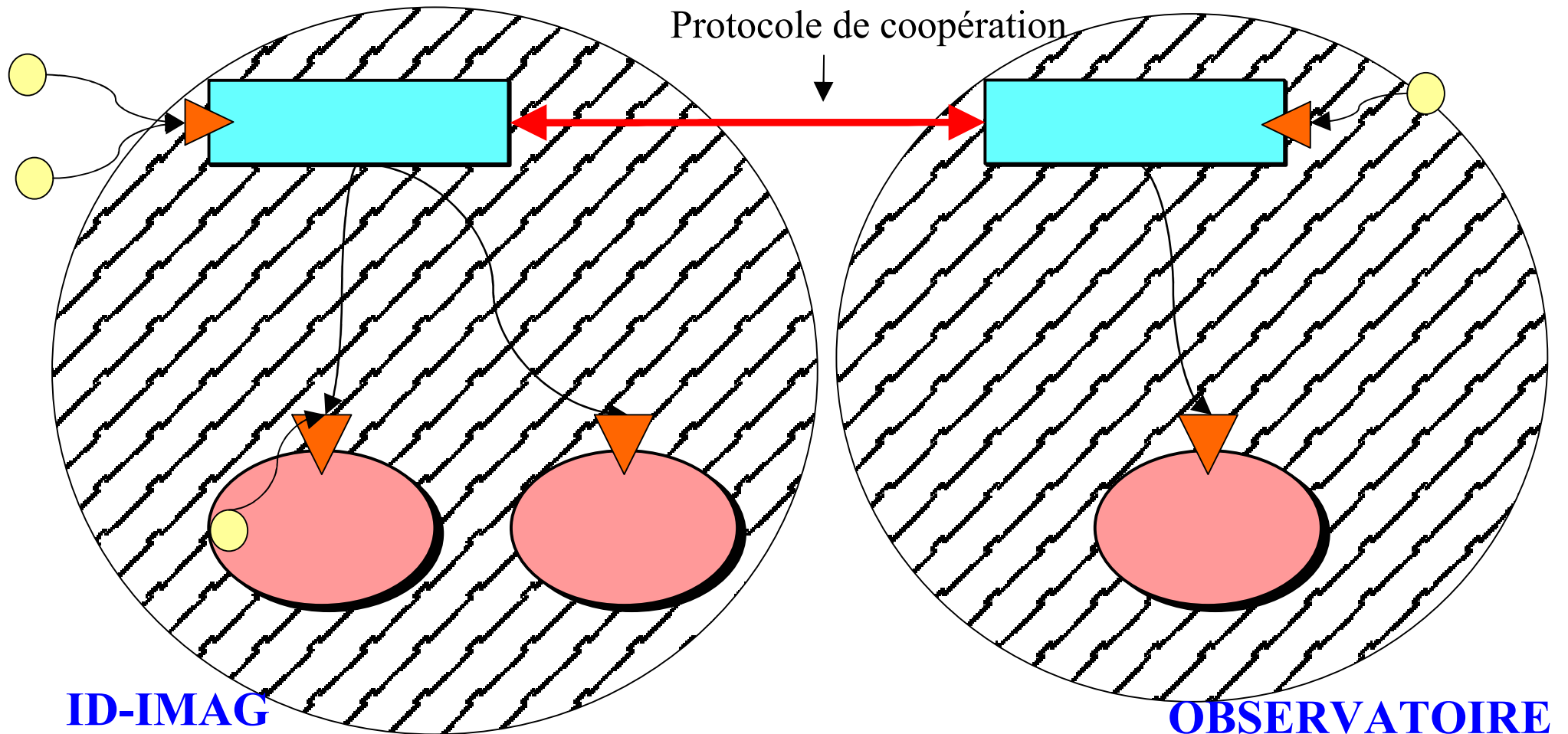
Choix :

Algorithmes de consensus  
(approche économique)

Ordonnancement par lots  
et *tâches malléables*



# Inter communautés



- ▶ Quelles politiques de coopérations / quels critères ?
  - ▶ Équité, garanties, dates limites, débits



---

# Interopérabilité

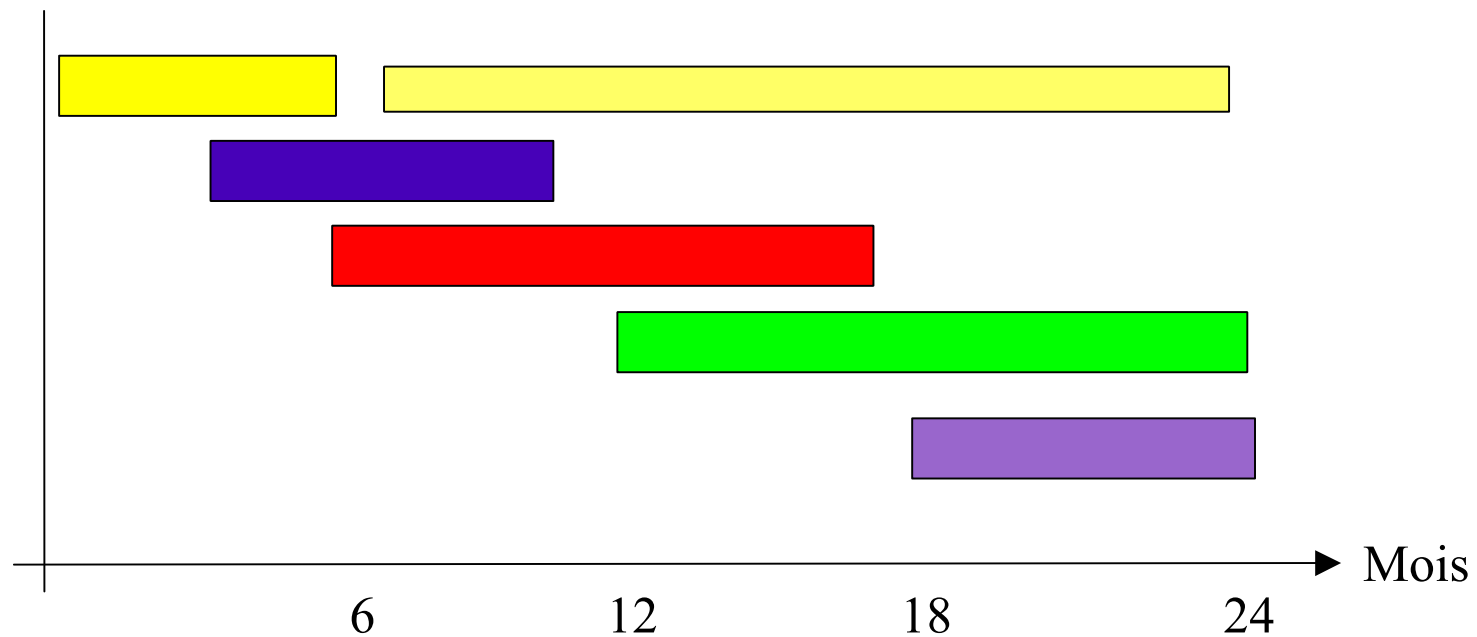
---

# Interopérabilité

- ▶ Objectifs interagir avec plusieurs systèmes
  - ▶ Diet
  - ▶ Xtremweb (module commun / ACI CGP2P ?)
  - ▶ Globus
- ▶ Les développements particuliers devront permettre une interopération aisée
  - ▶ Via RSL, xRSL, SNAP ?
  - ▶ Autre protocole/langage Ordonnancement/Coop.
  - ▶ Développement de wrapper spécifique

# Calendrier

- Etat de l'art + Tests systèmes
- Cahier des charges précis (utilisateur / travail application)+maquette
- Développement(s)
- Tests + Retour utilisateur
- Développements + test interopérabilité



# Conclusion

- ▶ Une **approche qui restreint** la problématique du calculs sur Grille
- ▶ Notion de **Grille Légère**
- ▶ Faire le tri dans les systèmes existants
- ▶ Tests
- ▶ Veille technologique
- ▶ Souhaits des utilisateurs



# Importance de l'interface utilisateur

- ▶ **Condition nécessaire** à l'utilisation du système
- ▶ Acquis -> soumission à la PBS
  - ▶ Ligne de commande
  - ▶ Scripts
- ▶ Particularités
  - ▶ Soumission d'un grand nombre de taches (nombre de paramètres en entrée élevés)  
**Notion de plan d'expérimentation**
- ▶ Accès aux résultats intermédiaires

# Applications (vue du système)

- ▶ **Limitations aux**
  - ▶ Applications de type multi-paramètres
  - ▶ Séquentielles
  - ▶ Indépendantes
  - ▶ Taille mémoire raisonnable ( $\leq 256$  Mo)
- ▶ Questions
  - ▶ Quels sont les tailles des données ?
    - ▶ Entrées et sorties
  - ▶ Quel est le nombre d'instances par soumission ?
  - ▶ Comment est spécifié chaque instance
    - ▶ Ligne de commande, fichiers, programmes,...
- ▶ Quels post-traitements ?