

# SIDUS & Reproductibilité

*Single Instance Distributing Universal System*



*Un petit pas pour la reproductibilité...  
Quel pas pour des ressources IT (identiques) ?*

# « Catalyser » l'informatique scientifique : CBP : Maison de la Simulation & Centre d'essais



## Nasa X29

- Cellule de F5
- Moteur de F18
- Servos de F16
- Études
  - Flèche inversée
  - Incidence  $>50^\circ$
  - « *Fly-By-Wire* »

**Le CBP (via son pilote d'essais) réutilise et explore...**



# Une Plate-Forme Expérimentale : Des plateaux techniques

- **Multi-nœuds** : 76 permanents de 4, 24 et 48 nœuds
- **Multi-cœurs** : 100 stations/serveurs/nœuds de 2 à 20 cœurs
- **Multi-shaders** : 28 stations/nœuds avec 20 (GP)GPU différents
- Intégration logicielle (versions de distribution)
- **Paillasse numériques** : expérience/démonstrateur/prototype
- Intégration matérielle (Sparc, PowerPC, ... et ARM)
- **Visualisation scientifique** : plateau 3D
- **Plateau COMOD** : « *Compute On My Own Device* »

**Pour étudiants, enseignants, chercheurs, ingénieurs**



# L'informatique moderne : système de « compliqué » à « complexe »

- **Compliqué** : « *cum plicare* » (plier ensemble)
  - « *le tout est la somme des parties* » (Descartes)
- **Complexe** : « *cum plexus* » (tisser ensemble)
  - « *impossible d'identifier tous les éléments, toutes les relations* »
- Nos « systèmes » de calcul scientifique : le socle de nos « codes »
  - Dépendances logicielles : versions, compilateurs, librairies, ...
  - Système d'exploitation : noyau, services, processus, supervision, ...
  - Nœud : processeur (cœur, L1/2/3, ALU, UC), mémoire, interfaces
  - Réseaux : haut débit/basse latence, bas débit/haute latence
- Assurer (la reproductibilité) pour rassurer (l'utilisateur) ?
  - Quelle marge de manœuvre pour l'opérateur ? Entre matériel & code.
  - Quelles solutions ? Offrir une « archive » du socle.



# Variabilité dans l'espace/temps numérique : Quelle marge de manœuvre ?

- Temps : même machine, instants différents ?
- Espace : même instant, machines différentes ?
- Les solutions :
  - Restauration d'une même image système :
    - Replicator, SystemImager, MondoRescue, ...
    - Kadeploy sur Grid'5000
    - Boot iSCSI avec Back Office Snapshot (sur LVM, ZoL, Btrfs)
  - Installation suivant le même protocole :
    - FAI, Kickstart, Debian-Installer Preseed
  - **SIDUS : *Single Instance Distributing Universal System***



Ce que SIDUS n'est pas...  
Mais ce qu'il partage avec eux !

***SIDUS n'est pas !***

- **LTSP** : *Linux Terminal Server Project*
  - Un serveur, une administration simplifiée des clients
- **FAI, Kickstart, Debian Installer Preseed** :
  - « *Et la machine remplace l'opérateur pendant l'installation* »
- **LiveCD en réseau** :
  - Une image ISO distribuée par le réseau

***Mais SIDUS partage avec eux***

**Boot PXE, TFTP, NFSroot, AUFS**



# Les Deux Propriétés de SIDUS

## Reproductibilité espace/temps

- **Unicité de la configuration**
  - Deux clients SIDUS : le même OS au bit près !
- **Exploitation des ressources locales**
  - Processeurs & mémoire vive exploités : ceux du client !
- **Reproductibilité ? Pour une Instance SIDUS inchangée**
  - Stabilité dans le **temps** (pour un même client)
    - Deux démarrages sur une même machine offre le même système
  - Stabilité dans l'**espace** (pour deux clients différents)
    - Deux clients démarrant au même instant dispose du même système

# SIDUS en 7 Questions : CQQCOQP Pourquoi ?

## Pourquoi ?

- **Uniformiser *de facto*** tous ses « **clients** »
- **Limiter** l'administration à un **unique** système
- **Comparer** les **matériels** avec un socle unique
- **Récupérer** des fluides (Watts & BTU)
- **Rationaliser** l'usage des **postes** de travail
- **Investiguer** un système sous **anesthésie**
- **Assurer** la **reproductibilité** sur l'OS & ses applications



# SIDUS en 7 Questions : CQQCOQP

## Quoi ? Qui ?

### **Pour Quoi ?**

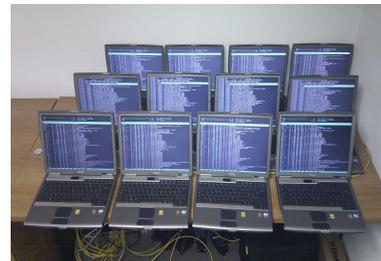
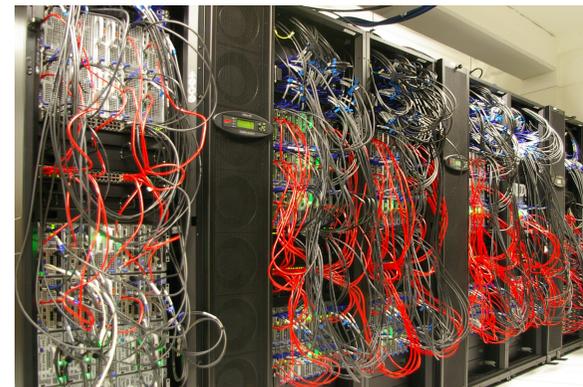
- **Nœuds de cluster de calcul scientifique**
- Postes de salle libre-service
- Stations de travail graphiques
- Paillasse d'expérimentation numérique
- *Compute On My Own Device*

### ***Pour Qui ?***

- Chercheur en informatique scientifique
- Ingénieur en calcul scientifique
- Gestionnaire de salle informatique
- Formateur exploitant des outils informatiques
- RSSI

# SIDUS en 7 Questions : CQQCOQP, la suite Où & Quand ?

- **Centre Blaise Pascal, ENS-Lyon** : salle
  - 12 Neoware en 2010Q1, **22 stations 2013Q4**
- Centre Blaise Pascal, ENS-Lyon : cluster
  - 24 nœuds en 2010Q1, 76 nœuds 2013Q4
- **Centre de calcul PSMN, ENS-Lyon**
  - 100 nœuds 2012Q2, **330 nœuds 2013Q4**
  - Tout [Equip@Meso](#)
- Laboratoires, ENS-Lyon
  - Chimie : 2012Q2, **IGFL 2013Q4**
- **École de physique des Houches**
  - éditions 2011, 2012, 2013



# SIDUS en 7 questions : CQQCOQP, la fin ! Comment ça marche ?

- AUFS : *Another Union File System*
  - Agréger des *File Systems* en un seul : astuce LiveCD
  - 4 étapes :
    1. Monter le NFSroot avec l'OS sur une racine
    2. Montrer un TMPFS sur une seconde racine
    3. Utiliser la glue AUFS entre les deux
    4. Offrir le résultat comme racine du système
  - Comportement d'un FS en *Read/Write* normal
  - Au redémarrage, toute modification disparaît
- Un prérequis : une « maîtrise » de *chroot*



# SIDUS en 7 questions : CQQCOQP, la fin !

## Comment ça s'installe : SIDUS en 8 étapes

- 1) Formation d'un système racine par Debootstrap**
- 2) Création d'un « cordon ombilical » avec l'hôte
  - Montages des `/proc /sys /dev/shm`
- 3) Installation (& purge des paquets spécifiques)
- 4) Adaptation à l'environnement local
- 5) Pointage vers les services tiers
- 6) Création de la séquence de démarrage (AUFs)**
- 7) Importation des noyau & initrd sur serveur TFTP**
- 8) Détachement du système hôte



# SIDUS en 7 questions : CQQCOQP, la fin ! Comment ça s'administre ?

- Une limitation : un `/proc` doit être unique...
  - Forte vigilance sur les opérations qui « tapent dedans »
    - Manipulation de Java, compilation avec optimisation, installation
- **La Bonne** :
  - Passage en chroot, opérations classiques directement
- **La Brute** :
  - Passage en chroot, mise en place du « cordon ombilical »
  - Opérations classiques, démontage du cordon
- **La Truande** :
  - Machine NFSroot en Read/Write, ...

# SIDUS en 7 questions : CQQCOQP, la fin ! Combien ça coûte ?

- Un réseau « idéal » : Gigabit Ethernet (HD local)
  - Mais ça fonctionne en 100 Mb/s !
- Un serveur « idéal » : 4 cpu, 16 Go RAM, 10G, SSD
  - Mais ça fonctionnait avec un v(eau)40z pour 330 nœuds !
- Un client « idéal » : tous identiques
  - Mais ça fonctionne pour TOUTES les gammes au PSMN
- Un installateur/administrateur « idéal » : ;-)
  - Déployé par L. Taulelle à partir de rushs : PSMN
  - Déployé par T. Bellembois via doc en ligne : IGFL

# Manque de reproductibilité ?

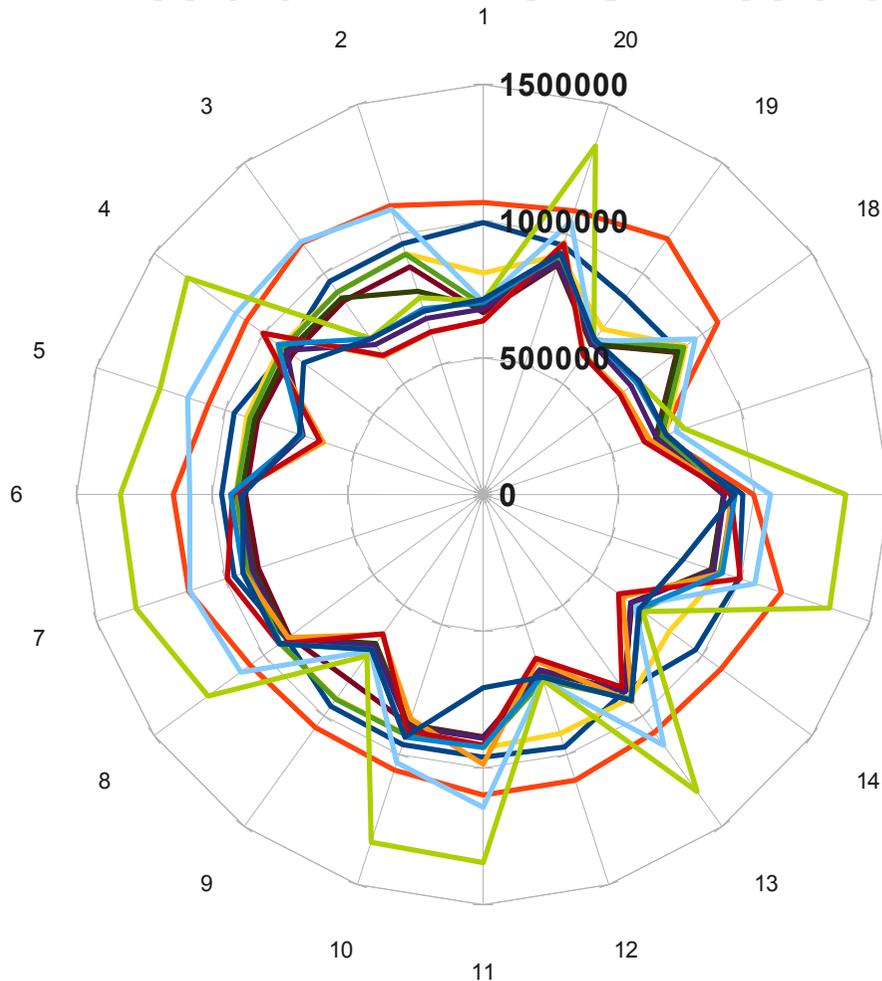
## Illustration par l'exemple : GlusterFS/IOZone

- Objectif :
  - Évaluation de GlusterFS comme /scratch de haute performance
- Plate-forme d'expérimentation : 20 nœuds + infrastructure
  - 20 nœuds Sandy Bridge 2x8 cœurs avec 64 GB de RAM
  - Un système **SIDUS** Debian Wheezy
  - Interconnexion InfiniBand FDR 56 Gb/s
  - **Pas de latence disque : RamDisk BRD/Ext2 et TMPFS de 60 GB**
  - 10 paires GlusterFS : 1 serveur sur RamDisk, 1 client
  - Usage de IOZone3 : 13 tests de lecture/écriture
  - 20 expériences pour un échantillon statistique représentatif

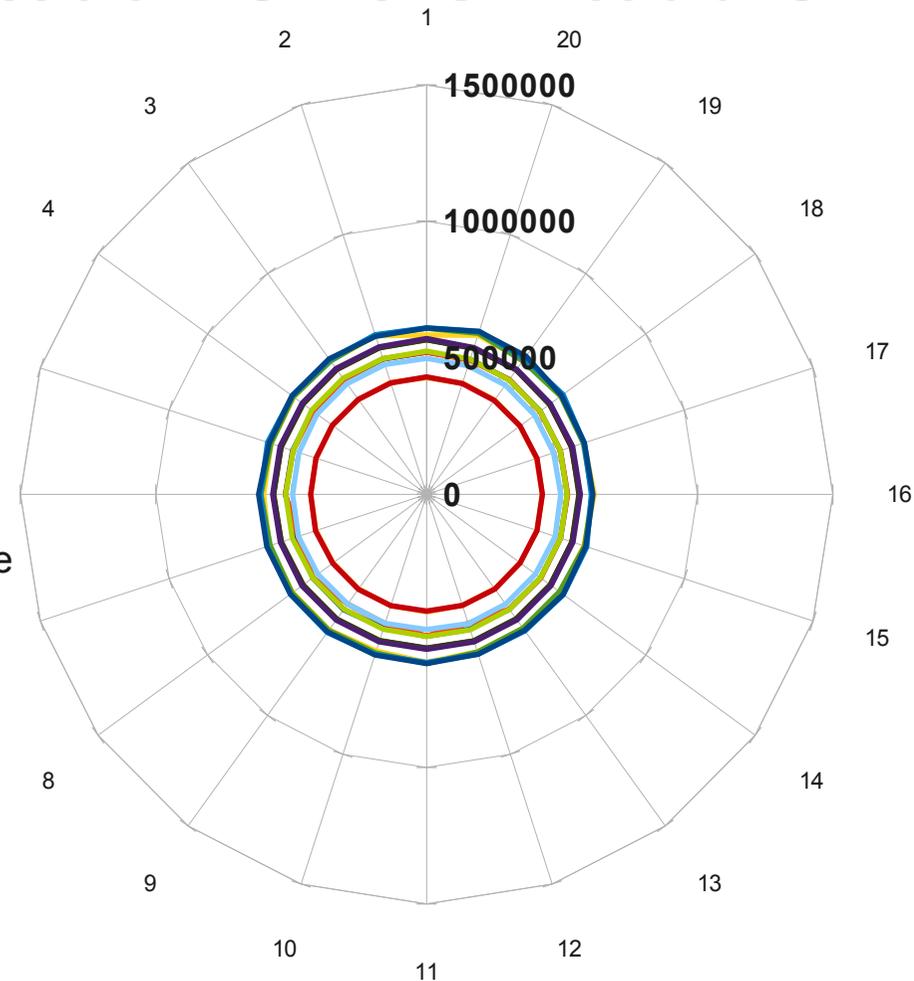
# Jour 1 : lancement des tests & surprises ! Sur les vitesses d'exécution I/O

## Nœud 11 vers Nœud 1

## Nœud 13 vers Nœud 3



- Write
- Rewrite
- Read
- Reread
- Rnd read
- Rnd write
- Bkwd read
- Record rewrite
- Stride read
- Fwrite
- Frewrite
- Fread
- Freread

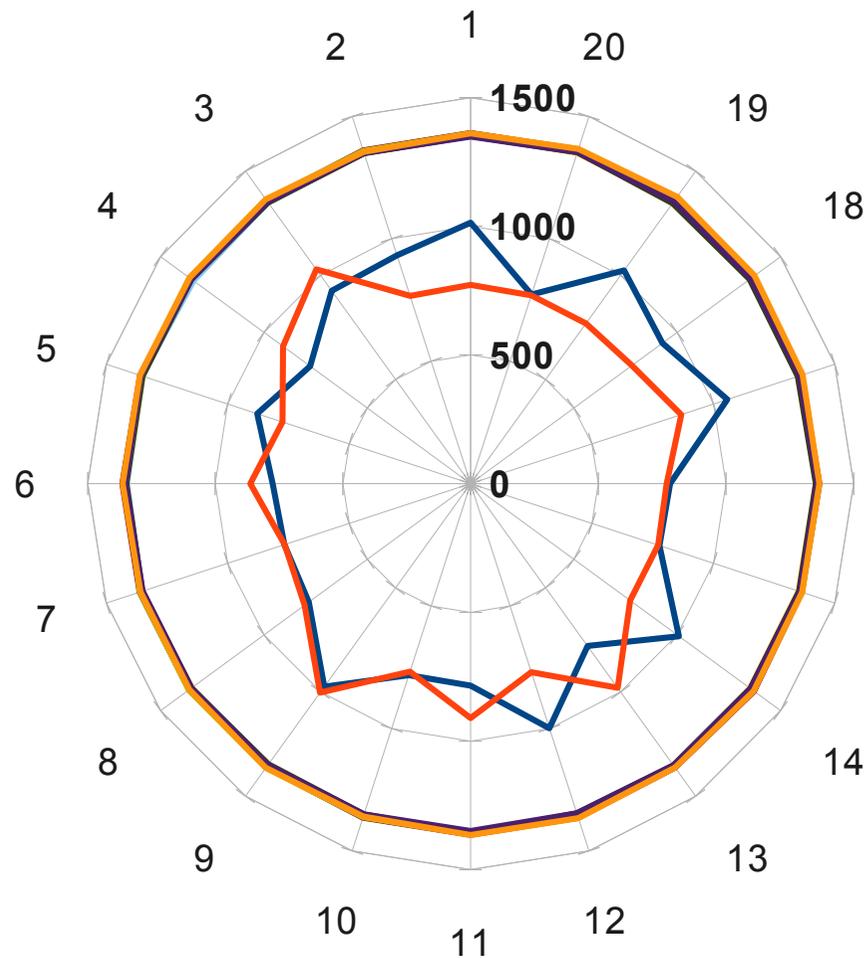


***Great is Better !***

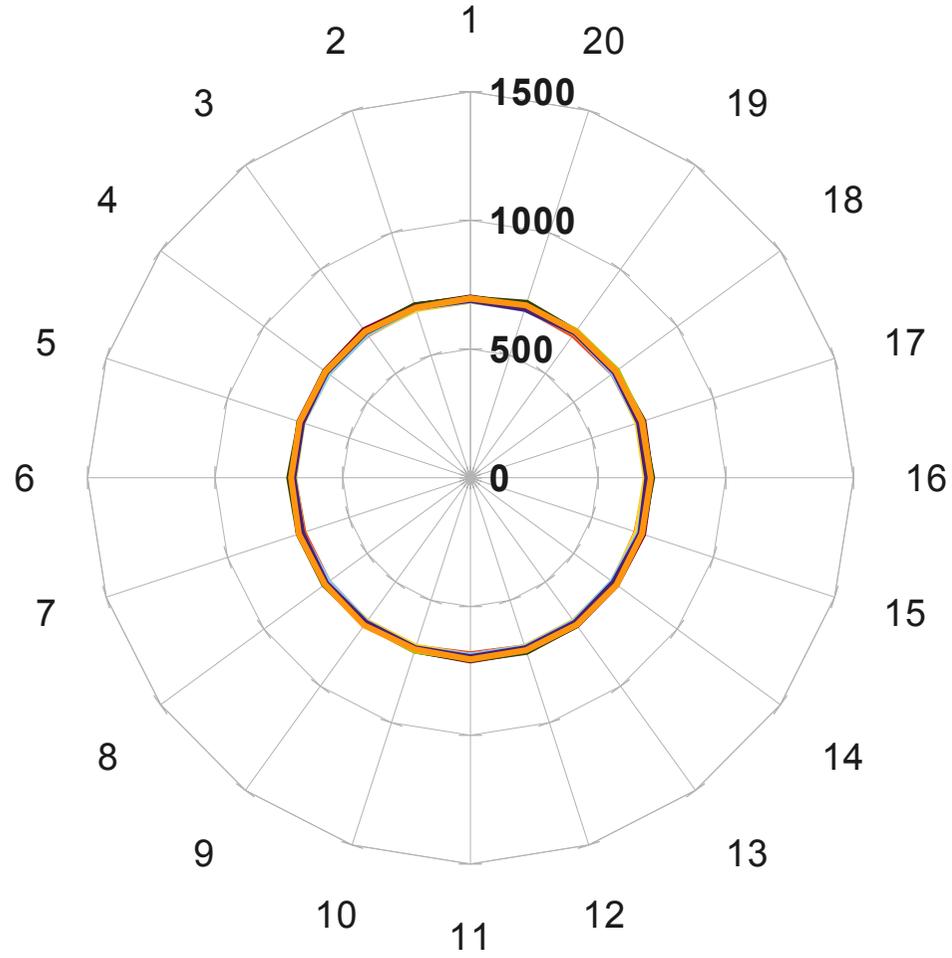
# Jours 1 & 2 : modification & nouveaux tests Sur les durées d'exécution (*User Time*)

Pour les 10 couples, **avant...**

Pour les 10 couples, **après !**



- 11v1
- 12v2
- 13v3
- 14v4
- 15v5
- 16v6
- 17v7
- 18v8
- 19v9
- 20v10

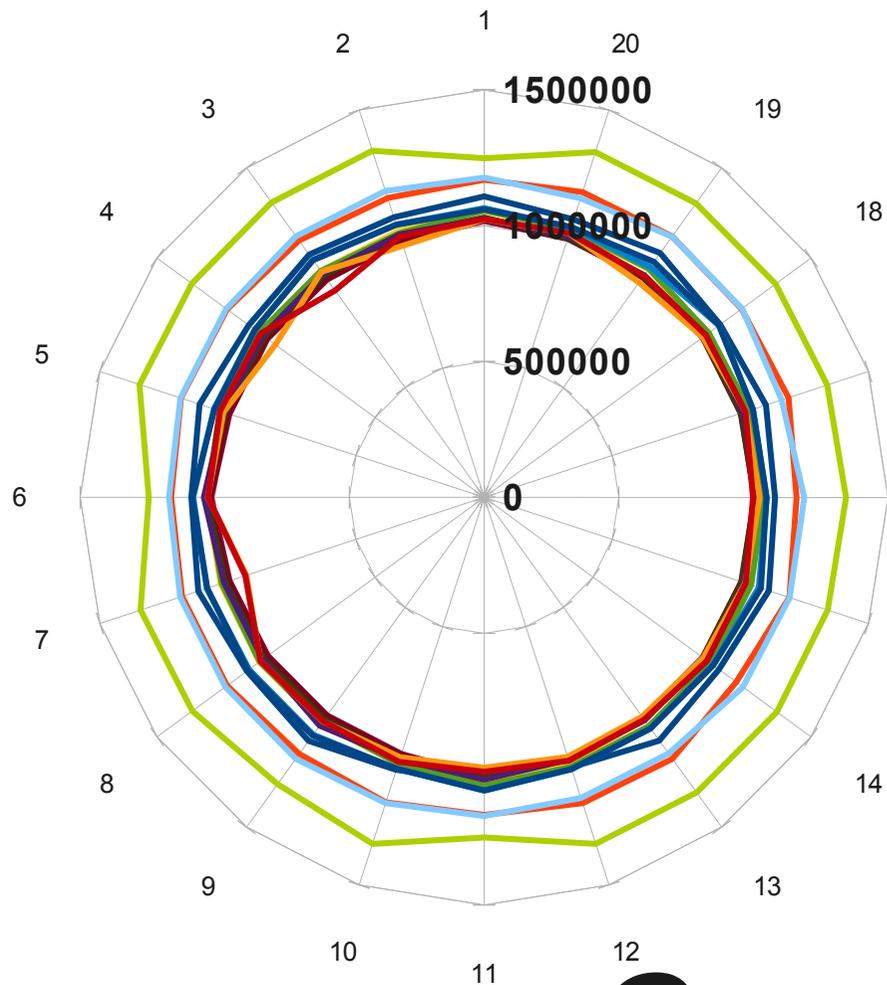


## *Less is Better !*

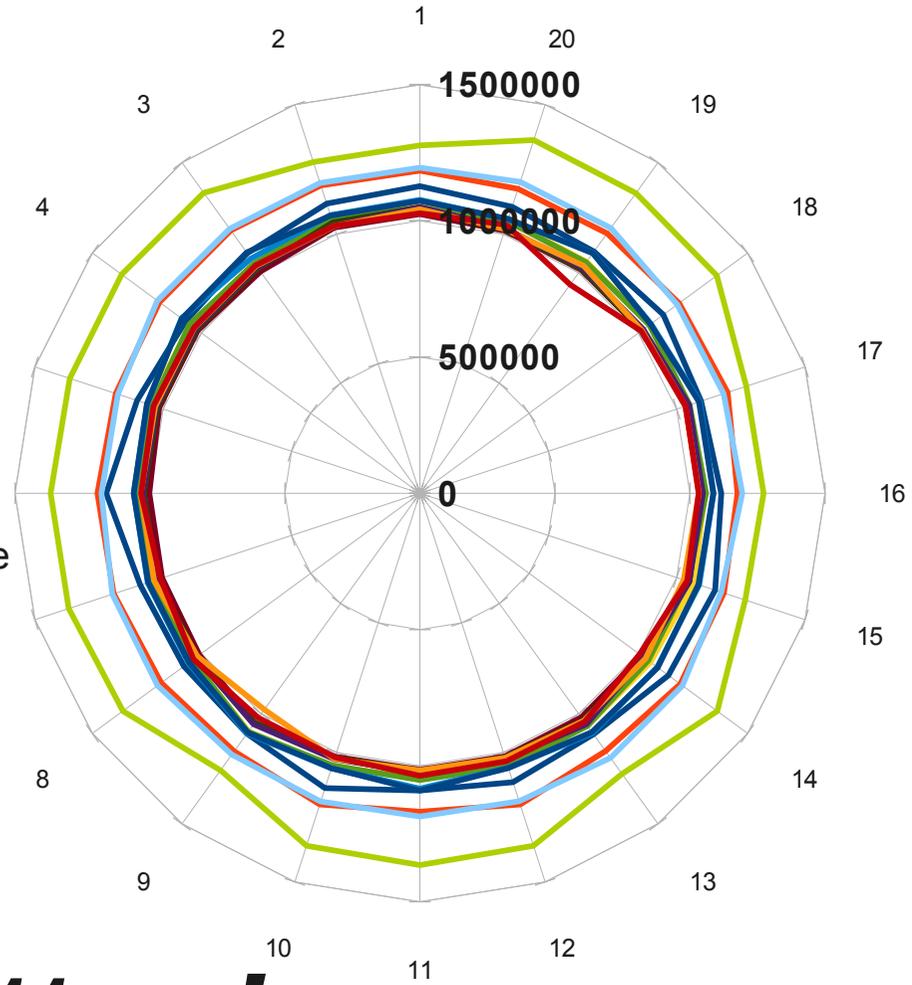
# Jour 2 : et sur les deux couples du début ? Sur les vitesses d'exécution

## Nœud 11 sur Nœud 1

## Nœud 13 sur Nœud 3



- Write
- Rewrite
- Read
- Reread
- Rnd read
- Rnd write
- Bkwd read<sup>6</sup>
- Record rewrite
- Stride read
- Fwrite
- Frewrite
- Fread
- Freread



# *Great is Better !*

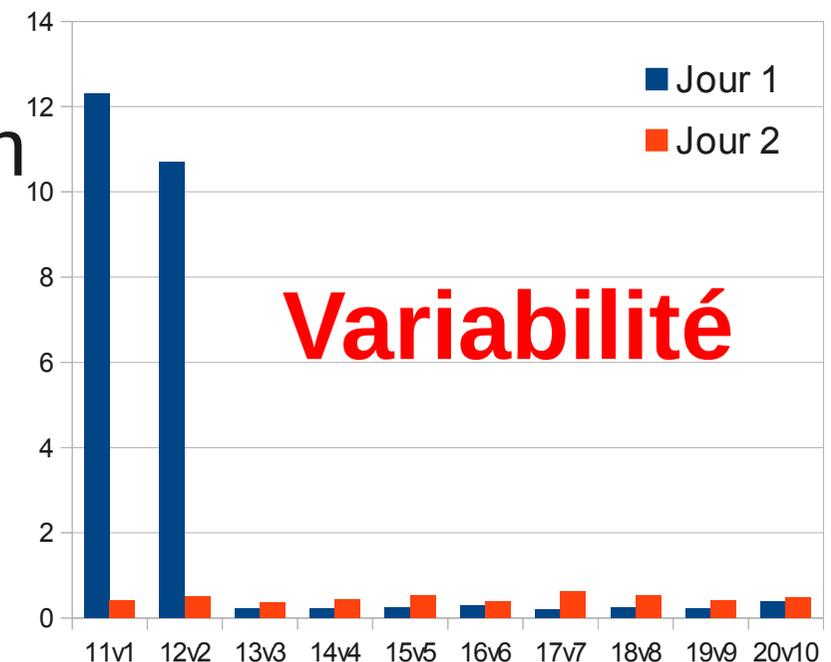
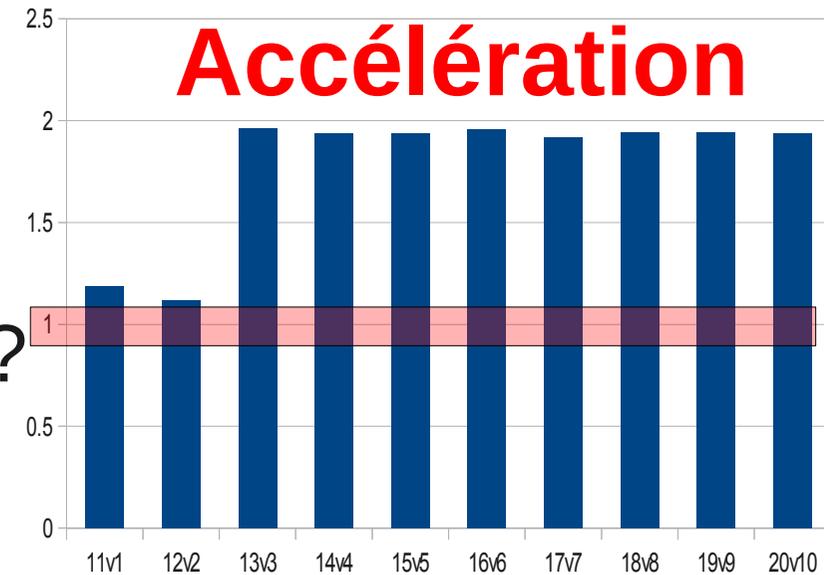
# Quel miracle entre jours 1 & 2 ?

Deux questions : Comment ...

- ... multiplier par 2 la vitesse ?
- ... diviser entre 20/30 sa variabilité ?

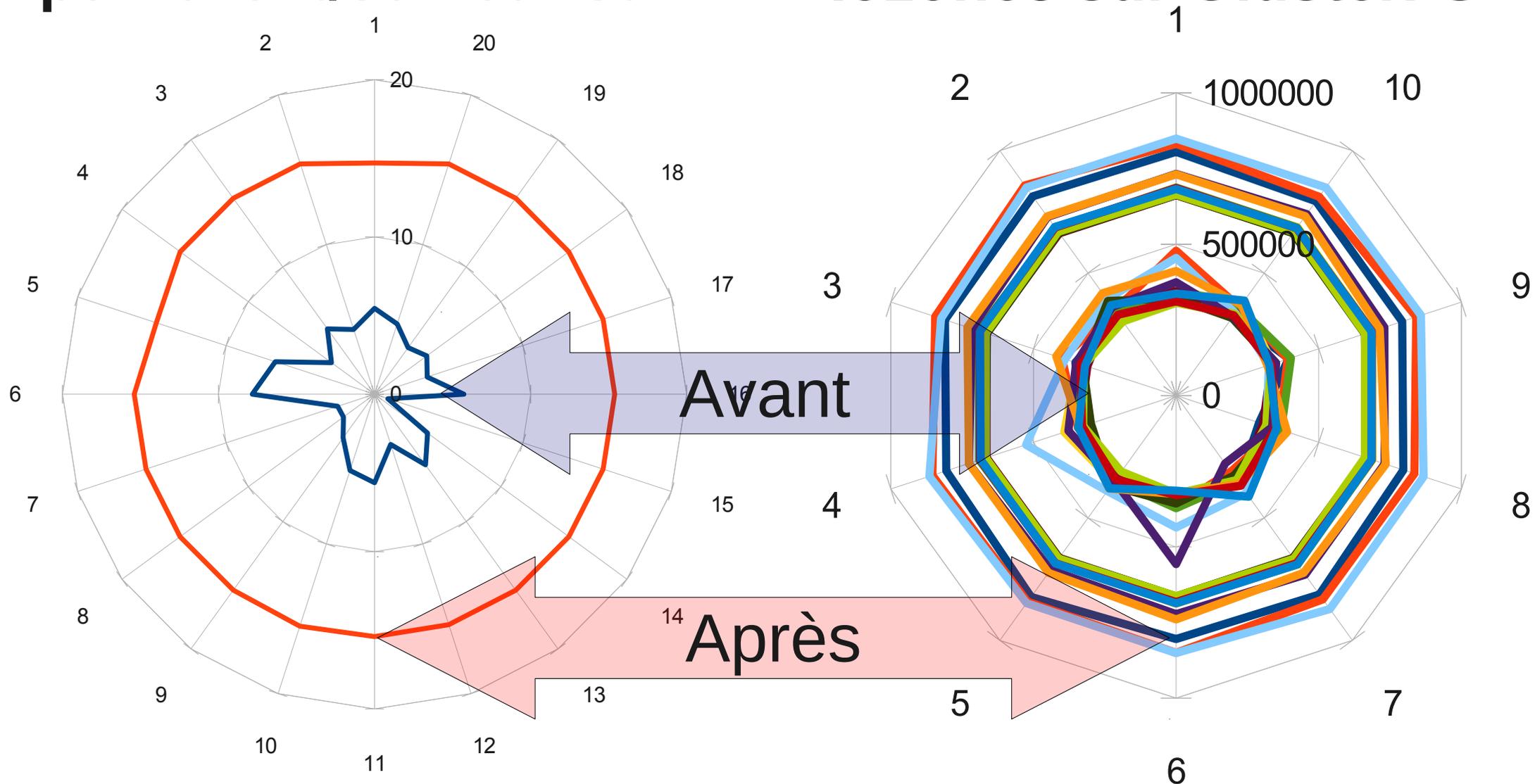
La réponse :

- Optimiser le réseau ? Non
- Optimiser les noyaux des OS ? Non
- Changer le BIOS ? **OUI !!!**
  - BIOS de 1 & 2 en Max Performance
  - BIOS de 3 à 20 par défaut
- Solution : BIOS en Max Perf !



# La non-reproductibilité reproductible ? Sur Equip@Meso

**Iperf client/serveur sur IB**      **iozone3 sur GlusterFS**



# Variabilité « détectée » des vieux matériels

## Des Sources Multiples

- Différence climatique
  - Position dans la baie, Ventilateurs HS, Pâte Thermique,
- Différence électrique
  - Adduction électrique, alimentation électrique,
- Différence électronique
  - « Maladie du condensateur » (régulation de tension)
- Différence informatique
  - Démons « en vrille » : OpenSM, kIPMI (pile pleine),

# Futur de SIDUS

- Valorisation
  - Calcul scientifique, Informatique Scientifique
  - Gestion de parc, Enseignement « à la demande »
- Simplification de l'administration
  - Connexion par SSH à l'instance & opérations classiques
  - Intégration au initrd standard Debian
- Déploiement sur Méso/Grille
  - Intégration d'une VM à un environnement contraint
- *SIDUS everywhere*
  - Lancement de SIDUS hors site (VPN...)

# Conclusion

- SIDUS : l'essayer, c'est l'adopter
  - Instance SciPy 2013 VirtualBox en libre téléchargement
  - Sa démonstration ici ou chez vous...
- L'adopter, c'est :
  - Simplifier à l'extrême l'administration de ressources
  - Récupérer des Watts et des BTU (plus de disques)
  - Comparer simplement de nouveaux matériels, logiciels, ...
- Reproductibilité : maîtriser ce que l'on peut...
  - Variabilité sur le socle OS supprimée
  - Introspection sur la partie matérielle



# Pour en savoir plus :

<http://www.cbp.ens-lyon.fr/sidus/>

## Linux Journal 11/2013

## Poster JRES 2013

## Site Web CBP



**Déduplication extrême d'OS avec SIDUS**  
Emmanuel Quémener & Loïs Taulelle  
Centre Blaise Pascal & Pôle Scientifique de Modélisation Numérique, ENS-Lyon

**Ce que SIDUS signifie :**  
Single Instance Distributing Universal System  
Une instance unique distribuant un système d'exploitation universel

**Ce que SIDUS n'est pas :**

LTSP	FAI ou Kickstart	LiveCD réseau
<b>Les plus</b> • bon recyclage des vieux PC • intégration aux distributions <b>Les moins</b> • toute la charge sur un seul serveur • périphériques locaux difficiles à intégrer	<b>Les plus</b> • automatisation de l'installation • processus mature et maîtrisés <b>Les moins</b> • paramétrage initial • adaptation spécifique par outil tiers	<b>Les plus</b> • unicité de la configuration • rapidité d'installation et de démarrage <b>Les moins</b> • personnalisation difficile • traçabilité quasi-inexistante

**Mais quelques composants que SIDUS partage :**

- ▶ PXE : utilisation d'un démarrage en réseau
- ▶ TFTP : fourniture d'un noyau et d'un système de démarrage
- ▶ NFSROOT : système racine unique partagé par tous les clients
- ▶ AUFS : superposition de systèmes en lecture seule et lecture/écriture

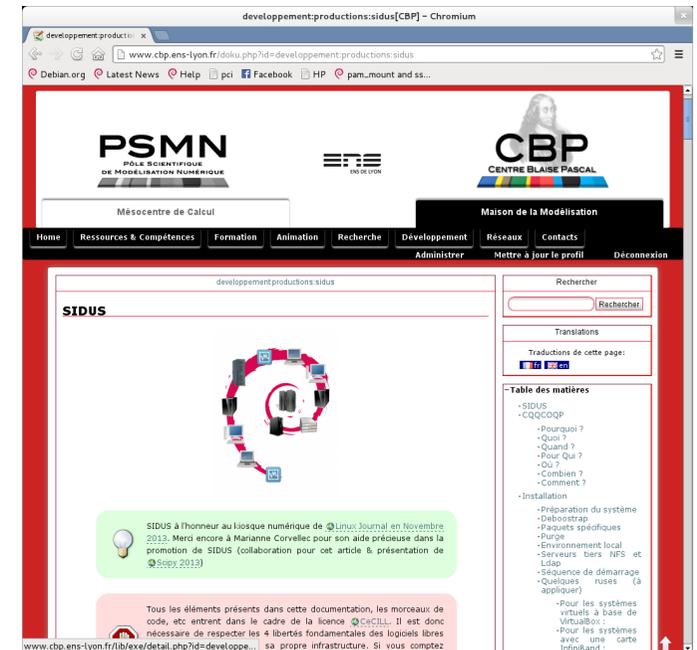
**Ce que SIDUS propose :**

- ▶ **Unicité du système** : tous les clients démarrent exactement le même système (au bit près)
- ▶ **Usage des ressources locales** : les processeurs et mémoire vive exploités sont ceux des clients

**SIDUS en 7 questions-réponses :**

Pourquoi ?	Où & Quand ?	Comment ?
▶ Uniformiser de facto tous les postes ▶ Limiter l'administration à un unique système ▶ Assurer la reproductibilité ▶ Rationaliser l'usage des postes de travail	▶ Centre Blaise Pascal, ENS-Lyon : salle • 12 clients légers bootés en mars 2010 • 22 stations avec GPU différents fin 2013 ▶ Centre Blaise Pascal, ENS-Lyon : cluster • 24 nœuds en mars 2010 • 76 nœuds permanents fin 2013 ▶ Centre de calcul PSMN, ENS-Lyon • 100 nœuds mi 2012 en qualification • 330 nœuds mi 2013 dont Equip@Meso ▶ Laboratoires, ENS-Lyon • Laboratoire de Chimie : été 2012 • Laboratoire LBMC & GFL : autonome 2013 ▶ Ecole de physique des Houches • éditions 2011, 2012, 2013 : jusqu'à 60 utilisateurs	<b>Socle AUFS</b> ▶ AUFS pour Another Union File System ▶ Un système NFSroot en lecture seule ▶ Un système TFTPFS en lecture/écriture ▶ AUFS comme glue entre les deux systèmes <b>Installation en 6 étapes, 3 fondamentales</b> 1 Formation d'un système racine par Debootstrap 2 Création de la séquence de démarrage (AUFS) 7 Importation des noyaux & initrd sur serveur TFTP <b>Administration simplifiée</b> ▶ Passage dans l'instance par chroot ▶ Application des commandes « standard » ▶ Montage des dossiers « système » au besoin
<b>Pour Quoi ?</b> ▶ Nœuds de cluster de calcul scientifique ▶ Postes de salle libre-service ▶ Stations de travail graphiques ▶ Paillasses d'expérimentation numérique ▶ COMOD pour Compute On My Own Device • démarrage d'un poste complet sous SIDUS • démarrage de SIDUS dans une machine virtuelle	<b>Pour Qui ?</b> ▶ Chercheur en informatique scientifique ▶ Ingénieur en calcul scientifique ▶ Gestionnaire de salle informatique ▶ Formateur exploitant des outils informatiques ▶ RSSI	<b>Combien ?</b> ▶ 1 réseau idéalement 1 Gb/s (~ débit disque) ▶ 1 serveur (virtuel) avec pour 100 clients : • 2 CPU • 8 Go de RAM, • 50 Go d'espace par architecture complète ▶ 1 personne motivée pendant 1 journée

**Pour en savoir plus :** <http://www.cbp.ens-lyon.fr/sidus/>



# Iconographie

- [http://en.wikipedia.org/wiki/Antikythera\\_mechanism](http://en.wikipedia.org/wiki/Antikythera_mechanism)
- <http://www.nasa.gov/centers/dryden/news/FactSheets/FS-008-DFRC.html>
- [http://en.wikipedia.org/wiki/Antikythera\\_mechanism](http://en.wikipedia.org/wiki/Antikythera_mechanism)
- <http://congrex.nl/ics0/Papers/Session%2014a/FCXNL-10A02-1977297-1-BERGEI>
- [http://upload.wikimedia.org/wikipedia/commons/8/8b/Babbage\\_Difference\\_Engine](http://upload.wikimedia.org/wikipedia/commons/8/8b/Babbage_Difference_Engine)
- [http://www.earsel.org/Advances/2-1-1993/2-1\\_22\\_Harger.pdf](http://www.earsel.org/Advances/2-1-1993/2-1_22_Harger.pdf)
- <http://en.wikipedia.org/wiki/File:NewmarkAnalogueComputer.jpg>
-