



CNRS - DSI

**Retour d'expérience sur la mise en œuvre de
My CoRe - Annexes**



Contenu

P. 2 / 18

- 1** Annexe 1 : tests de charges
- 2** Annexe 2 : dimensionnement
- 3** Annexe 3 : pourquoi Scality ?
- 4** Annexe 4 : quelques liens
- 5** Annexe 5 : développements spécifiques
- 6** Annexe 6 : fonctionnement de la sauvegarde/restauration

Contenu

- 1** Annexe 1 : tests de charges
- 2 Annexe 2 : dimensionnement
- 3 Annexe 3 : pourquoi Scality ?
- 4 Annexe 4 : quelques liens
- 5 Annexe 5 : développements spécifiques
- 6 Annexe 6 : fonctionnement de la sauvegarde/restauration

Annexe 1 : tests de charges ^(1/2)

Hypothèses d'usage du service

- ☐ Service potentiellement ouvert à une population de 100000 utilisateurs (agents travaillant dans des unités du CNRS)
- ☐ 50% des utilisateurs vont au maximum réellement utiliser le service
- ☐ Le quota par défaut sera de 10 Go par utilisateur
- ☐ Environ 1000 fichiers par utilisateur
- ☐ Chaque fichier fait 5 Mo
- ☐ 50 mises à jour de fichiers par jour pour un utilisateur
- ☐ Chaque fichier est répliqué vers différentes cibles :
 - ▶ 3 terminaux par utilisateur
 - ▶ 15% des fichiers d'un utilisateur sont partagés vers 5 autres utilisateurs

Annexe 1 : tests de charges ^(2/2)

P. 5 / 11

Postula techniques

- ☐ Hypothèses de tenue de charge sur les serveurs web de type Apache (8 coeurs, 16 Go de RAM) : 530 requêtes simultanées
- ☐ Hypothèses de tenue de charge sur MariaDB (8 coeurs, 16 Go de RAM) :
 - ▶ SELECTs : maximum 3857 par seconde
 - ▶ INSERTs : maximum 22000 par seconde
 - ▶ UPDATEs : maximum 3857 par seconde

Infrastructure sur laquelle les tests ont été passés

- ☐ 2 Virtual Machines (VM) Piranha en actif/passif
- ☐ 2 VM Apache/ownCloud (v6 communautaire) en actif /actif
- ☐ 1 VM MariaDB
- ☐ 1 VM exposant en NFS le stockage présent sur une baie de stockage
- ☐ 1 VM Shinken pour le monitoring
- ☐ 5 VM pour les injecteurs Tsung



Contenu

P. 6 / 18

- 1 Annexe 1 : tests de charges
- 2 Annexe 2 : dimensionnement**
- 3 Annexe 3 : pourquoi Scality ?
- 4 Annexe 4 : quelques liens
- 5 Annexe 5 : développements spécifiques
- 6 Annexe 6 : fonctionnement de la sauvegarde/restauration

Annexe 2 : dimensionnement BDD pour ownCloud

Nombre de nœuds de BDD nécessaires selon les usages

SQL servers	Number of SQL nodes (~ VM) for the estimated SQL load (8 cores/16GB RAM per node)						
	Number of users (N)						
% of active users	1	1000	5000	30000	50000	70000	100000
5,00%	1	1	1	3	5	7	10
10,00%	1	1	1	6	9	12	18
15,00%	1	1	2	8	13	18	25
20,00%	1	1	2	11	18	24	35
30,00%	1	1	3	16	27	38	54
50,00%	1	1	3	18	30	41	59

Annexe 2 : dimensionnement web pour ownCloud

Nombre de nœuds web nécessaires selon les usages

Web servers	Number of web nodes for the estimated load (8 cores/16GB RAM per node)						
	Number of users (N)						
% of active users	1	1000	5000	30000	50000	70000	100000
5,00%	1 serv	1 serv	1 serv	3 serv	5 serv	7 serv	10 serv
10,00%	1 serv	1 serv	1 serv	6 serv	10 serv	14 serv	19 serv
15,00%	1 serv	1 serv	2 serv	9 serv	15 serv	20 serv	29 serv
20,00%	1 serv	1 serv	2 serv	12 serv	19 serv	27 serv	38 serv
30,00%	1 serv	1 serv	3 serv	17 serv	29 serv	40 serv	57 serv
50,00%	1 serv	1 serv	5 serv	29 serv	48 serv	67 serv	95 serv

Annexe 2 : dimensionnement réseau pour ownCloud

Bande passante réseau nécessaire selon les usages

Total DL		Network bandwidth simulation for download [Sync own+Sync share] (global architecture)						
		Number of users (N)						
% of active users		1	1000	5000	30000	50000	70000	100000
5,00%	0 mb/s	13 mb/s	64 mb/s	382 mb/s	637 mb/s	891 mb/s	1 273 mb/s	1 273 mb/s
10,00%	0 mb/s	25 mb/s	127 mb/s	764 mb/s	1 273 mb/s	1 782 mb/s	2 546 mb/s	2 546 mb/s
15,00%	0 mb/s	38 mb/s	191 mb/s	1 146 mb/s	1 910 mb/s	2 674 mb/s	3 819 mb/s	3 819 mb/s
20,00%	0 mb/s	51 mb/s	255 mb/s	1 528 mb/s	2 546 mb/s	3 565 mb/s	5 093 mb/s	5 093 mb/s
30,00%	0 mb/s	76 mb/s	382 mb/s	2 292 mb/s	3 819 mb/s	5 347 mb/s	7 639 mb/s	7 639 mb/s
50,00%	0 mb/s	127 mb/s	637 mb/s	3 819 mb/s	6 366 mb/s	8 912 mb/s	12 731 mb/s	12 731 mb/s
Total UL		Network bandwidth simulation for upload [Sync own+Sync share] (global architecture)						
		Number of users (N)						
% of active users		1	1000	5000	30000	50000	70000	100000
5,00%	0 mb/s	5 mb/s	23 mb/s	139 mb/s	231 mb/s	324 mb/s	463 mb/s	463 mb/s
10,00%	0 mb/s	9 mb/s	46 mb/s	278 mb/s	463 mb/s	648 mb/s	926 mb/s	926 mb/s
15,00%	0 mb/s	14 mb/s	69 mb/s	417 mb/s	694 mb/s	972 mb/s	1 389 mb/s	1 389 mb/s
20,00%	0 mb/s	19 mb/s	93 mb/s	556 mb/s	926 mb/s	1 296 mb/s	1 852 mb/s	1 852 mb/s
30,00%	0 mb/s	28 mb/s	139 mb/s	833 mb/s	1 389 mb/s	1 944 mb/s	2 778 mb/s	2 778 mb/s
50,00%	0 mb/s	46 mb/s	231 mb/s	1 389 mb/s	2 315 mb/s	3 241 mb/s	4 630 mb/s	4 630 mb/s



Contenu

P. 10 / 18

- 1 Annexe 1 : tests de charges
- 2 Annexe 2 : dimensionnement
- 3 Annexe 3 : pourquoi Scality ?**
- 4 Annexe 4 : quelques liens
- 5 Annexe 5 : développements spécifiques
- 6 Annexe 6 : fonctionnement de la sauvegarde/restauration

Annexe 3 : pourquoi Scalify ? ^(1/4)

P. 1

Scalify ? Solution logicielle de stockage distribué

- ☐ Stockage objet "scalable"
- ☐ Protection des données via la technologie ARC

Qu'avons nous aimé ?

- ☐ Compatible avec tout serveur x86
- ☐ Meilleur ratio entre données brutes/données utiles
- ☐ Hautement disponible par conception
- ☐ RAID désactivé sur le matériel

Et ce qui nous paraît moins bien ...

- ☐ Editeur encore jeune
- ☐ Limitations du connecteur sfused (ne gère pas les accès concurrents entre plusieurs nœuds, présente des lenteurs pour des grands nombres de fichiers)



Annexe 3 : pourquoi Scalify ? (2/4)

P. 12 / 19



SCALIFY

Supervisor
Administration

Logged in as root

Hardware | Logout

Local > owncore

Dashboard | Operation | Administration | Provisioning

Status

State RUN
Autojoin Off
Online **12** Nodes
0 RS2
Connectors
Alerts 0

Storage capacity

Objects
Unique objects 1
Average size 0.00 KB
Avg size (unique) 0.00 KB

Unique 0.00 % 0 GB
Stored 0.00 % 0 GB
Used 1.84 % 0 GB
Available 98.16 % 20.00 GB
Total 21.00 GB

Welcome to the provisioning wizard, step one: choose the servers you want to configure.

	Server	Zone	State	# nodes
<input checked="" type="checkbox"/>	192.168.55.21		RUN	6
<input checked="" type="checkbox"/>	192.168.55.22		RUN	6

Warning! We recommend having at least 6 servers.

Configure

[\[Manual provisioning\]](#)



P. 13 / 18

Annexe 3 : pourquoi Scalify ? (3/4)



Administration Interface

Log out

Welcome, Joe Madureira

Supervisor > Local > Anaconda

Dashboard Administration Preferences

Nodes Connectors Actions

Status
Anaconda



Nodes 24
Connectors 12
Alerts 3
Tasks 9

Nodes

Objects 65,134,803
Unique objects 21,728,249
Average size 23.61 KB
Avg size (unique) 171.00 KB

Unique 8.43 TB 44.18 %
Stored 8.43 TB 18.24 %
Used 8.43 TB 27.26 %
Available 8.43 TB 72.74 %
Total 8.43 TB

Nodes

All by tasks by keys

nodea.ring2.devscs.com

Name	Key	Tasks	Objects	CPU	State	Action
nodea_r2_01	CE38E3	0	1809651	2%	RUN	Leave
nodea_r2_02	BE38E3	0	1806412	2%	RUN	Leave
nodea_r2_01	CE38E3	0	1809651	2%	RUN	Leave
nodea_r2_02	BE38E3	0	1806412	2%	RUN	Leave
nodea_r2_01	CE38E3	0	1809651	2%	RUN	Leave
nodea_r2_02	BE38E3	0	1806412	2%	RUN	Leave

Disk Name	Stored	Used	Avail	Total (TB)	Stored/Used
disk1(OK)	0.07	0.10	0.65	0.74	
disk1(OK)	0.07	0.10	0.65	0.74	

nodeb.ring2.devscs.com

Name	Key	Tasks	Objects	CPU	State	Action
nodea_r2_01	CE38E3	0	1809651	2%	RUN	Leave
nodea_r2_01	CE38E3	0	1809651	2%	RUN	Leave

4.1.1 (codename Isildur ; build r33425) Ring by Scalify Copyright 2007-2013 © Scalify - All rights reserved

Annexe 3 : pourquoi Scalify ? (4/4)

P. 14





Contenu

P. 15 / 18

- 1 Annexe 1 : tests de charges
- 2 Annexe 2 : dimensionnement
- 3 Annexe 3 : pourquoi Scality ?
- 4 Annexe 4 : quelques liens**
- 5 Annexe 5 : développements spécifiques
- 6 Annexe 6 : fonctionnement de la sauvegarde/restauration

Annexe 4 : quelques liens

P. 16 / 18

URLs en relation avec My CoRe

- Tests de charge sur ownCloud 6 = https://github.com/CNRS-DSI-Dev/mycore_press/blob/master/CERN-CNRS-meeting-20140513.pdf
- Comparaison entre dimensionnement théorique et métriques réelles = https://github.com/CNRS-DSI-Dev/mycore_press/raw/master/myCore_comparison_estimate-load_real-load_on_ownCloud.pdf
- Conférence JoSy de mai 2014 = https://github.com/CNRS-DSI-Dev/mycore_press/blob/master/CNRS-JoSy-20140519.pdf
- Scality en détail = https://github.com/CNRS-DSI-Dev/mycore_press/blob/master/CNES-CNRS-Scality-20140619.pdf
- Outil de test du CERN = <https://github.com/cernbox/smashbox>
- Besoins métier à l'issue de la phase pilote = https://github.com/CNRS-DSI-Dev/mycore_press/blob/master/besoins/besoins_mycore_apres_pilote.png
- Autres présentations et informations = https://github.com/CNRS-DSI-Dev/mycore_press



Contenu

- 1 Annexe 1 : tests de charges
- 2 Annexe 2 : dimensionnement
- 3 Annexe 3 : pourquoi Scality ?
- 4 Annexe 4 : quelques liens
- 5 Annexe 5 : développements spécifiques**
- 6 Annexe 6 : fonctionnement de la sauvegarde/restauration



Annexe 5 : développements spécifiques ^(1/2)

P. Quelques apps développées et/ou améliorées

- ☐ App de tableau de bord =
<https://github.com/CNRS-DSI-Dev/dashboard>
- ☐ App pour faciliter la gestion des groupes pour les administrateurs =
<https://github.com/CNRS-DSI-Dev/lotsofgroups>
- ☐ App de gestion personnelle de groupes =
https://github.com/CNRS-DSI-Dev/group_custom
- ☐ App pour renforcer la dureté des mots de passe locaux =
https://github.com/CNRS-DSI-Dev/password_policy
- ☐ App de gestion des CGU = <https://github.com/CNRS-DSI-Dev/gtu>
- ☐ App de filtrage applicatif =
<https://github.com/CNRS-DSI-Dev/gatekeeper>
- ☐ App d'intégration à shibboleth pour l'authentification et le provisionning des comptes via Janus ^a =
https://github.com/CNRS-DSI-Dev/user_servervars2

^a. Janus est la solution de gestion des identités du CNRS. Un agent disposant d'un compte dans l'annuaire Labintel du CNRS peut donc s'authentifier dans Janus et accéder à My CoRe

Quelques apps développées et/ou améliorées^(suite)

- ❑ App de gestion des partages = <https://github.com/LydSC/ShareWatcher>
- ❑ App d'actions lors de la création/suppression d'utilisateurs = https://github.com/CNRS-DSI-Dev/user_account_actions
- ❑ App de migration de fichiers entre deux comptes d'un même utilisateur = https://github.com/CNRS-DSI-Dev/user_files_migrate
- ❑ App et scripts pour la sauvegarde et la restauration de fichiers = https://github.com/CNRS-DSI-Dev/mycore_backup_restore_user_files et https://github.com/CNRS-DSI-Dev/user_files_restore
- ❑ Script d'intégration à des listes SYMPA = https://github.com/CNRS-DSI-Dev/mycore_sympa
- ❑ Thème spécifique = <https://github.com/CNRS-DSI-Dev/mycore>
- ❑ Script de build = https://github.com/CNRS-DSI-Dev/mycore_build

Ces développements concernent la partie serveur ownCloud, aucun développement n'a encore été fait sur les divers clients ownCloud. Le code est mis à disposition via Github pour des raisons internes projet mais il serait plus approprié de le faire via SourceSup.



Contenu

- 1 Annexe 1 : tests de charges
- 2 Annexe 2 : dimensionnement
- 3 Annexe 3 : pourquoi Scality ?
- 4 Annexe 4 : quelques liens
- 5 Annexe 5 : développements spécifiques
- 6 Annexe 6 : fonctionnement de la sauvegarde/restauration**

Annexe 6 : fonctionnement de la sauvegarde/restauration ^(1/2)

Sauvegarde des fichiers utilisateur

- ☐ Fait via un script d'intégration entre ownCloud et Tivoli = https://github.com/CNRS-DSI-Dev/mycore_backup_restore_user_files/blob/master/backup_mycore.sh
- ☐ La sauvegarde se fait en tâche de fond sans interaction de la part de l'utilisateur, les données sont sauvegardées tous les jours sur le service Tivoli du centre de calcul de l'IN2P3

Restauration des fichiers utilisateur

- ☐ Fait via un script d'intégration entre ownCloud et Tivoli = https://github.com/CNRS-DSI-Dev/mycore_backup_restore_user_files/blob/master/mycore_restore.sh
- ☐ Et une app pour permettre à l'utilisateur de demander des restaurations = https://github.com/CNRS-DSI-Dev/user_files_restore

Principe de fonctionnement

- ☐ L'utilisateur peut demander une restauration, il a le choix entre 3 versions (J-1, J-6, J-15). Une fois la demande traitée, l'élément restauré écrase la version courante
- ☐ Les demandes de restauration ne sont pas traitées en temps réel par l'app mais ajoutées dans une table de BDD qui sert d'interface avec le script de gestion de restaurations (script planifié en crontab qui traite effectivement les demandes)
- ☐ L'utilisateur peut annuler une demande de restauration non encore "en cours"



Annexe 6 : fonctionnement de la sauvegarde/restauration (2/2)

QbEE Mes Fichiers ▾ myzone

Tous les fichiers

Partagés avec vous

Partagés avec d'autres

Partagés par lien

Nouveau Téléverser

rec115 54 kB Il y a 20 jours

test 337 kB Il y a 23 jours

Poweralarm.png 32 kB Il y a 9 mois

sweethome3d.desktop <1 kB Il y a 30 jours

2 dossiers et 2 fichiers

1 jour avant Restaurer

6 jours avant Restaurer

15 jours avant Restaurer

QbEE Restaurer ▾ myzone

Requêtes de restauration

Informations préalables

- Si vous cherchez à restaurer un fichier que vous avez effacé par erreur, pensez à le chercher dans la corbeille. C'est plus simple et rapide.
- Si vous pouvez également restaurer des éléments en passant par la page par défaut de l'appli.
- Les tâches de restauration sont lancées en tâche de fond. Vous pouvez suivre dans cette page l'évolution de vos demandes et leurs résultats.
- Attention :** les fichiers sauvegardés écrasent les fichiers existants sur le répertoire, s'ils ont le même nom.
- Si vous avez été victime d'un crypto-virus, veuillez vous rapprocher du CSSI de votre unité ou, en son absence, du RSSI de votre délégation régionale du CNRS avant de faire vos demandes de restauration.

Demander une restauration complète

1 jour

Valider

A traiter

/ (dossier) ; (- 1)
/rec115 (dossier) ; (- 15)Annuler
Annuler

En cours

(dossier) /test

Traité

(dossier) /rec115
(dossier) /rec115ttt

Voir plus