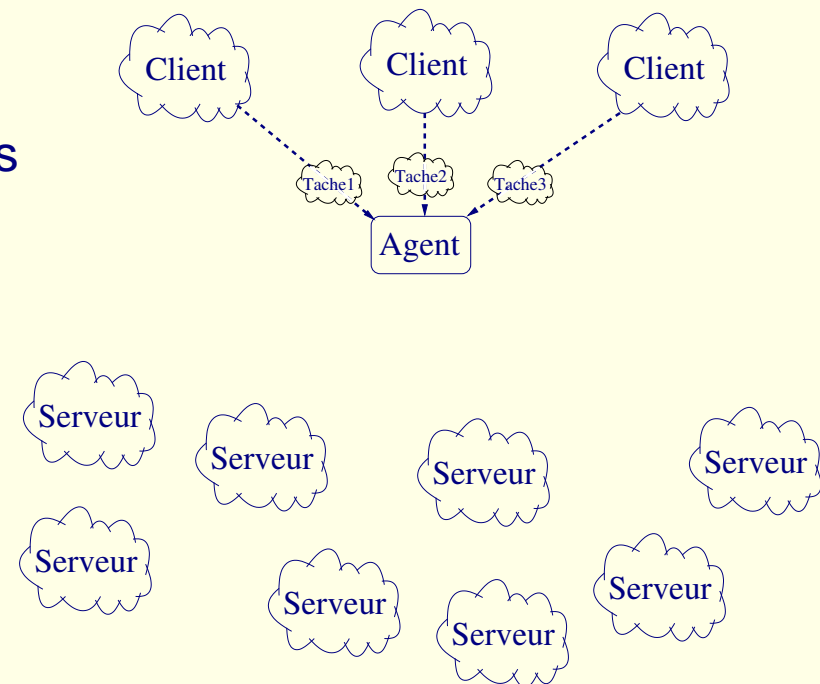


# Résumé graphique de ma thèse

**Objectif** : Fournir les informations nécessaires à l'ordonnanceur  
(à propos des tâches et des serveurs)

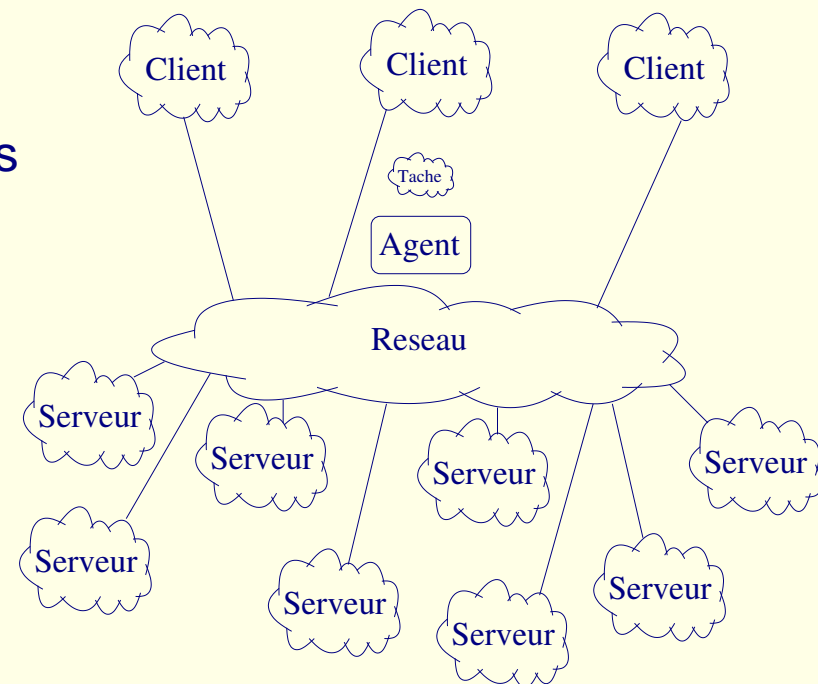
**But de l'ordonnanceur** :  
placer des tâches soumises par les clients  
sur des serveurs de calcul.



# Résumé graphique de ma thèse

**Objectif** : Fournir les informations nécessaires à l'ordonnanceur  
(à propos des tâches, des serveurs et du réseau)

**But de l'ordonnanceur** :  
placer des tâches soumises par les clients  
sur des serveurs de calcul.



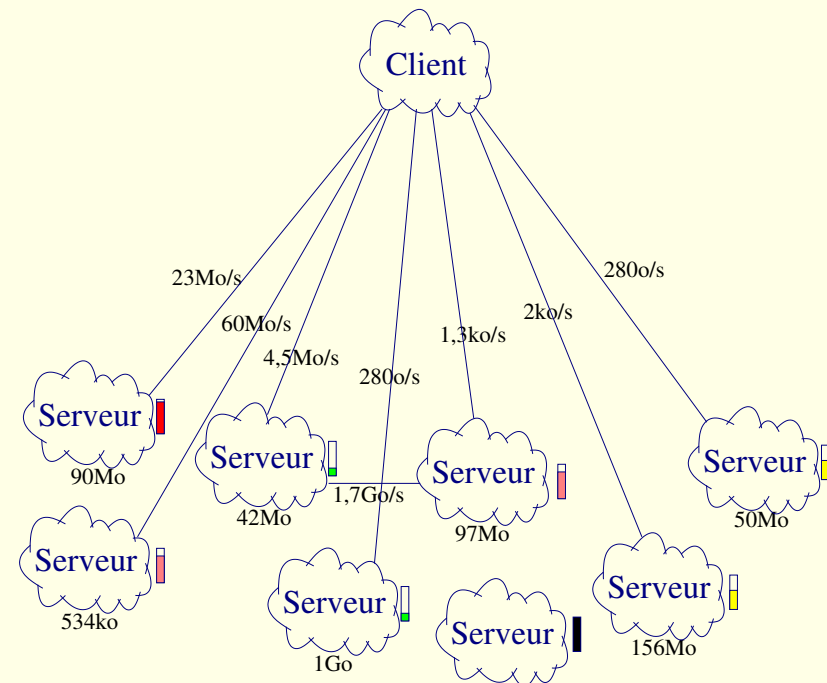
# Résumé graphique de ma thèse

**Objectif** : Fournir les informations nécessaires à l'ordonnanceur  
(à propos des tâches, des serveurs et du réseau)

**Solution** : NWS

**NWS** [Wolski] prédit :

- la bande passante disponible des liens
- taux de remplissage des machines



# Résumé graphique de ma thèse

**Objectif** : Fournir les informations nécessaires à l'ordonnanceur  
(à propos des tâches, des serveurs et du réseau)

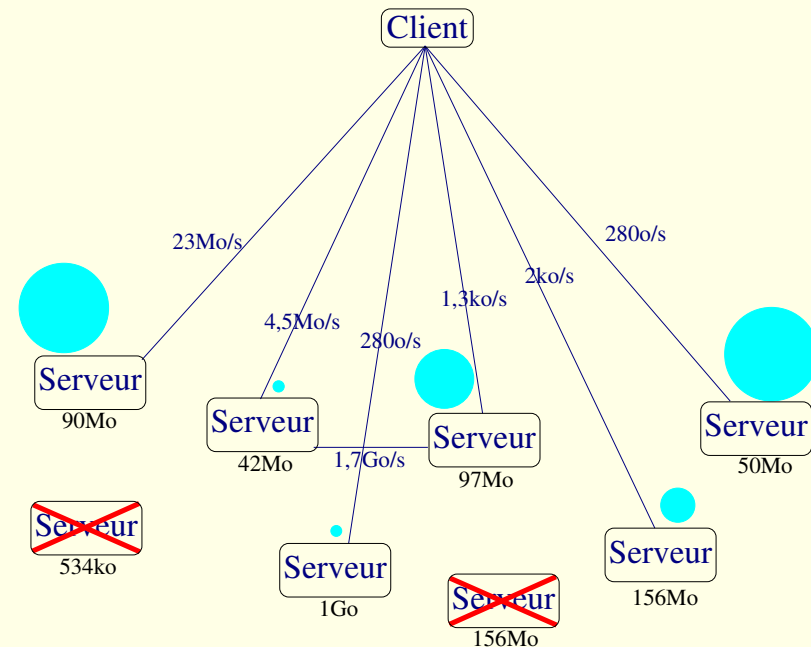
**Solution** : NWS et FAST

**NWS** [Wolski] prédit :

- la bande passante disponible des liens
- taux de remplissage des machines

**FAST** [Quinson] fournit :

- Besoins de la tâche par étalonnage préalable mémoire et temps (adéquation de la machine)
- ⇒ **Durée de la tâche sur chaque machine**
- Simule la réservation (ordonancement avancé)



# Résumé graphique de ma thèse

**Objectif** : Fournir les informations nécessaires à l'ordonnanceur  
(à propos des tâches, des serveurs et du réseau)

**Solution** : NWS et FAST (utilise et améliore NWS)

**NWS** [Wolski] prédit :

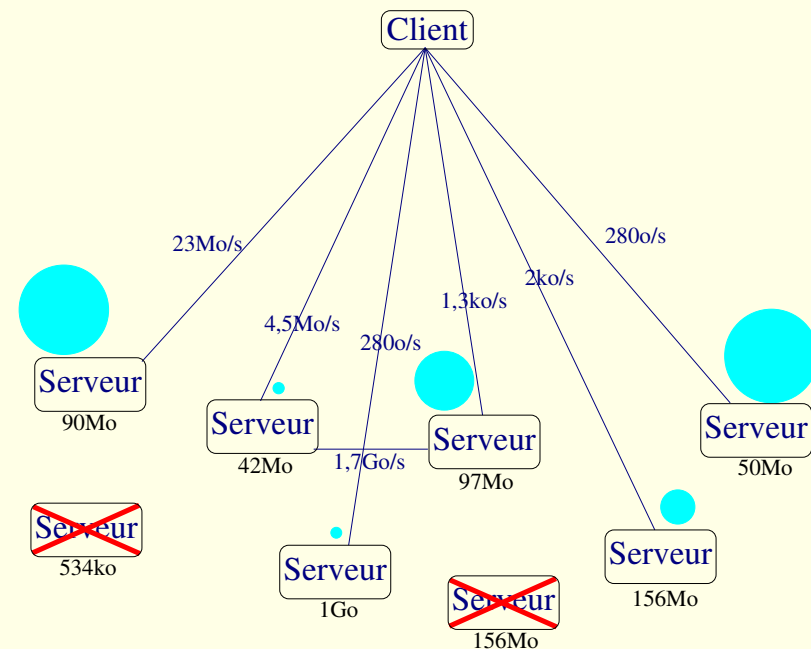
- la bande passante disponible des liens
- taux de remplissage des machines

**FAST** [Quinson] fournit :

- Besoins de la tâche par étalonnage préalable mémoire et temps (adéquation de la machine)
- ⇒ **Durée de la tâche sur chaque machine**
- Simule la réservation (ordonancement avancé)

Améliorations à NWS :

- Réactivité (vis à vis des changements ; pour l'interactivité)
- Combinaison des mesures réseaux



# Résumé graphique de ma thèse

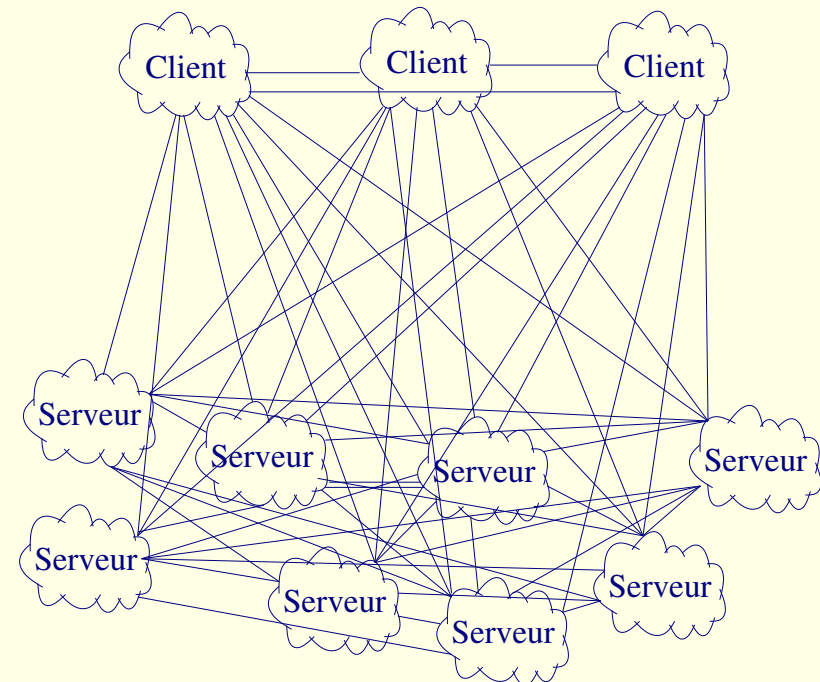
---

**Objectif** : Fournir les informations nécessaires à l'ordonnanceur  
(à propos des tâches, des serveurs et du réseau)

**Solution** : NWS et FAST (utilise et améliore NWS)

**Problème** : Configurer NWS

**Le plus simple** : Mesurer tous les liens



# Résumé graphique de ma thèse

---

**Objectif** : Fournir les informations nécessaires à l'ordonnanceur  
(à propos des tâches, des serveurs et du réseau)

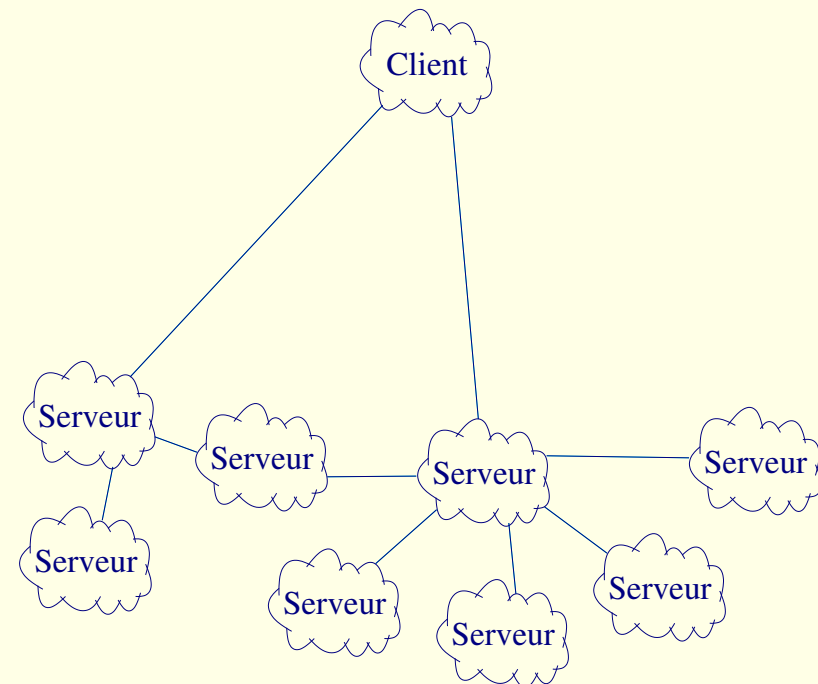
**Solution** : NWS et FAST (utilise et améliore NWS)

**Problème** : Configurer NWS

**Le plus simple** : Mesurer tous les liens

**Préférable** : monitoring étagé

⇒ Connaître la topologie du réseau



# Résumé graphique de ma thèse

**Objectif** : Fournir les informations nécessaires à l'ordonnanceur  
(à propos des tâches, des serveurs et du réseau)

**Solution** : NWS et FAST (utilise et améliore NWS)

**Problème** : Configurer NWS  $\Rightarrow$  connaître la topologie

**Première solution** : Utiliser ENV

**ENV** [Shao] permet de :

- Découvrir la topologie du réseau
- Par mesure  $\Rightarrow$  sans être root

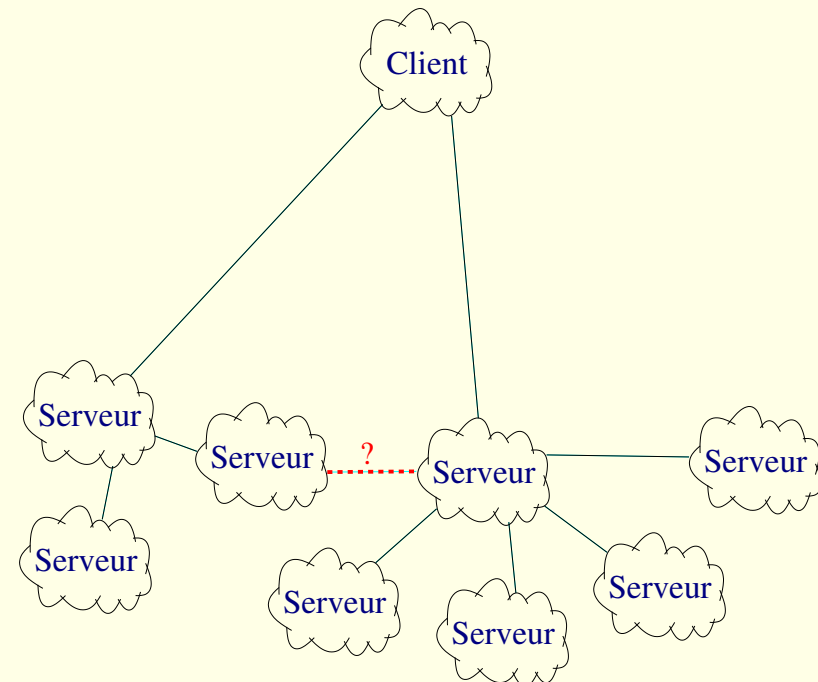
**Ne donne qu'une vue hiérarchique**

$\Rightarrow$  Nécessité d'améliorer l'outil

**Autres défauts d'ENV :**

- Code en Python
- Multiples dépendances

$\Rightarrow$  Expérimentations et évolutions difficiles





# Résumé graphique de ma thèse

**Objectif** : Fournir les informations nécessaires à l'ordonnanceur

(à propos des tâches, des serveurs et du réseau)

**Solution** : NWS et FAST (utilise et améliore NWS)

**Problème** : Configurer NWS  $\Rightarrow$  connaître la topologie

**Notre solution** : Un nouvel outil de communication

**GRAS** [QuiAumLeg] permet :

l'exécution de code sans modification

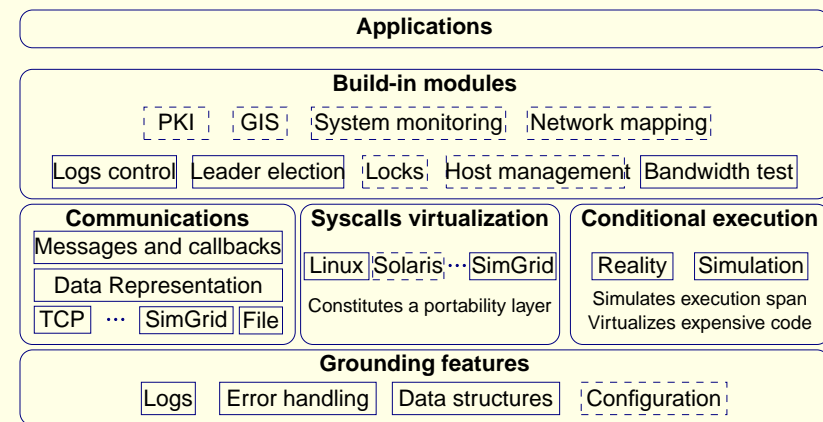
- Sur simulateur (SIMGRID [CasLeg])
- Sur une vraie grille

Applications cibles :

programmes distribués faiblement couplés avec protocole applicatif

Objectifs supplémentaires :

- Efficacité et petite taille
- Simplicité d'usage
- Applications interopérables



# Résumé graphique de ma thèse

**Objectif** : Fournir les informations nécessaires à l'ordonnanceur

(à propos des tâches, des serveurs et du réseau)

**Solution** : NWS et FAST (utilise et améliore NWS)

**Problème** : Configurer NWS  $\Rightarrow$  connaître la topologie

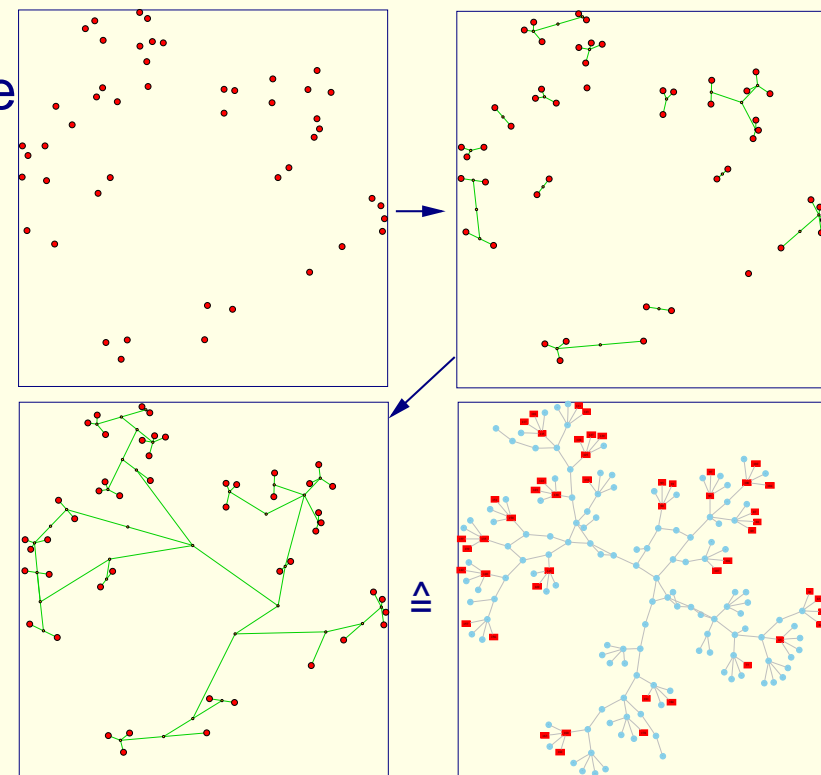
**Notre solution** : ALNeM (utilisant GRAS)

**ALNeM** [LegQui] reconstruit un graphe mimant les interférences mesurées.

$\Rightarrow$  pistes pour un meilleur déploiement

Problèmes d'ALNeM restants :

- Phase de mesures non optimisée
- Traitement des erreurs de mesure
- Campagne de test en grandeur réelle



# Résumé graphique de ma thèse

**Objectif** : Fournir les informations nécessaires à l'ordonnanceur

(à propos des tâches, des serveurs et du réseau)

**Solution** : NWS et FAST (utilise et améliore NWS)

**Problème** : Configurer NWS  $\Rightarrow$  connaître la topologie

**Notre solution** : ALNeM (utilisant GRAS)

**ALNeM** [LegQui] reconstruit un graphe mimant les interférences mesurées.

$\Rightarrow$  pistes pour un meilleur déploiement

Problèmes d'ALNeM restants :

- Phase de mesures non optimisée
- Traitement des erreurs de mesure
- Campagne de test en grandeur réelle

Vers l'auto-configuration de NWS :

- Enrichissement du protocole de verrouillage
- Meilleure intégration d'ALNeM et NWS

