



Nouvelles d'ALICE au CC



Quoi de neuf depuis la dernière fois?



- Activités jobs (config, soumission, efficacité)
 - ◆ CE
 - ◆ AFS vs bit-torrent
 - ◆ Efficacité des jobs
- Stockage
 - ◆ Partie T1 : réplication des raw data, xrootd + HPSS
 - ◆ Partie T2 : disk-only



Configuration des CE



- CREAM-CE uniquement depuis automne 2010
- 4 CE :
 - ◆ cccreamceli02.in2p3.fr:8443/cream-bqs-verylong
 - ◆ cccreamceli03.in2p3.fr:8443/cream-bqs-long
 - ◆ cccreamceli05.in2p3.fr:8443/cream-sge-verylong
 - ◆ cccreamceli06.in2p3.fr:8443/cream-sge-long
- Queues 'verylong' adaptées à la reconstruction des raw data, plus consommatrices en mémoire.



Utilisation des CE par les services centraux



- Problème pas trivial car deux systèmes de batch parallèles au CC (BQS/SGE)
- Trouver le juste nombre de jobAgents à laisser en queue
 - ◆ Pas assez → possible manque à gagner si workpoints disponibles rapidement
 - ◆ Trop → problèmes d'expiration de proxy
- Jusqu'à récemment, les jobAgents étaient soumis indifféremment sur les 4 CE
 - ◆ problème si 1 batch lent, 1 batch rapide
 - ◆ le batch lent récupère au fur et à mesure tous les jobs en queue
 - ◆ nécessite de pouvoir équilibrer la charge sur les différents CE
- Nouvelle approche en test

Bilan : mieux mais pas encore top



Installation du software sur AFS



- Solution utilisée par ALICE au CC jusqu'au printemps 2011
- Config initiale ALICE :
 - 1 volume AFS R/W
 - Ne supporte pas la charge induite par 1000 jobs ALICE simultanés
- Config mise en place par X.Canehan
 - 1 volume R/W pour installation par la vobox
 - 1 volume R/O utilisés par les jobs
 - la vobox s'occupe de déclencher la réplication R/W → R/O
- Au niveau du CERN
 - n'ont pas su utiliser cette config et automatiser la réplication
 - développement d'une alternative bit-torrent en cours dans la même période
 - → forte volonté d'utiliser bit-torrent au CC de la part du CERN



Installation du software par bit-torrent

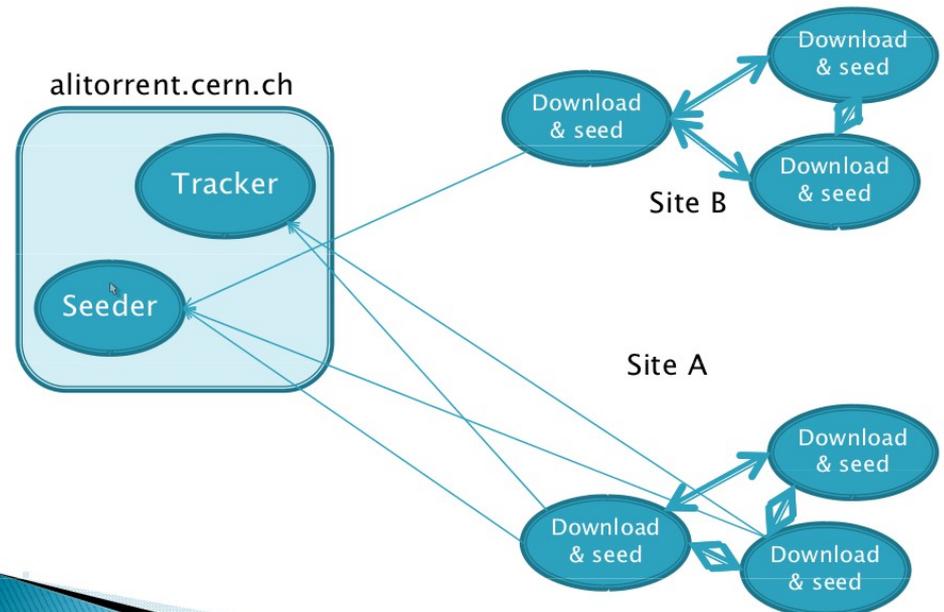


■ Idée :

- ne plus être dépendant de l'espace de stockage partagé
- technologie adaptée à la montée en puissance des besoins LHC
- charge réseau engendrée équivalente (?)
- système de checksum des paquets

- le WN télécharge les .torrent (qq kB) des paquets demandés
- Puis il télécharge les paquets (O(GB))
- Il devient « seeder »

**AFS remplacée par bit-torrent
au CC depuis fin juin
Pas de problème constaté**





Efficacité des jobs



- définition : efficacité = cpu consommé / wall time.
- Efficacité des jobs ALICE basse (~25-30%) en 2011
 - ◆ Sur tous les sites
- Raison :
 - ◆ - tous les jobs ALICE accèdent à un SE du CERN pour obtenir les fichiers OCDB (calibration, alignement etc.)
 - ◆ - 30000 jobs ALICE en régime de croisière
 - ◆ → SE sature, temps de connection énormes, zéro CPU consommé par le worker etc....
- Problème résolu en août

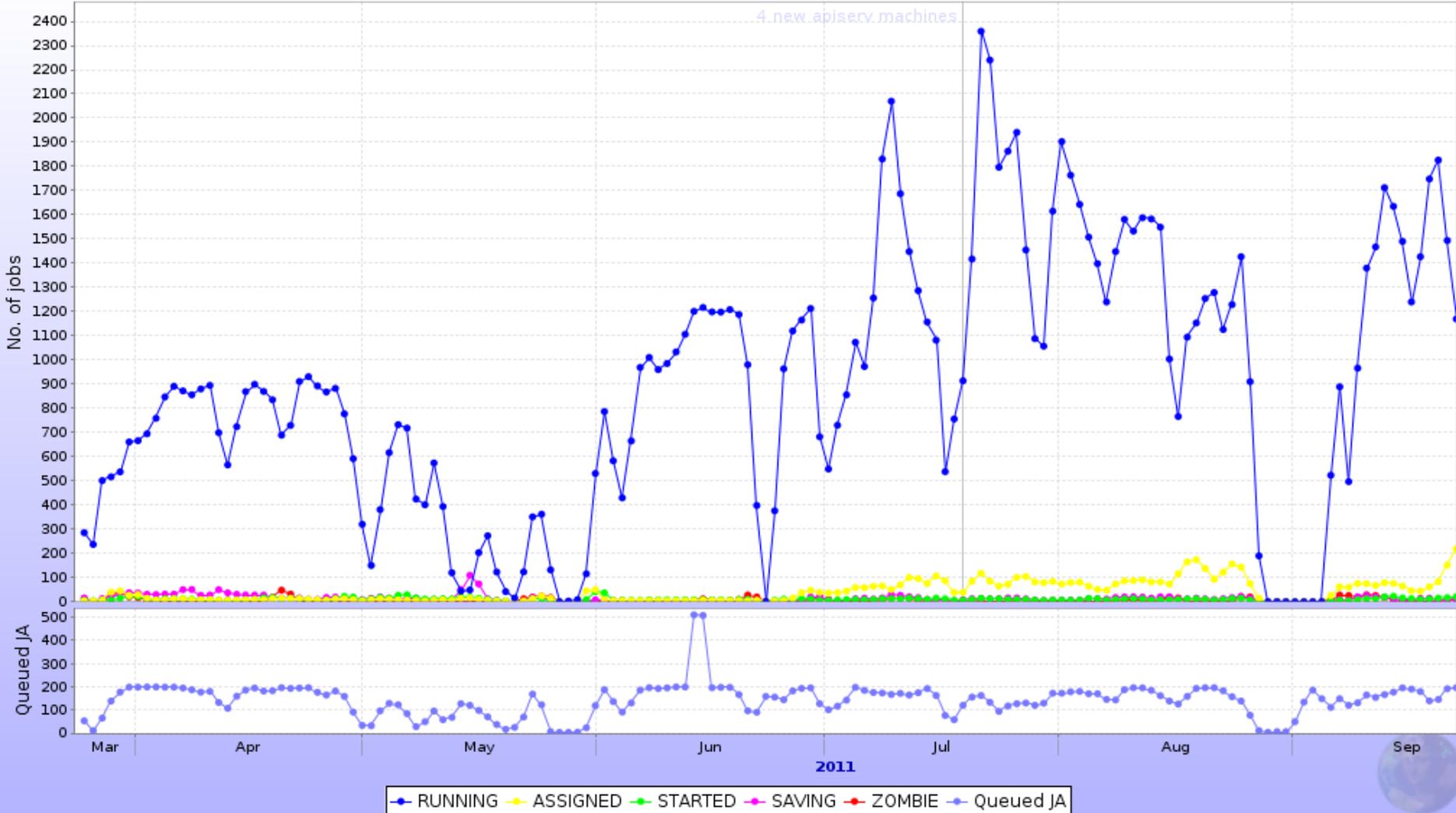
Efficacité actuelle : 90%



ALICE jobs : 6 derniers mois

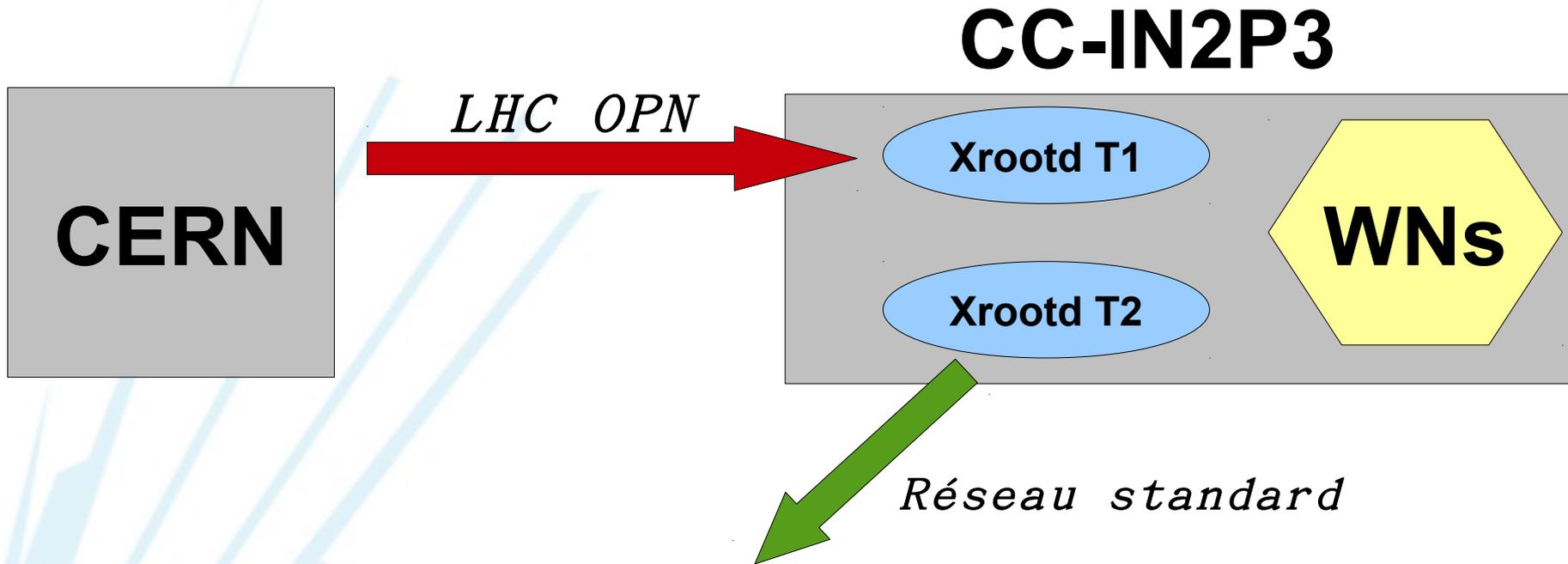


Active jobs in CCIN2P3





Stockage ALICE (1)





Stockage ALICE au CC (2)



- Pledges 2011 : 900 TB disque xrootd + HPSS
 - ◆ 600 TB disk-only → partie Tier2
 - petits fichiers, fichiers utilisateurs etc.
 - ◆ 300 TB buffer-disk pour backend HPSS → partie Tier1
 - reçoit raw data, fichiers ROOT zippés de 2-3 GB
- Aujourd'hui
 - ◆ Tier2 : 220 TB déployés (vs 600 TB pledged)
 - ◆ Tier3 : 110 TB deployes (vs 300 TB pledged)

Nous n'avons déployé que la moitié de nos pledges de disque



Stockage ALICE au CC (3)



- 100% xrootd pour la partie disque
- Arrivée des machines Linux au printemps, abandon progressif de Solaris
 - ◆ → cette étape a facilité GRANDEMENT le déploiement des services de stockage
 - ◆ → situation débloquée avec experts xrootd du CERN
 - ◆ 290 TB de raw transférés cet été
 - 5 % echecs

Le CC tient maintenant son rôle de T1 pour ALICE !



Mes questions



- Comment accélérer le déploiement des serveurs de stockage ?
- Jusqu'à quand ATLAS aura-t-il des machines réservées ?
- Tout le monde fait du reprocessing !!
 - ◆ Je ne comprends pas la priorité donnée aux « reprocessings »
- Gestion des ressources manuelle et obscure
 - ◆ Trop peu de place laissée au fair share
 - ◆ Trop d'interventions manuelles, perte de temps (mon avis)
 - ◆ Objectifs calculés sur CPU consommé et non Wall time
 - → les jobs inefficaces pénalisent tout le monde