



Club des Utilisateurs de Micro-ordinateurs dans  
l'Éducation – Journée virtualisation, mars 2008

---

# Linux VServer et Redhat cluster

Virtualisation sous Linux et cluster haute-  
disponibilité, une solution de consolidation  
économique

Xavier Montagutelli  
Université de Limoges  
Service Commun Informatique  
[xavier.montagutelli@unilim.fr](mailto:xavier.montagutelli@unilim.fr)





# Licence

---

Copyright (c) 2005 Stéphane Larroque, Xavier Montagutelli

Copyright (c) 2006, 2007, 2008 Xavier Montagutelli

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

<http://www.gnu.org/licenses/licenses.html#FDL>



# Plan

---

- Introduction
- Virtualisation par « conteneur » ou « isolateur »
- Linux VServer
- Red Hat Cluster Suite
- Linux VServer en cluster
- Conclusions & perspectives



# Introduction

---

- Concrètement, pourquoi virtualiser ? Un point de vue personnel :
  - Ne pas « mélanger » plusieurs services sur une même machine : maintenance, délégation de l'administration, risques d'intrusion
  - Rapidité de déploiement : test ou production, par création de nouvelle machine ou duplication
  - Mieux utiliser les ressources
- Début avec VServer en 2004, sur une machine dédiée à l'hébergement web (et VMware ESX fin 2004)



## Introduction (2)

---

### □ Inconvénients

- Multiplication des systèmes à maintenir – mais tous identiques
- Perte de maîtrise / craintes d'une technologie non maîtrisée par certaines personnes ?



# Plan

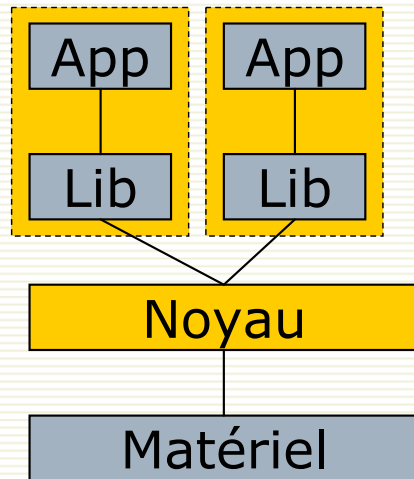
---

- Introduction
- Virtualisation par « conteneur » ou « isolateur »
- Linux VServer
- Red Hat Cluster Suite
- Linux VServer en cluster
- Conclusions & perspectives



# Virtualisation par « conteneur »

- ❑ **Séparation des applications**, regroupées dans des « cages » étanches
- ❑ Chaque cage contiendra un ou plusieurs processus
- ❑ La virtualisation et le cloisonnement sont réalisés **par le noyau**
- ❑ Ancêtre UNIX : **chroot** (isolation au niveau du système de fichiers uniquement)



...



## Virtualisation par « conteneur » (2)

---

- ❑ Exemples : BSD Jails, Solaris Zones, **Linux VServer**, OpenVZ, Virtuozzo (Windows)
- ❑ Un vocabulaire variable : container, serveur privés virtuels (VPS), jails, ...
- ❑ Le noyau répartit les ressources (mémoire, CPU, IO), classiquement, en utilisant des mécanismes déjà éprouvés et optimisés (ordonnanceur de tâches, gestionnaire mémoire, etc.)
- ❑ Performances excellentes



# Virtualisation par « conteneur » et Linux

---

- ❑ Pas de solution « native » et complète sous Linux
- ❑ Des projets anciens : Linux-VServer, OpenVZ
- ❑ Des composants déjà existant dans le noyau
- ❑ Un intérêt croissant et une volonté d'intégrer des composants manquants dans le noyau
  - 2.6.18 : *UTS namespace*
  - 2.6.24 : *PID namespace*
  - 2.6.24 : *control groups* (cgroups), nommé dans un premier temps *process container*
  - 2.6.24 : *network namespace*
- ❑ Des éléments qui pourraient servir pour du checkpoint/restart, de la migration de processus, des politiques d'ordonnancement par usager, ...



# Plan

---

- Introduction
- Virtualisation par « conteneur » ou « isolateur »
- **Linux VServer**
- Red Hat Cluster Suite
- Linux VServer en cluster
- Conclusions & perspectives



# Linux VServer – Introduction (1)

---

- ❑ Idée : séparer l'espace utilisateur d'un système GNU/Linux (« hôte ») en unités distinctes (« serveurs privés virtuels » ou « vservers »)
- ❑ <http://linux-vserver.org/>, liste de diffusion, IRC
- ❑ Début du projet en 2001, utilisable depuis 2003
- ❑ Une équipe restreinte, stable et réactive



# Linux VServer – Introduction (2)

---

## □ Historique

- Liste de diffusion : 2001
- Version 1.0, novembre 2003 (Linux 2.4.20)  
patch : 1146 lignes ajoutées ou modifiées
- Version 1.2, décembre 2003 → janvier 2005  
patch > 2000 lignes
- Version 2.0, août 2005 (Linux 2.6.12)  
patch ≈ 10000 lignes
- Version 2.2.0, nov. 2006 (Linux 2.6.20)
- Version 2.2.0.6, fév. 2008 (Linux 2.6.22.18)  
patch ≈ 17000 lignes, 458 fichiers



## Linux VServer – Introduction (3)

---

- ❑ Minimise la surcharge : une approche par **isolation** plus que par **virtualisation** (moins générique ?)
- ❑ Un projet en « marge » du noyau, pas de volonté d'intégration – mais suit les évolutions actuelles
- ❑ Pas de support par les « grandes » distributions Linux (Redhat ou Suse) – intégré dans Debian stable
- ❑ Pour l'installer « à la main » :
  - Patch sur le noyau Linux  
(`patch-<version_linux>-vs<version_vserver>`)
  - Commandes utilisateurs (`util-vserver`)



## VServer – Isolation de processus

---

- ❑ **Contexte** : nouvelle structure du noyau, identifié par un entier
- ❑ Chaque processus fait partie d'un contexte
- ❑ Interactions entre processus (signaux, IPC...) limitées à un contexte ( $\Rightarrow$  isolation plus que virtualisation)
- ❑ Contexte de l'hôte : 0
  - Peut créer de nouveaux contextes
  - Peut changer de contexte
- ❑ Contexte « spectateur » : 1
  - Peut voir les processus de tous les contextes
- ❑ Un contexte  $\approx$  un vserver



# VServer – Jouer avec les contextes

```
# chcontext --xid 33 /bin/bash  
New security context is 33
```

```
# sleep 1000 &
```

```
# ps auxw
```

```
USER PID COMMAND  
root 4400 /bin/bash  
root 4418 sleep 1000  
root 4419 ps auxw
```

```
# ps auxw | grep sleep  
[1]+ Terminated sleep 1000
```

```
# ps auxw | grep sleep  
root 4421 grep sleep
```

```
# vps auxw | grep sleep  
USER PID CONTEXT COMMAND  
root 4418 33 sleep 1000  
root 4427 1 grep sleep
```

```
# vkill -xid 33 4418
```



## VServer – Isolation réseau

---

- ❑ L'hôte dispose de plusieurs adresses réseaux (en général des alias sur une seule interface physique)
- ❑ Les processus d'un vserver sont limités à une (ou plusieurs) adresse(s). Plus précisément, les processus vont être rattachés à un « *network context* »
- ❑ Attention, les applications de l'hôte doivent être « bindées » sur l'adresse IP qui lui est dédiée



## VServer – Isolation système de fichiers

---

- ❑ Chroot
- ❑ Nouvel attribut du système de fichiers pour se prémunir de l'évasion (barrier)
- ❑ Utilisation des espaces de noms (namespaces) de la couche VFS : chaque VServer a son namespace et une vue différente du FS
- ❑ Possibilité d'associer un fichier a un contexte
  - Clé d'accès
  - Nécessaire pour avoir une limite disque par VServer et des quotas par VServer dans le cas d'une partition partagée



# VServer – Limitation du super-utilisateur

---

## □ Capacités

- Norme POSIX, partiellement supportée depuis Linux 2.2
- Jeton présenté par un processus pour prouver qu'il est autorisé à faire une action
- Exemple : créer un fichier périphérique (MKNOD)
- Par défaut, un VServer aura une limitation de ses capacités  $\Rightarrow$  *root* ne pourra pas tout faire

## □ Nouvelles capacités (*context capabilities*)

Exemple :

- CAP\_NET\_RAW trop fort. Mais sans lui, pas de ping...
- Solution : VXC\_RAW\_ICMP



# VServer – Isolation et extension de /proc

---

- /proc
  - Système de fichiers virtuel
  - Accès (lecture ou lecture/écriture) aux informations du noyau
- Nécessaire dans un vserver : uptime, liste des processus, type de cpu, mémoire utilisée, points de montage, ...
- Mais pas tout
  - Les processus des autres contextes n'apparaissent pas
  - Certaines entrées sont « cachées » à l'aide d'attributs supplémentaires
- Extensions sous /proc/virtual/ et /proc/virtnet/



## VServer – Limiter les ressources

---

- ❑ Coopération des processus et allocation des ressources : par le noyau (classiquement)
- ❑ ulimit par vserver : limitation de la mémoire, du nombre de processus, ...
- ❑ Consommation de CPU limitable par un algorithme « seau de jeton »
- ❑ Disque
  - Une partition par vserver...
  - ... ou utiliser le marquage des fichiers par contexte



## VServer – Autres éléments

---

- Virtualisation d'informations systèmes
  - Nom d'hôte, version et release d'OS, type de machine, processeur (*utsname*, `uname -a`)
  - Uptime
  - Quantité de mémoire disponible (en fonction des limites fixées)
- Unification
  - Partager des fichiers entre VServer à travers des liens en dur, idéalement tout / sauf ...
  - Gain de disque
  - Mise à jour



## VServer – Limites

---

- ❑ Interface de boucle locale : pseudo interface dans la branche de développement 2.3
- ❑ Support IPv6 : dans 2.3
- ❑ NFS en mode noyau : non
- ❑ Migration à chaud de VServer : non
- ❑ Des VServers sur plusieurs VLAN : possible
- ❑ Netfilter par VServer : non, nécessite une virtualisation complète de la couche réseau
- ❑ Pas d'interface de gestion, pas de commande « vserver-top »
- ❑ Nécessité de bien connaître GNU/Linux et les mécanismes ci-dessus...



## VServer – Construire un VServer

```
# vserver vs build -m yum --context 33 \  
  --hostname vs1.cume.org \  
  --interface vs=eth0:10.1.1.2 \  
  --flags virt_uptime,virt_mem \  
  -- -d rhel5
```

- ❑ Construit un nouveau VServer en téléchargeant et installant avec yum les paquets rpm nécessaires pour une distribution « rhel5 »
- ❑ Plus rapide : on peut aussi faire une copie d'une archive tar pour le répertoire du chroot
- ❑ P2V : on peut utiliser la méthode « rsync » pour virtualiser un serveur physique existant



# VServer – Démarrer le VServer

---

```
# vserver vs start
```



# VServer - Surveiller

```
# vserver-stat
CTX  PROC   VSZ    RSS  userTIME  sysTIME  UPTIME  NAME
0    40    81.9M  7.9M  0m06s69   1m06s94  2h25m57  root server
33   9    37.6M  12.4M  0m00s20   0m00s25  0m05s10  vs
```

- Autres commandes :
  - vps
  - vtop
- /proc/virtual



## VServer – Gérer les paquets

```
# vyum vs -- install mysql-server  
# vserver vs pkgmgmt internalize
```

- ❑ Gestion des paquets possible depuis l'hôte
- ❑ Ou rendre le VServer « indépendant »



# VServer – Bilan (1)

---

- Un VServer est composé :
  - D'un répertoire contenant sa configuration, sous `/etc/vservers/<vserver>/`  
Exemple : fichier `interfaces/0/ip`
  - Répertoire contenant l'arborescence du VServer (dédié au « chroot »), sous `/vservers/<vserver>/`
  - Répertoire contenant les données du gestionnaire de paquets (si gestion externe), sous `/vservers/.pkg/<vserver>/`



# VServer - Bilan (2)

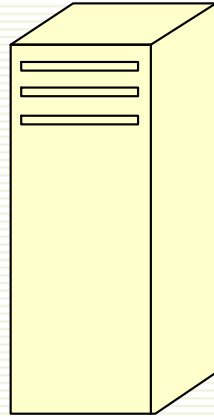
```
sshd
syslog
ntp
```

Hôte

192.168.1.10

192.168.1.23

192.168.1.45

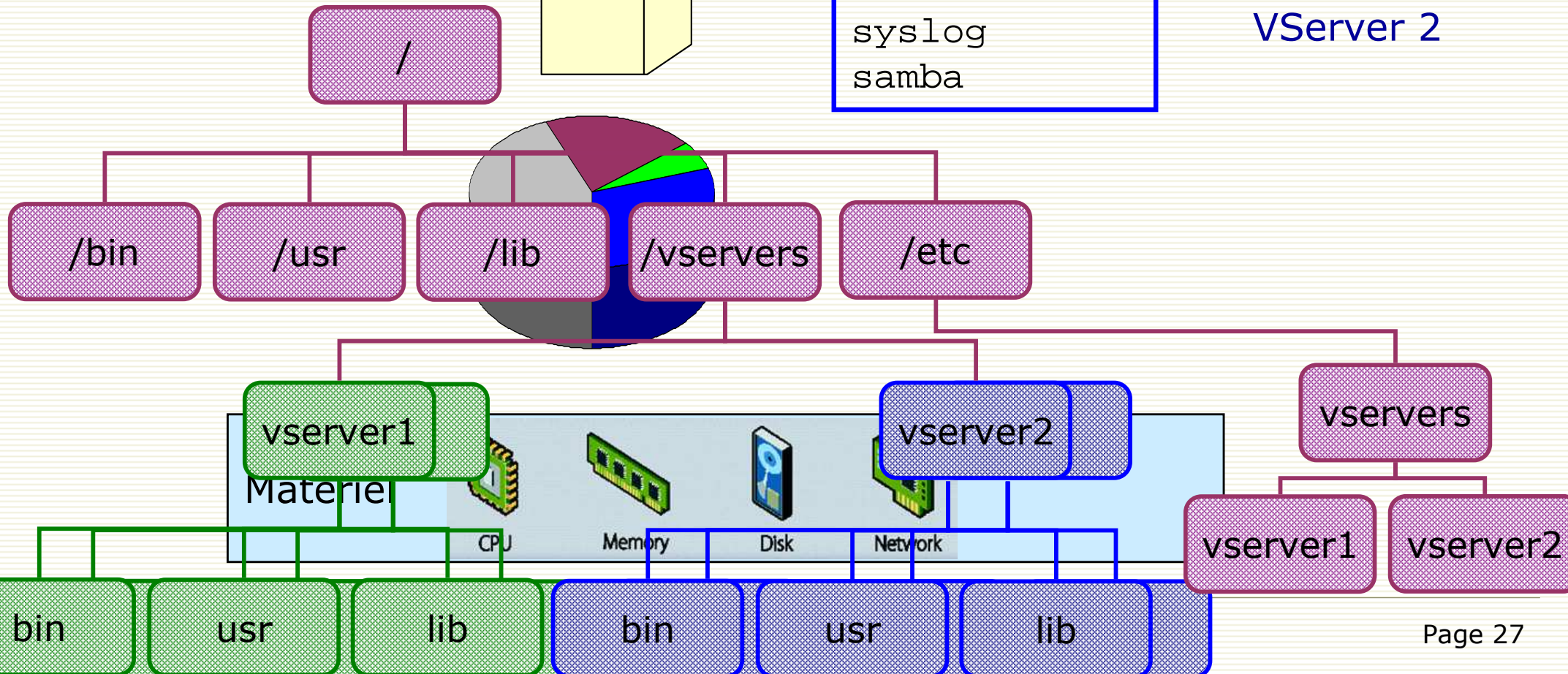


```
sshd
syslog
httpd
mysql
```

VServer 1

```
sshd
syslog
samba
```

VServer 2



ME - Virtualisation



# Plan

---

- Introduction
- Virtualisation par « conteneur » ou « isolateur »
- Linux VServer
- Red Hat Cluster Suite
- Linux VServer en cluster
- Conclusions & perspectives



## Red Hat Cluster Suite

---

- ❑ *Cluster* (grappe) : plusieurs machines (physiques) qui participent à une même infrastructure globale
- ❑ Objectifs : *haute-disponibilité, répartition de charge*, ou accès partagé à un espace disque
- ❑ Les membres du cluster sont identiques et interchangeables
- ❑ Ils partagent l'accès à des ressources communes
- ❑ Un cluster héberge des *services*, surveillés, qui peuvent tourner de manière transparente sur n'importe quel membre du cluster



## Red Hat Cluster Suite (2)

---

- ⇒ Un cluster découple l'hôte (machine + noyau + système de base) du service hébergé (application)
- Livré avec *Red Hat Enterprise Linux 5 **Advanced Platform***
- La suite des composants Red Hat est bien intégrée, cohérente, aisée à mettre en œuvre
- Licence GPL – disponible sous d'autres distributions (CentOS, Debian, Ubuntu, ...)
- Basé sur des normes ouvertes (AIS, LSB, ...)



## Red Hat Cluster Suite (3)

- Composants de base :
  - CCS : accès à la configuration du cluster, propagation des modifications
  - CMAN : gestion du cluster : quorum, appartenance au cluster, heartbeat
  - Fencing : arrêt forcé des machines défectueuses (garantie d'intégrité des données)
  - DLM : gestion des verrous (*lock*) distribués
- Composants « utiles » :
  - CLVM : LVM avec des espaces partagés entre plusieurs machines
  - GFS2 : accès simultané à un même système de fichiers depuis plusieurs machines
  - GNBD
  - rgmanager : gestionnaire de ressources
  - LVS (Linux Virtual Server) : mécanisme de répartition de charge
- Interface d'administration par le web, *Conga* ; composant serveur (*luci*) et agents sur les membres du cluster (*ricci*)



## Red Hat Cluster Suite (4)

---

- En haute-disponibilité, le cluster exécute des *services*
- Chaque service est composé d'une ou plusieurs *ressources*. Exemples :
  - Adresse IP (adresse d'un serveur web)
  - Système de fichiers (contenant les données du serveur web)
  - Script de type LSB (lancement/arrêt d'apache)
- Un service peut-être restreint à un *domaine de failover* (sous-partie du cluster). Utilités :
  - Limiter l'exécution d'un service à certains membres
  - Ou indiquer un ou des hôtes préférentiels



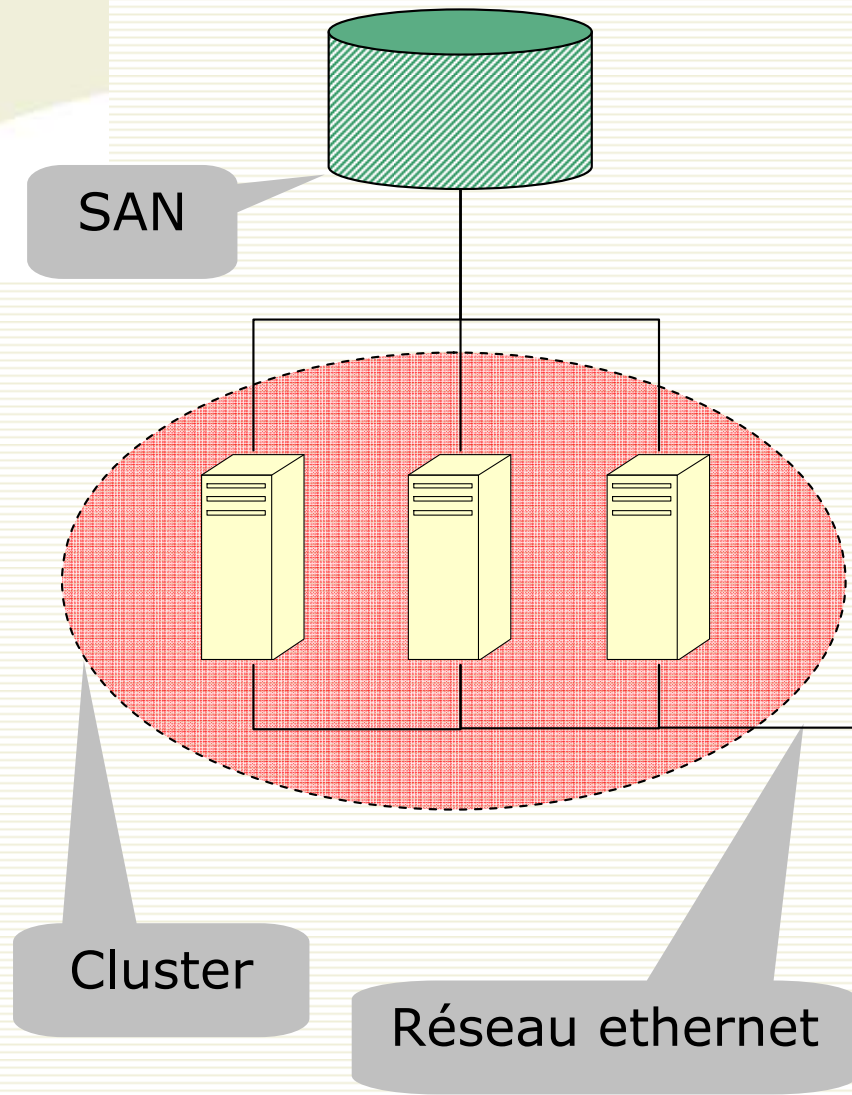
# Plan

---

- Introduction
- Virtualisation par « conteneur » ou « isolateur »
- Linux VServer
- Red Hat Cluster Suite
- Linux VServer en cluster
- Conclusions & perspectives



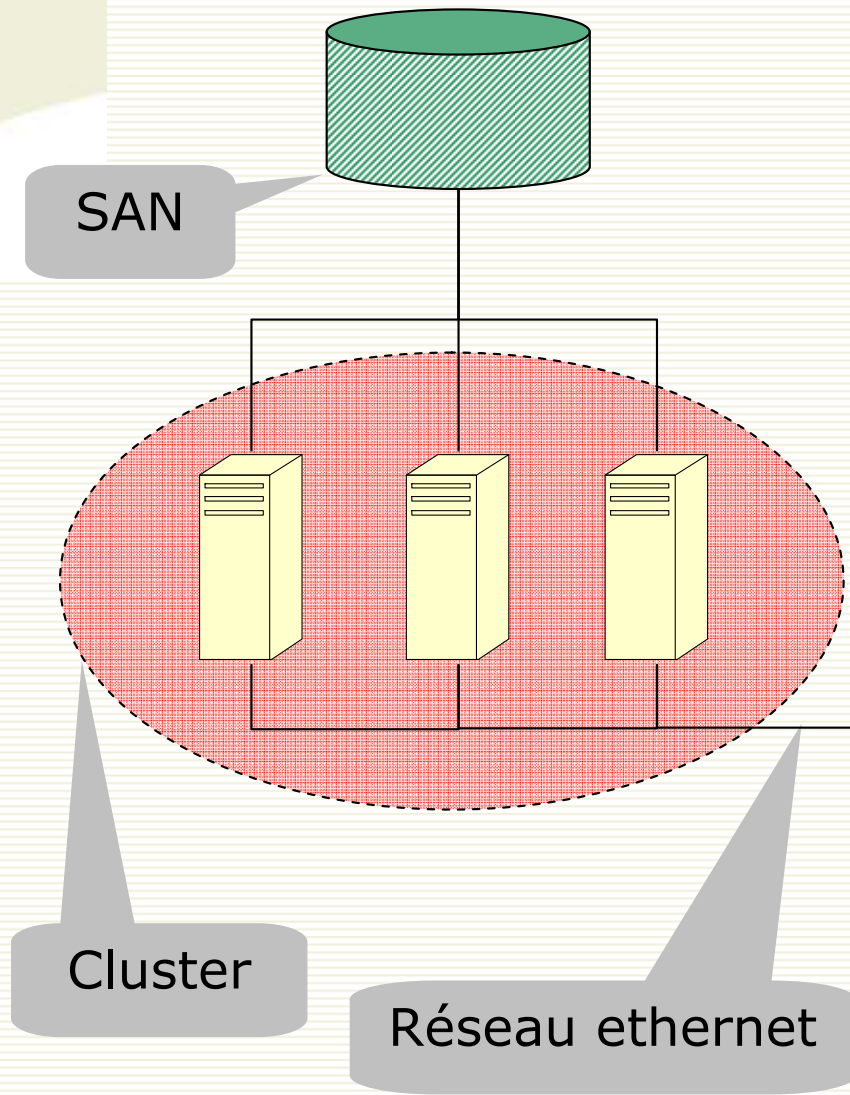
# Cluster et VServer



- En production à l'université de Limoges (janvier 2008)
- 3 serveurs (bi-Xeon EM64T, RAM 8 Go) formant un cluster
- Un disque (du SAN, 700 Go) accessible à tous les membres
- Disque intégré dans un *Volume Group CLVM*



## Cluster et VServer (2)



- `/etc/vservers` : un Logical Volume, GFS2, monté sur tous les hôtes
- Chaque VServer a son Logical Volume, ext3 ou XFS – plus performants que GFS2
- Un VServer = un service du cluster, composé de deux ressources
  - Montage du LV
  - Démarrage du VServer (qui se charge de positionner l'IP)



## Cluster et VServer (3)

---

- ❑ VServers hébergés : courrier des étudiants (14000 usagers), serveurs webs institutionnel et hébergés, FTP, samba, webmail, ENT (portail ESUP, cas, webdav, ...), plate-forme moodle, DNS secondaire, ...
- ❑ Les VServers sont en redhat 4 et 5, en 32 et 64 bits
- ❑ Pas de limitation de ressources (sauf espace disque à travers les LV)
- ❑ Sauvegarde effectuée depuis l'hôte (si le VServer migre d'hôte : incrémentale → totale ☹)



Le cluster

Etat du cluster

Membres du cluster

Services du cluster (un service = un VServer)

clusters

- Cluster List
- Create a New Cluster
- Configure

### Choose a cluster to administer

Cluster Name: Vserver\_2

Restart this cluster

Go

- Status: Quorate
- Total Cluster Votes: 5
- Minimum Required Quorum: 3

#### Nodes

- lame2.unilim.fr
- lame3.unilim.fr
- lame6.unilim.fr
- lame7.unilim.fr
- lame8.unilim.fr

#### Services

- ftp
- checol
- esupmaq
- etu
- ident
- listes
- mediatheque
- scidom
- webdav
- webmail
- ldap-new
- limdns2



homebase cluster storage help quitter

clusters

- Cluster List
- Create a New Cluster
- Configure

vserver\_2

- Nodes
  - Add a Node
  - Configure
  - lame6.unilim.fr
  - lame7.unilim.fr
  - lame8.unilim.fr
  - lame3.unilim.fr
  - lame2.unilim.fr
- Services
- Resources
- Failover Domains
- Shared Fence
- Devices

**Vserver\_2**

Node Name: lame2.unilim.fr Choose a Task... Go

Status: Cluster Member

Services on this Node:

- No cluster services are currently running here

Failover Domain Membership:

- VServers\_lame2

Manage Fencing for this Node Show recent log activity for this node

---

Node Name: lame3.unilim.fr Choose a Task... Go

Status: Cluster Member

Services on this Node:

- ftp

Failover Domain Membership:

- VServers\_lame3

Manage Fencing for this Node Show recent log activity for this node

---

Node Name: lame6.unilim.fr Choose a Task... Go

Status: Cluster Member

Services on this Node:

- checol
- esupmaq
- etu
- ident
- listes
- mediathèque

Failover Domain Membership:

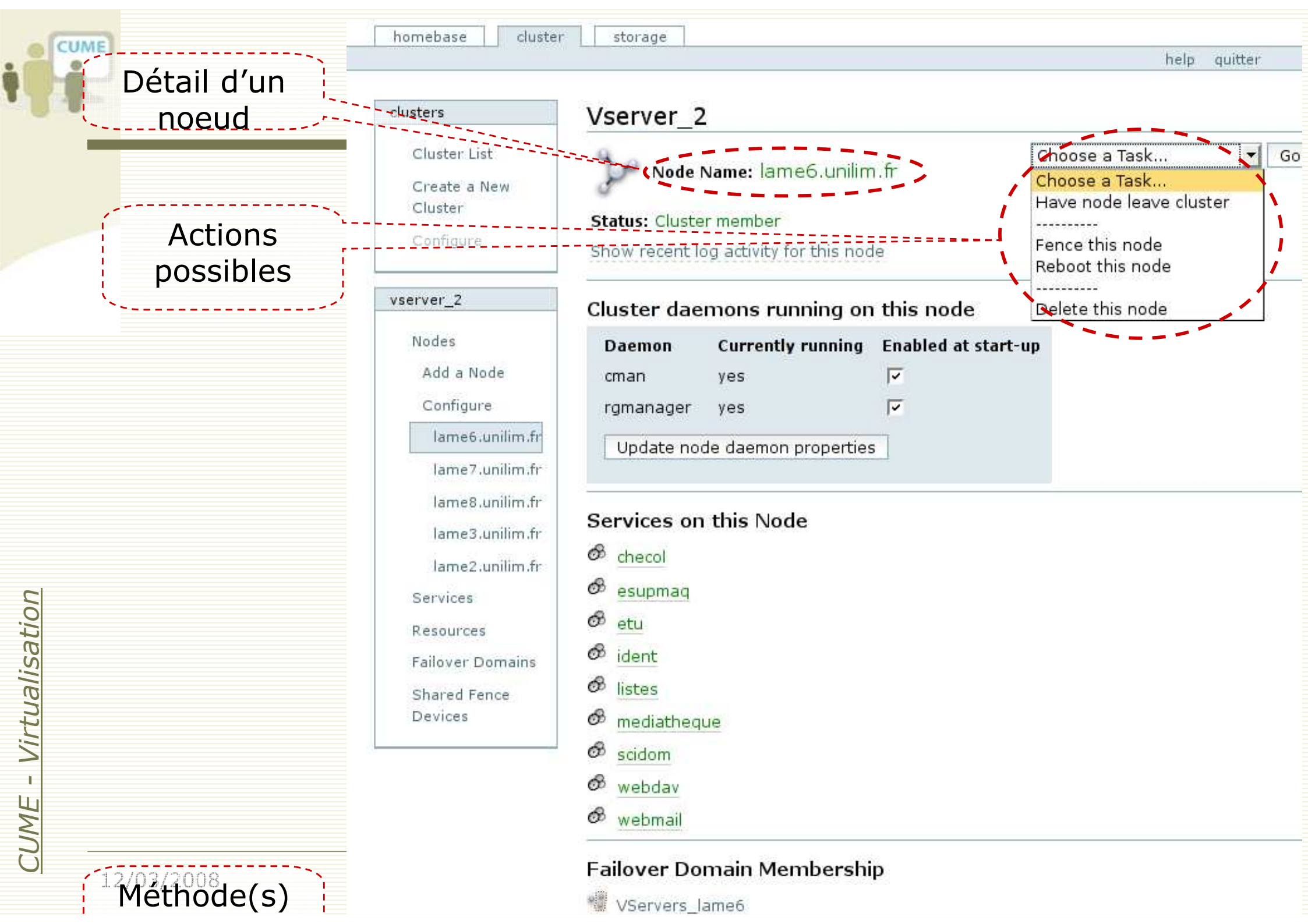
- VServers\_lame6

Cluster

Un noeud

Ses services

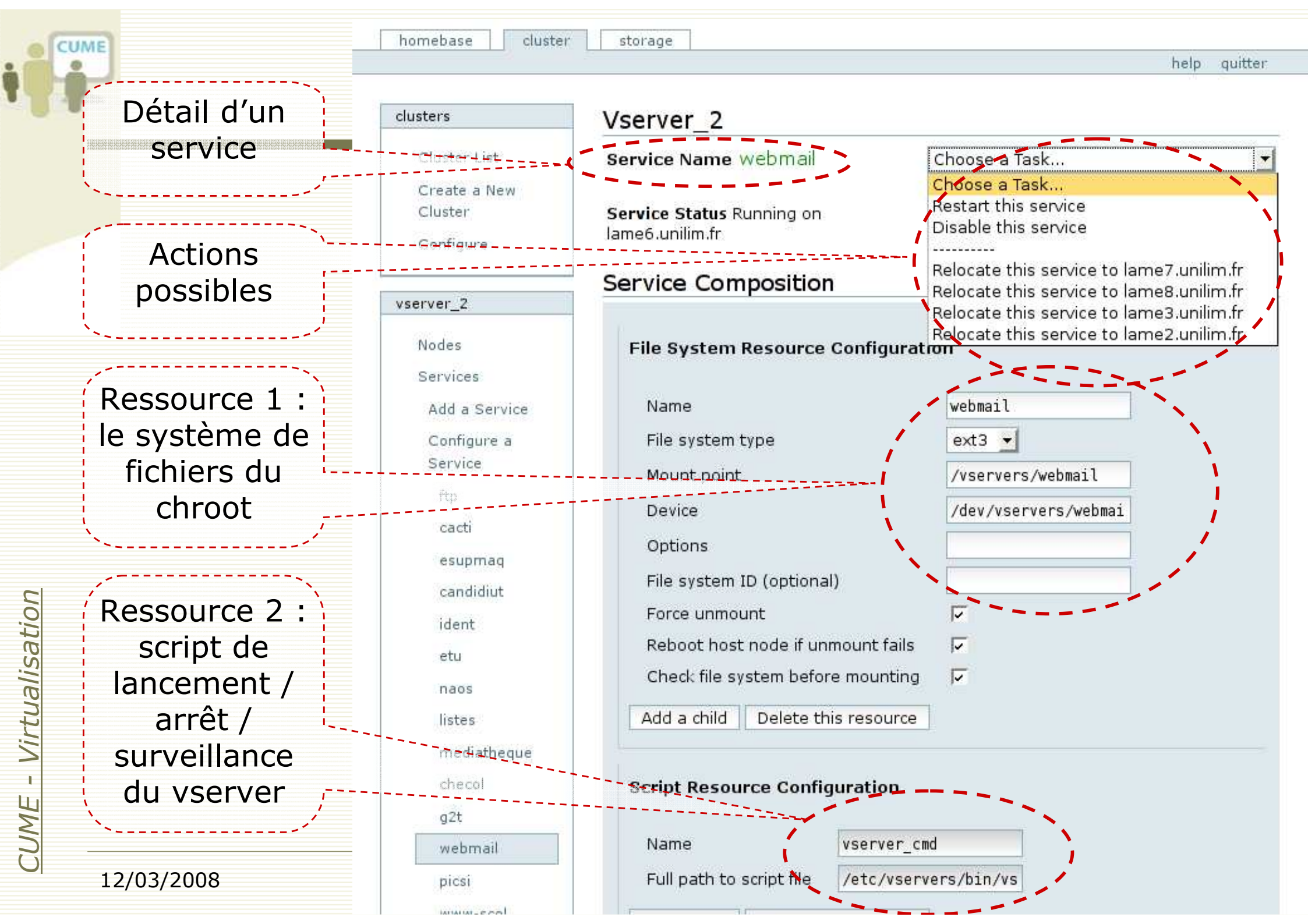
Ses domaines de failover



Détail d'un noeud

Actions possibles

12/02/2008  
Methode(s)



Détail d'un service

Actions possibles

Ressource 1 : le système de fichiers du chroot

Ressource 2 : script de lancement / arrêt / surveillance du vserver

clusters

- Cluster List
- Create a New Cluster
- Configure

vserver\_2

- Nodes
- Services
  - Add a Service
  - Configure a Service
- ftp
- cacti
- esupmaq
- candidiut
- ident
- etu
- naos
- listes
- mediatheque
- checol
- g2t
- webmail
- picis
- www.ecol

Vserver\_2

Service Name webmail

Service Status Running on lame6.unilim.fr

Service Composition

File System Resource Configuration

Name: webmail

File system type: ext3

Mount\_point: /vservers/webmail

Device: /dev/vservers/webmai

Options: [ ]

File system ID (optional): [ ]

Force unmount:

Reboot host node if unmount fails:

Check file system before mounting:

Add a child Delete this resource

Script Resource Configuration

Name: vserver\_cmd

Full path to script file: /etc/vservers/bin/vs

Choose a Task...

- Choose a Task...
- Restart this service
- Disable this service
- Relocate this service to lame7.unilim.fr
- Relocate this service to lame8.unilim.fr
- Relocate this service to lame3.unilim.fr
- Relocate this service to lame2.unilim.fr

**Volume Group vservers**

Graphical View (Uncheck if volumes are too small to select)

Logical Volumes:

Physical Volumes:

Click cylinders to view properties, unselect all to view Volume Group's properties

Le VG

Clic pour choisir un LV

Etat : taille, snapshots, système de fichiers, point de montage ...

**Logical Volume 'web' - /dev/vservers/web**

Logical Volume Name	web
Volume Group Name	vservers
Extent Size	4.0 MB
Size	55.0 GB
Mirrored	false
Attributes	-wi-a-
Clustered	true
Snapshots	
UUID	H8tegz-dhiK-hL4N-4b3n-G9Y7-qAgG-c3HcsD

**Content** Linux Extended FS

Label	web
Block Size	4.0 KB
/etc/fstab Mountpoint	<input type="text"/>
Mountpoint	<input type="text"/>
Mountable	true
Journaling Enabled - ext3	true
State	clean
Use Hashed Binary Trees	true
Number of Blocks	14417920
UUID	1a2e415d-7c35-4ad1-9859-08d5c01f747b



# Plan

---

- Introduction
- Virtualisation par « conteneur » ou « isolateur »
- Linux VServer
- Red Hat cluster suite
- Linux VServer en cluster
- Conclusions & perspectives



## Conclusions & perspectives

---

- ❑ Efficace, léger, robuste
- ❑ Après un apprentissage ...
- ❑ Plate-forme de test : coûteuse ?
- ❑ Espace disque partagé : SAN, ISCSI, (C)NBD, NFS ?
- ❑ Supervision : à intégrer (MRTG, Cacti, OpenNMS, Munin, ...)
- ❑ Outils d'administration pour les VServers ?
- ❑ Equilibrage de charge sur le cluster ?
- ❑ Approche de consolidation à intégrer dans une démarche globale ?



# Merci et questions ?

---

