2010/06/24

Déploiement de Glexec et Glite-ARGUS

Pierre Girard

LCG France, Marseille, 2010-06-24

dapnia
cea
saclay
CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE

Content

- Rappel du contexte historique
  - Job pilote multi-utilisateurs
  - Principes de fonctionnement
  - Raisons et Conséquences
- Glexec
  - Principes de fonctionnement
  - Installation au CCIN2P3
- Glite-ARGUS
  - Principes de fonctionnement
  - Installation au CCIN2P3
- Premiers tests
- Conclusions

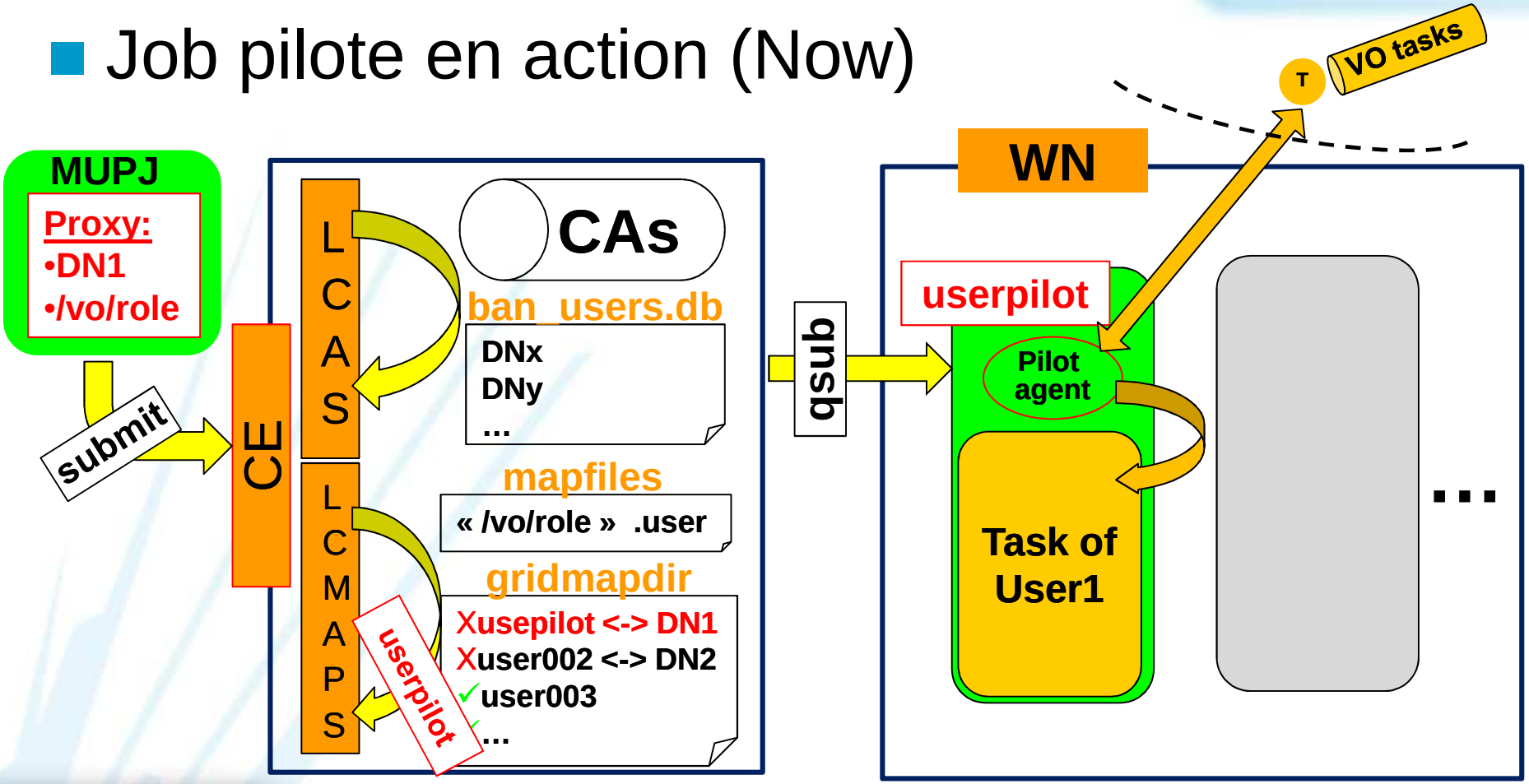
Rappel du contexte historique

Job pilote multi-utilisateurs

- MUPJ: Multi User Pilot Jobs
  - http://litmaath.home.cern.ch/litmaath/Multi_User_Pilot_Jobs.html
  - Definition:
    - On the Worker Node (WN) a pilot job deploys a pilot agent that contacts the task queue managed by the VO's framework, to obtain the highest-priority task (a.k.a. payload) that is compatible with the WN environment.
    - A pilot job can either be "single-user" (a.k.a. "private"), i.e. run only payloads submitted with the same credentials as its own, or "multi-user", in which case it may run payloads submitted by any user authorized by the VO.

Principes de fonctionnement (2)

- Job pilote en action (Now)



Raisons

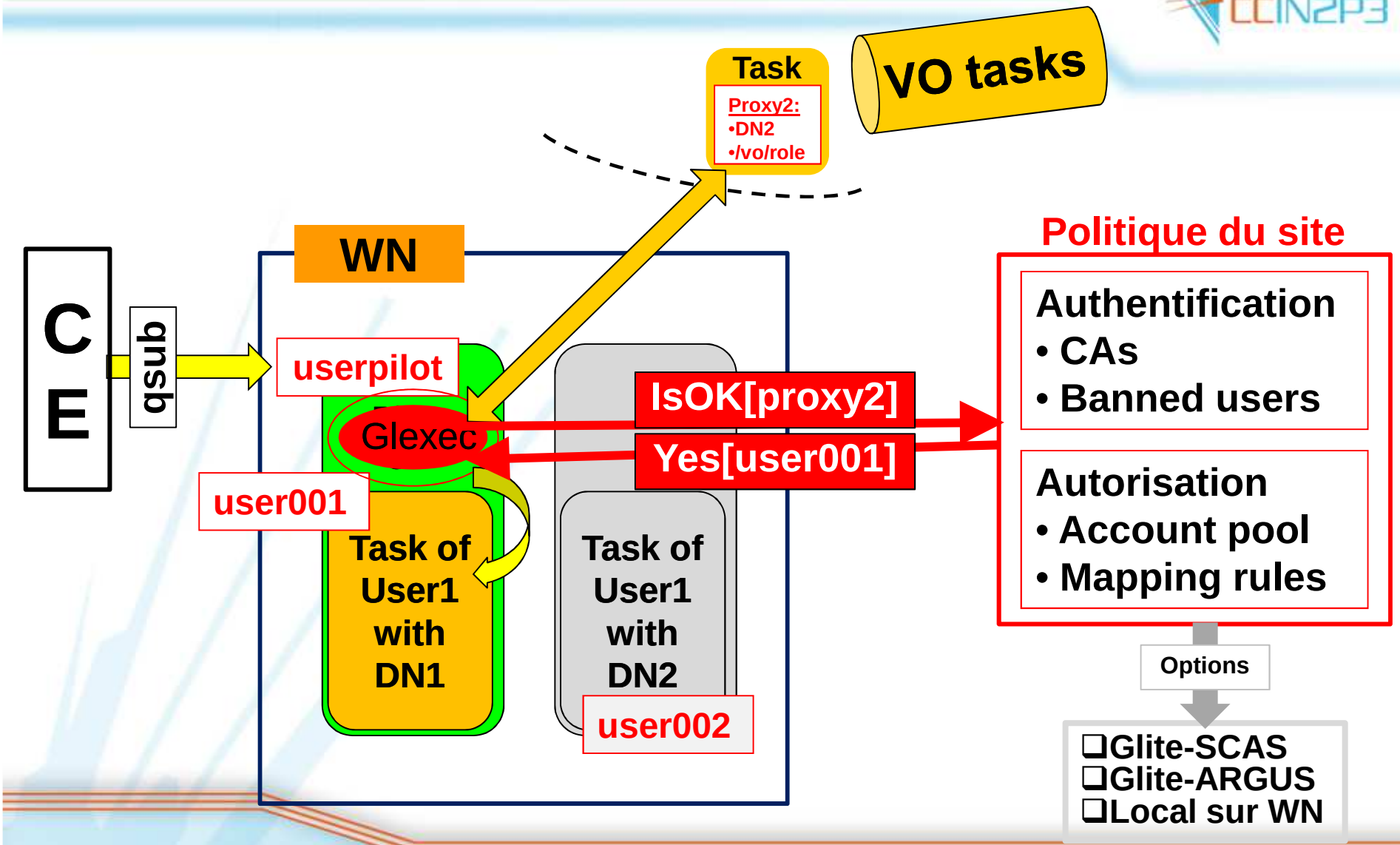
- Pourquoi les VOs sont passées aux jobs pilotes
  - Contourner les problèmes de soumission sur la grille
    - Trop de problèmes avant même d'exécuter la tâche de l'utilisateur
      - Pendant la chaîne de soumission elle-même
      - Une fois sur le Worker Node (problèmes de configuration, mauvaise version du MW, etc.)
  - Gérer « leurs » ressources de calcul
    - Quand elles en ont besoin
      - Grâce à la réservation de « Bonnes » ressources faites par les jobs pilotes
    - Comme elles le souhaitent
      - Sans être subordonnés aux sites pour mettre en place leurs politiques de priorité dans leur propre communauté d'utilisateurs

Conséquences

- Plaintes des sites
  - Crise de confiance
    - La plupart des sites ont découvert l'existence des jobs pilotes à cause de jobs problématiques sur leur cluster
  - Jobs (pilotes) à problèmes
    - Beaucoup de jobs en attente sur les WNs
      - Ressources bloquées inutilement pour les autres VOs
    - Beaucoup de jobs qui sortent juste après leur entrée en machine
  - Problèmes d'accounting
    - long wall time vs. short CPU Time
  - Contournement des politiques locales des sites
    - Impossible de savoir « a priori » les ressources du site que va utiliser un job pilote, donc de contrôler
    - Pas d'Authentication/Autorisation au sein d'un job pilote

Glexec

Principe de fonctionnement(1)



Principe de fonctionnement(2)

- Configuration sur les WNs (setuid/SCAS)
  - /opt/glite/etc/glexec.conf

```
[glexec]
...
user_white_list = dteam049,lhcb049,atlas048,atlas050
...
lcmaps_db_file  = /opt/glite/etc/lcmaps/lcmaps-glexec.db
...
log_destination  = syslog
```

Comptes autorisés à soumettre (ex.: userpilot)

  - /opt/glite/etc/lcmaps/lcmaps-glexec.db

```
...
scasclient = "lcmaps_scas_client.mod"
        " -capath /usr/local/shared/grid/grid-security/certificates"
        " -endpoint https://cclcgvmli18.in2p3.fr:8443"
        " -endpoint https://cclcgvmli19.in2p3.fr:8443"
        " -resourcetype wn"
        " -actiontype execute-now"
```

Services de gestion de la politique d'autorisation du site

Installation au CCIN2P3 (1)

- 1ière tentative: site candidat (may/july 2009)
  - Objectif
    - Tester en charge le service Glite-SCAS
  - Mode d'installation glexec:
    - setuid avec service Glite-SCAS en « backend »

  ⇒Problème d'installation de glexec au CC-IN2P3

  ⇒Incompatibilité avec notre méthode d'installation de Glite-WN

  ⇒Partagée sur AFS

  ⇒Configuration de glexec nécessaire sur chaque WN

  ⇒Rend impossible la gestion de plusieurs version de glite-WN

  ⇒Environnement grille « perdu » par Glexec

  ⇒Décision du projet de livrer Glexec comme un service à part

Installation au CCIN2P3 (2)

- 2ième tentative: en production pour OPS (mai 2010)
  - Objectif: Déploiement sur tous les T1 WLCG
    - Plan de déploiement proposé par WLCG Technical Forum
    - Accord de tous les sites LCG-France (janvier 2010)
  - Mode d'installation glexec
    - setuid avec service Glite-ARGUS en « backend »

  ⇒Quelques bugs mais compatible avec notre installation

  ⇒Accessible via les CEs depuis début juin

  ⚠Chemin d'accès: $GLEXEC_LOCATION/sbin/glexec

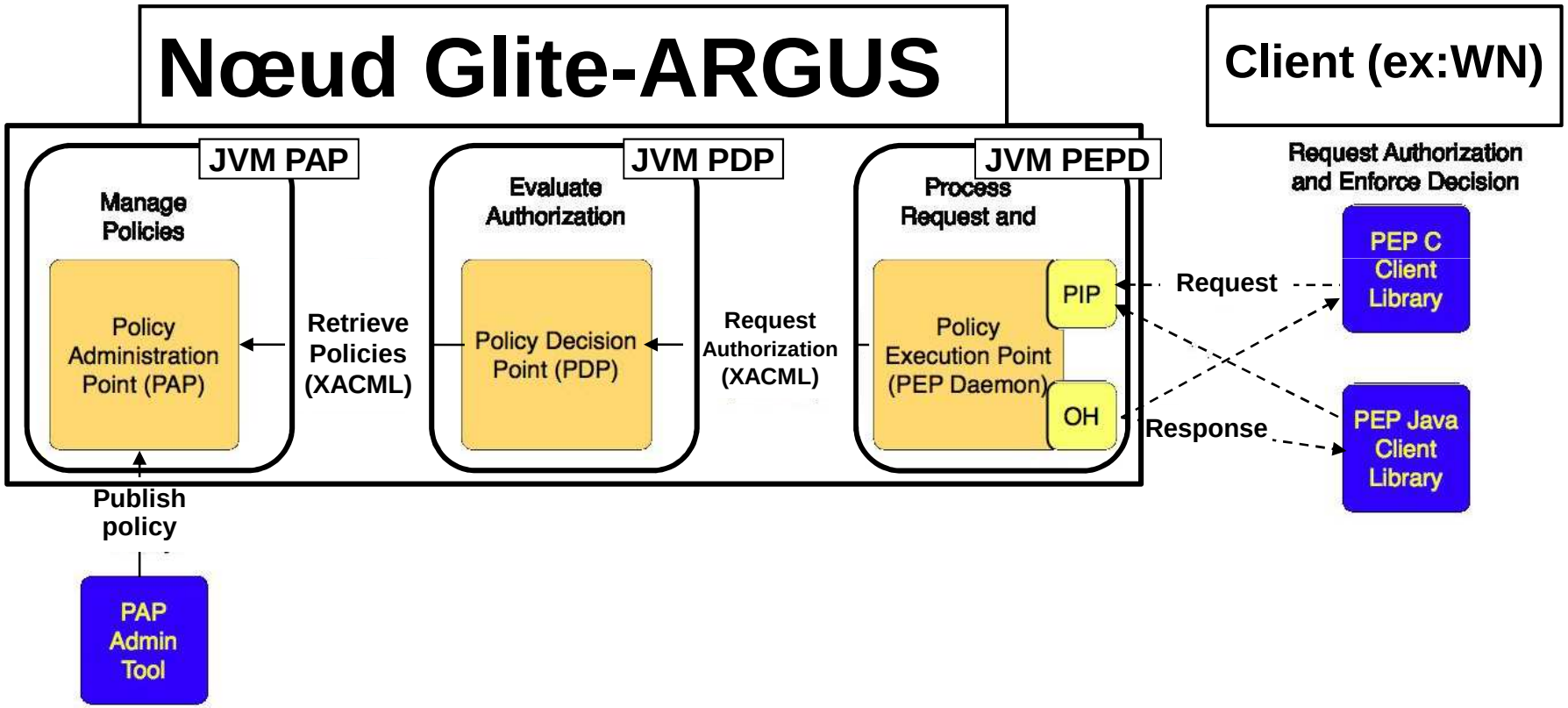
  ⚠ Et non: $GLITE_LOCATION/sbin/glexec

  ⚠ Appel d'un « wrapper »: glexec_wrap.sh

  ⇒reconstruit l'environnement grille nécessaire aux tâches utilisateurs

Glite-ARGUS

Principes de fonctionnement (1)



Source: https://twiki.cern.ch/twiki/bin/view/EGEE/AuthorizationFramework

Principes de fonctionnement (2)

- Gestion des politiques

```
cclcgvmli19_root# /opt/argus/pap/bin/pap-admin lp

default (local):

resource "http://cc.in2p3.fr/wn" {
  obligation "http://glite.org/xacml/obligation/local-environment-map" {
  }

  action "http://glite.org/xacml/action/execute" {
     rule permit { vo="dteam" }
     rule permit { pfqan="/ops/Role=pilot" }
     rule permit { pfqan="/atlas/Role=pilot" }
     rule permit { pfqan="/lhcb/Role=pilot" }
  }
}
```

A une obligation est associé un plugin pouvant ajouter des informations à la réponse
ex: (gid, uid)

Principes de fonctionnement (2)

- Utilisation côté client (Glexec)

```
pepc      = "lcmaps_c_pep.mod"

  "--pep-daemon-endpoint-url https://cclcgvmli18.in2p3.fr:8154/authz"
  "--pep-daemon-endpoint-url https://cclcgvmli19.in2p3.fr:8154/authz"

  "--resourceid http://cc.in2p3.fr/wn"
  "--actionid http://glite.org/xacml/action/execute"

  "--capath /etc/grid-security/certificates/"
  "--pep-certificate-mode implicit"
```

Qui contacter

Quoi demander

Installation au CCIN2P3

- 2 machines virtuelles (VMWare)
  - 1 CPU
  - 2 Go Ram
- Partage du « gridmapdir » via NFS
  - /etc/grid-security/gridmapdir
  - Espace pour « mémoriser » les « mappings »
    - (DN+Role) ⇒ Compte de pool

⇒ Installation temporaire de test !!

⇒La mise en évidence de ses limitations permettra de définir les vrais besoins pour la production

Premiers tests

Submission sur DTEAM

- Description du test (basique)
  - ~400 ou 600 jobs
    - 2 « bulk » jobs
    - 200 ou 300 jobs par « bulk » job
    - 2 WMS du testbed
  - 100 tests / job
  - 2 glexec par test
    - Test de l'identité (/usr/bin/id)
    - Test de l'environnement grille (voms-proxy-info)

Premiers résultats

Test	JVM PEPD	round-robin	#jobs soumis	#tests	#erreurs	Taux d'erreurs	Commentaires
1	256Mo (default)	glexec	400	36998	34656	93,67%	Les deux serveurs (pepd) ne répondaient plus
2	512Mo	glexec	400	39900	79	0,20%	Mauvais round-robin: un serveur sur-chargé et pas l'autre
3	512Mo	Alias (lbnamed)	400	39800	39800	100,00%	Ne fonctionne pas avec les certificats serveurs.
4	512Mo Certificat service	Alias (lbnamed)	400	39800	0	0,00%	Meilleur répartition de la charge sur les 2 serveurs ARGUS
5	512Mo	Alias (lbnamed)	600	59900	228	0,38%	200 erreurs étaient liées à un WN mal-configuré (Taux réel: 0,05%)

Conclusions

Conclusions

- Les jobs pilotes procurent souplesse et efficacité aux expériences
  - Une partie de l'exploitation n'est plus sous le contrôle des sites et migre de facto vers les VOs
- Glexec est une réponse au problème posé par les sites sur les jobs pilotes
  - Il est aussi dans le CREAM CE
- La règle d'or pour un site reste « Un compte n'est utilisé que par une même personne (certificat proxy) »
  - La cible est le WN
  - Attention aux recouvrements de mapping en multipliant les sources de prise de décision de « mappings » (CE, glexec)

Conclusions (suite)

- Plusieurs implémentations pour la mise en place des politiques d'autorisation d'un site pouvant être couplées avec glexec
  - Locale au WN
  - Centralisable via un service
    - Glite-SCAS
    - Glite-ARGUS
- Glite-ARGUS est prometteur
  - Architecture claire et modulaire
  - Langage de politiques expressif et extensible par des plugins
  - Possibilité d'une hiérarchie d'héritage et d'aggrégation de politiques
- Glite-ARGUS doit être déployé de façon à absorber la production utilisant « glexec »
  - Tests à poursuivre pour trouver le bon « Tuning » et l'infrastructure suffisamment robuste
  - Déploiement proposé par défaut semble sous-dimensionné
- Pour une robustesse optimale, la gestion des autorisations localement aux WNs reste la plus sûre pour MUPJ/Glexec
  - Pas de service, pas de panne
  - Le nombre de comptes par pool est proportionnel au nombre de jobs pouvant tourner sur le WN, ce n'est donc pas une limitation comme dans le cas d'un service centralisé
  - Mais la configuration des politiques d'autorisation est à gérer sur chaque WN