
	Titre du document : Compte rendu d'étude
	Référence : L 1.6.1
	Version du 24/09/2011

# Suivi des accès et statistiques


<b>Livrable du au titre du projet</b>	COCLICO
<b>Lot</b>	1
<b>Tache</b>	0.6
<b>Livrable</b>	1.6.1

<b>Rédacteur(s)</b>	<b>Vérificateur(s)</b>	<b>Approbateur(s)</b>
N Guerin, Bull	C Bayle	


	Titre du document : Compte rendu d'étude
	Référence : L 1.6.1
	Version du 24/09/2011

## Table des matières

I. Forges: Suivi des accès et statistiques .....	4
A. Classement des outils en catégories .....	4
B. Analyse de l'utilisation des forges .....	4
C. Analyse des projets .....	4
D. Solutions externes .....	5
E. Supervision de l'infrastructure.....	6
F. Supervision Métier.....	7
i. Informations globales.....	7
ii. Informations contextuelles.....	7
II. Solutions proposées.....	9
A. Collecte de données à l'aide de sondes dédiées.....	9
B. Utilisation de Munin .....	9
i. Mise en oeuvre de plugins Munin spécifiques.....	9
a) Création de la sonde.....	9
b) Configuration.....	10
c) Activation du plugin.....	11
ii. Sondes Munin dédiées aux forges .....	11
a) Nombre total de commits Subversion .....	11
b) Nombre d'utilisateurs: total et actifs .....	11
c) Nombre de projets: total et actifs .....	11
d) Nombre total d'artifacts créés .....	11
iii. Exemple.....	12
C. Plugin de calcul de statistiques depuis la base de données.....	14
i. Fréquence.....	14
a) Export des données temporelles.....	15

	Titre du document : Compte rendu d'étude
	Référence : L 1.6.1
	Version du 24/09/2011

ii.Utilisation du disque.....	15
a)Utilisation globale par service.....	15
b)Utilisation détaillée par projet.....	17

	Titre du document : Compte rendu d'étude
	Référence : L 1.6.1
	Version du 24/09/2011

# I. Forges: Suivi des accès et statistiques

## A. Classement des outils en catégories

Les différents outils de métrologies identifiés comme utilisés dans la mesure d'activité des forges peuvent être classifiés en catégories différentes :

- Outils système / Analyse des logs
  - Awstat
  - Munin
- Outils système / Temps réel
  - Basés sur SNMP
  - Nagios
  - Zabbix
- Outils de développement
  - Sonar
- Forges
  - Page des statistiques
  - Plugin statistique de Codendi

## B. Analyse de l'utilisation des forges


Les administrateurs de forges peuvent être intéressés par l'utilisation de la plateforme, que ce soit:

- d'un point de vue système:
  - Utilisation des ressources: CPU, disque
  - Temps de réponse et performances
  - Disponibilité de la plate-forme
- du point de vue de l'usage:
  - Qui sont les principaux utilisateurs
  - Quels sont les projets morts ou en sommeil
  - Quels sont les services les plus (ou les moins) utilisés
  - ...

## C. Analyse des projets

Chaque utilisateur, et en particulier les administrateurs et membres d'un projet, peuvent être intéressés par une analyse de ce qui se passe dans le projet.

On pourrait fournir des informations en temps réel telles que:

	Titre du document : Compte rendu d'étude
	Référence : L 1.6.1
	Version du 24/09/2011

- les utilisateurs les plus actifs dans le projet
- les zones les plus actives dans le wiki, le code source, les listes de diffusion et autres canaux de communication
- les demandes en attente
- des informations de reporting sur l'usage

Cela fournirait une information sur ce qui se passe dans le projet, à qui demander de l'aide, et où aider. C'est d'autant plus utile pour les gros projets ainsi que pour les projets distribués géographiquement.

## D. Solutions externes


Il existe aujourd'hui des outils qui fournissent des métriques sur des projets logiciels en utilisant un simple accès au dépôt de code source (généralement Subversion) du projet. Le type d'information retourné concerne donc uniquement le code est les contributeurs au code. Néanmoins, ce type de métrique est intéressant pour les contributeurs comme pour les utilisateurs d'un projet.

La plate-forme la plus connue aujourd'hui est Ohloh, mais il existe aussi des alternatives open-source (voir par exemple Melchiades).

Exemples:

- Le projet FusionForge analysé par Melchiades
- Le projet Codendi analysé par Ohloh

Les outils d'analyse de code ne rentrent pas directement dans la cible de cette tâche, mais sont à considérer si l'on souhaite approfondir le travail sur les statistiques.


	Titre du document : Compte rendu d'étude
	Référence : L 1.6.1
	Version du 24/09/2011

## E. Supervision de l'infrastructure

La supervision de l'infrastructure comprend les éléments de base nécessaires au suivi du/des système(s) mis en place pour le fonctionnement de la plate-forme.

Il est nécessaire d'avoir des indicateurs sur les éléments suivants :

Indicateur	Représentation	Definition
CPU – global	N Courbes (de couleurs différentes) exprimées en %	Pour chaque processus de la forge une courbe est affichée suivant les valeurs de la charge processeur.
Empreinte mémoire	N Courbes (de couleurs différentes) exprimées en %	Pour chaque processus de la forge une courbe est affichée suivant les valeurs de l'empreinte mémoire
Log-activité	N Courbes (de couleurs différentes) représentant les hits (requêtes)	Suivre le foisonnement des requêtes à travers les fichiers de logs
Log-activation	NOP	Activation à chaud du niveau de log et des logger spécifiques.

	Titre du document : Compte rendu d'étude
	Référence : L 1.6.1
	Version du 24/09/2011

## F. Supervision Métier


L'ensemble des indicateurs métiers (liés au domaine de la forge) est défini ci-dessous.

### i. Informations globales


Indicateur	Représentation	Definition
Nb-projet-status	Numérique	Nombre de projets courants en activité (avec le status « activated »)
Service-activité	Histogramme ou nuage de mots (la taille représente la valeur) représentant les services de la forge	Métrique ordonnant les services suivant leurs activités (requête)
User-globale	Nombre d'utilisateur de la forge	Nombre d'utilisateur de la forge

### ii. Informations contextuelles

Indicateur	Représentation	Definition
Service-détail	Ensemble de propriétés	Pour chaque service définir une représentation détaillée de son activité (exemple nombre de commit,update pour surbversion)
Service-détail-projet	Ensemble de propriétés	définir une représentation détaillée de son activité (exemple nombre de commit,update pour surbversion) pour un projet spécifique
Projet-détail	Ensemble de propriétés d'un projet	<ul style="list-style-type: none"> <li>- propriétaire</li> <li>- date de création</li> <li>- nombre d'utilisateur</li> <li>- activité (nombre de sessions ouvertes sur celui-ci)</li> <li>- volumétrie estimée prise par le projet</li> </ul>
User-détail	Ensemble de propriétés d'un	- Ses projets (ordonnés

	Titre du document : Compte rendu d'étude
	Référence : L 1.6.1
	Version du 24/09/2011

	user	<p>suivant le nombre de connexion ) où se trouve l'utilisateur.</p> <ul style="list-style-type: none"> <li>- Les services que l'utilisateur utilise le plus souvent.</li> </ul>
--	------	---

	Titre du document : Compte rendu d'étude
	Référence : L 1.6.1
	Version du 24/09/2011

## II. Solutions proposées

### A. Collecte de données à l'aide de sondes dédiées

Une pratique courante dans la collecte de données systèmes est l'utilisation du format [RRDTool <http://fr.wikipedia.org/wiki/RRDTool> RRDTool]. RRDTool est utilisé par de nombreux outils open source, tels que Cacti, Munin, ou Nagios, pour la sauvegarde et le tracé de graphiques de données chronologiques. Cet outil a été créé pour superviser des données serveur, telles la bande passante et la température d'un processeur. Le principal avantage d'une base RRD est sa taille fixe.

Dans le cas d'une forge, il est possible d'utiliser des sondes pré-existantes pour surveiller les paramètres du serveur: espace disque total, utilisation CPU, latence réseau... Néanmoins il est également possible de créer des sondes dédiées qui vont générer à intervalle régulier des informations concernant l'utilisation de la plateforme. Les informations collectables sont nombreuses: données de commit, utilisation des trackers, utilisation de l'espace disque par les projets, éditions wiki, envois de messages aux listes de diffusion, etc.

Ce genre d'outil a déjà été proposé pour les forges; en particulier, Cacti pour Fusion Forge et Munin pour Codendi.

Il semble donc intéressant de proposer des sondes ou plugins pouvant fournir ce genre de données, qui soient facilement déployables sur des forges diverses. Le type de reporting pourrait être unifié, même si les outils de backend peuvent être différents (par exemple Mailman vs. Sympa pour la gestion des listes de diffusion, wikis différents, etc.).

### B. Utilisation de Munin

Codendi propose déjà une intégration avec l'outil Munin. Néanmoins, c'est une installation standard de Munin, et aucune sonde spécifique n'est fournie: c'est aux administrateurs de la forge de créer les leurs si besoin.


Nous allons décrire comment créer des sondes spécifiques, et fournir un certain nombre d'indicateurs utiles.

#### i. Mise en oeuvre de plugins Munin spécifiques

Dans ce paragraphe, nous allons décrire comment créer une sonde Munin qui renvoie à chaque appel le nombre total de commits SVN réalisés dans la forge.

##### a) Création de la sonde

Les sondes Munin sont stockées dans un endroit comment. Sur RedHat/CentOS5, c'est dans `/usr/share/munin/plugins`.

	Titre du document : Compte rendu d'étude
	Référence : L 1.6.1
	Version du 24/09/2011

Pour commencer, il faut donc créer un fichier: `/usr/share/munin/plugins/svnrev`

```
#!/usr/bin/perl -w

sub autoconf {
    print "yes\n";
    exit 0;
}

sub config {
    print "graph_title Total number of Subversion Revisions\n";
    print "graph_vlabel num rev\n";
    print "graph_info This graph shows how many SVN revisions exist on all
repositories\n";
    print "graph_category forge\n";
    print "forge_svnrev.label Subversion Revisions\n";
}

if ($ARGV[0]) {
    my $arg = $ARGV[0];
    my %funcs = (
        config => \&config,
        autoconf => \&autoconf,
    );

    if (exists $funcs{$arg}) {
        $funcs{$arg}->();
    } else {
        die "Unknown argument '$arg'\n";
    }
} else {
    my @revlist= `find /svnroot/ -maxdepth 1 -mindepth 1 -name '*' -type d
-exec /usr/bin/svnlook youngest {} \`; `;
    my $total=0;
    foreach (@revlist) {
        $total = $total + $_;
    }
    print "forge_svnrev.value $total\n";
}
```

Un appel à ce script retourne la ligne suivante:

```
forge_svnrev.value 12345
```

Lorsque le plugin sera mis en place, cette donnée sera stockée dans une base RRD avec la date et l'heure. Munin utilisera ensuite cette information pour produire ses graphiques.

## b) Configuration

Pour calculer les nombres de commits SVN, le script doit pouvoir lire dans tous les dépôts Subversion hébergés par la forge. L'utilisateur "munin" n'a pas les droits suffisants. On doit donc configurer munin-node pour exécuter le script en tant qu'un utilisateur autorisé

Dans `/etc/munin/plugin-conf.d/munin-node` :



Titre du document : Compte rendu d'étude

Référence : L 1.6.1

Version du 24/09/2011

```
[svnrev]
user codendiadm
```

Et dans `/etc/munin/plugins.conf`

```
[svnrev]
user codendiadm
```

### c)Activation du plugin

Il faut que le plugin soit présent dans `/etc/munin/plugins` pour être pris en compte. On crée donc un lien vers le fichier depuis ce répertoire:

```
cd /etc/munin/plugins
ln -s /usr/share/munin/plugins/svnrev
```

Enfin, il faut redémarrer le démon "munin-node":

```
service munin-node restart
```

## ii.Sondes Munin dédiées aux forges

### a)Nombre total de commits Subversion

Voir exemple ci-dessus.

Cet indicateur donne une bonne idée des activités de type développement qui ont lieu au jour le jour dans la forge.

Le code source est accessible sur [codendi.org](http://codendi.org): `svnrev`.

### b)Nombre d'utilisateurs: total et actifs

Ce script permet de suivre l'évolution du nombre total d'utilisateurs enregistrés dans la forge, ainsi que le nombre d'utilisateurs actifs (non suspendus).

Le code source est accessible sur [codendi.org](http://codendi.org): `forge_users`.


### c)Nombre de projets: total et actifs

Ce script permet de suivre l'évolution du nombre total de projets créés dans la forge, ainsi que le nombre de projets actifs (approuvés, et non supprimés).

Le code source est accessible sur [codendi.org](http://codendi.org): `forge_projects`.

### d)Nombre total d'artifacts créés

L'idée ici est de compter le nombre total d'artifacts créés dans la forge. Le script correspondant

	Titre du document : Compte rendu d'étude
	Référence : L 1.6.1
	Version du 24/09/2011

nécessite un accès à la base de données de la forge afin d'y effectuer des requêtes.

Le code source est accessible sur [codendi.org](http://codendi.org): `forge_art`.

Un autre plugin dérivé de celui-ci permet de calculer le nombre moyen d'artifact par tracker, ainsi que le nombre moyen d'artifact par tracker non vide: `forge_artifact_per_tracker`.

On le voit, la souplesse des sondes Munin permet facilement de suivre les indicateurs pertinents pour l'administrateur de la forge.

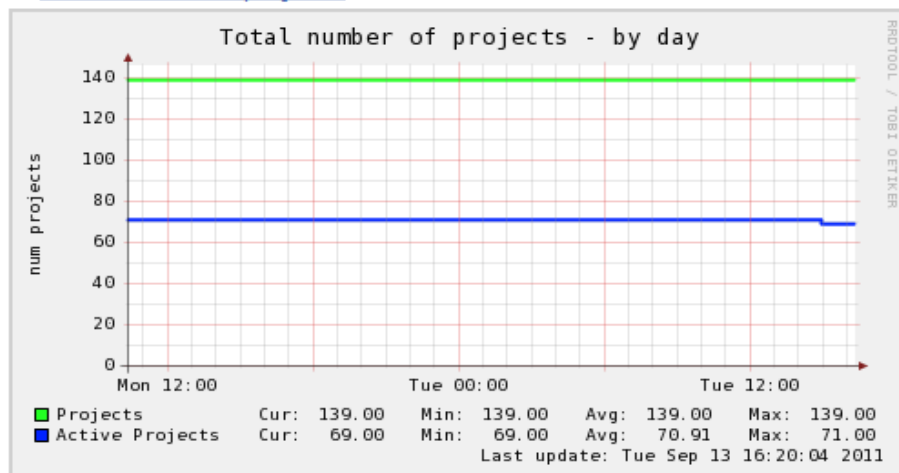
### iii.Exemple

Voici une capture d'écran de Munin contenant des indicateurs liés à une forge Codendi.

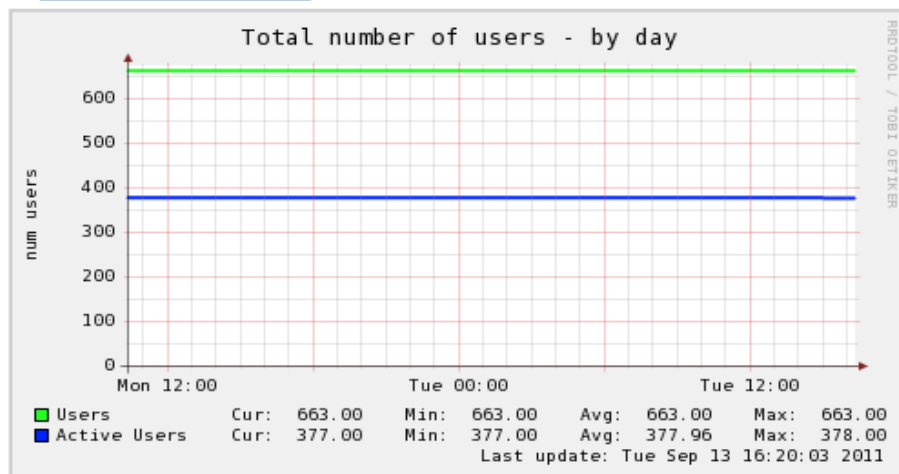
On le voit, cette forge (de test) ne montre pas une très forte activité!



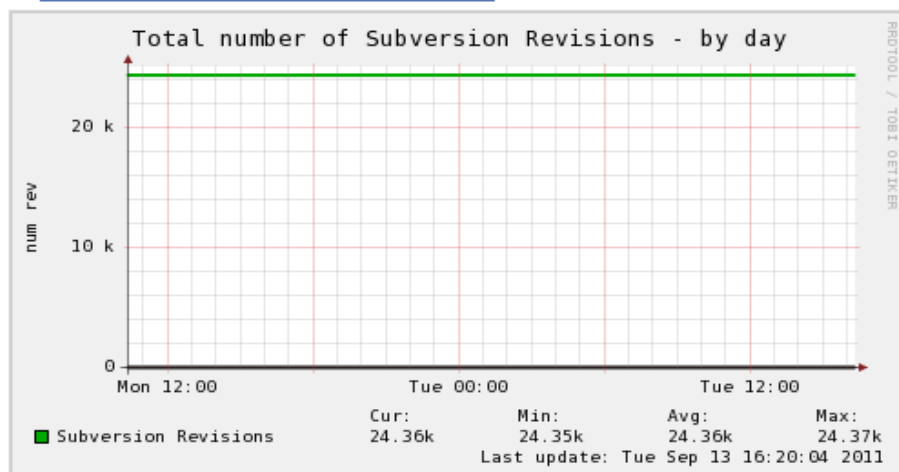
:: [Total number of projects](#)




:: [Total number of users](#)



:: [Total number of Subversion Revisions](#)



	Titre du document : Compte rendu d'étude
	Référence : L 1.6.1
	Version du 24/09/2011

## C.Plugin de calcul de statistiques depuis la base de données

### i.Fréquence

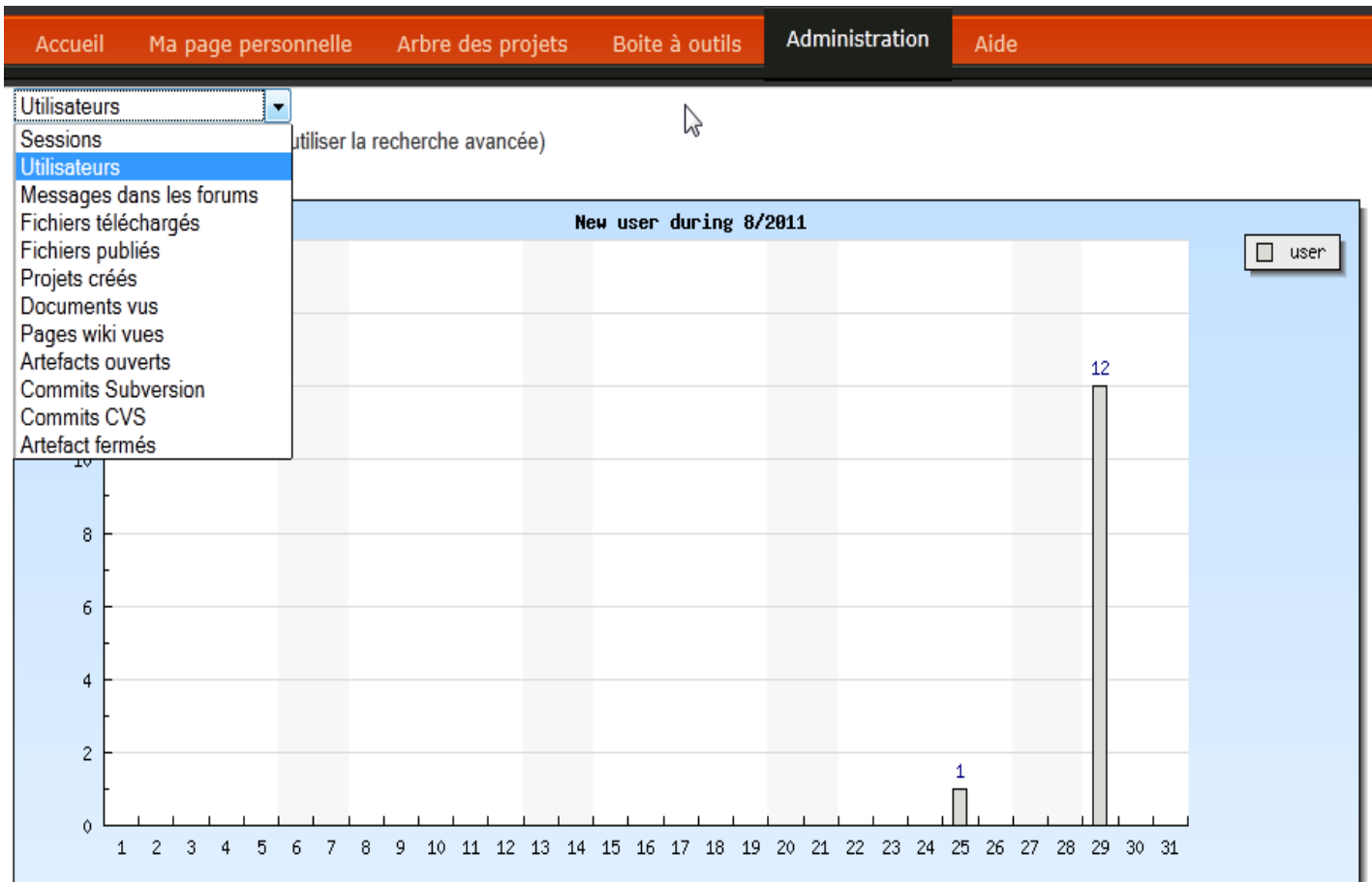
On l'a vu, l'utilisation de sondes RRD permet de stocker des données temporelles dans un format standard, qui peut ensuite être exploité pour créer des graphiques. Pour certains indicateurs, il est néanmoins possible de retrouver l'ensemble des données nécessaires dans la base de données. Dans ce cas, les graphiques peuvent être calculés en temps réel sans nécessiter le stockage de données temporelles.

C'est pour ce type d'indicateurs qu'un plugin de statistiques a été créé dans Codendi. Il calcule à la demande des statistiques pertinentes sur l'usage de la forge.

Voici le type d'indicateurs fournis:

- nombre de sessions ouvertes
- nombre de messages dans les forums
- fichiers publiés
- fichiers téléchargés,
- documents ouverts
- pages wiki affichées
- etc.

Ce plugin s'appuie sur la bibliothèque graphique JPGraph pour générer des graphiques:



### a) Export des données temporelles


Le plugin permet un export au format CSV de certaines statistiques (utilisateurs enregistrés, projets) depuis le démarrage de la forge. Cela permet à l'administrateur de choisir la mise en forme de ces informations avec des outils tiers (tableurs par exemple).

## ii. Utilisation du disque


Le plugin de statistique fourni également une analyse de l'utilisation du disque, globale, mais aussi par projet, utilisateur, et service, tout en permettant de visualiser l'évolution au cours du temps.

Les données sont collectées chaque jour par un événement système qui mesure l'espace disque occupé précisément par les services et projet. Cela permet d'avoir un suivi détaillé (par user/par service/par projet) de l'utilisation de l'espace de stockage, ce qui ne serait pas possible avec une sonde Munin globale.

### a) Utilisation globale par service

	Titre du document : Compte rendu d'étude
	Référence : L 1.6.1
	Version du 24/09/2011

Une première vue globale est disponible. Elle permet aux administrateurs de la forge de surveiller l'espace pris respectivement par chaque service (Subversion, documents, fichiers).


Connecté: guerin | [Se déconnecter](#) | [Enregistrer un projet](#) | [Cette page en favori](#)

Projets logiciel

Tous les mots

Accueil
Ma page personnelle
Arbre des projets
Boite à outils
Administration
Aide

## Disk usage

- Service/Projects
- Users
- Services
- Top users
- Top projects
- One user details
- One project details

[<<Back to all statistics](#)

## Utilisation par service

Date	Users	MySQL	Codendi Logs	Backup	/	/boot
2011-09-01	18.6 MiB	32.7 MiB	316 KiB	48.0 KiB	2.68 GiB	26.0 MiB

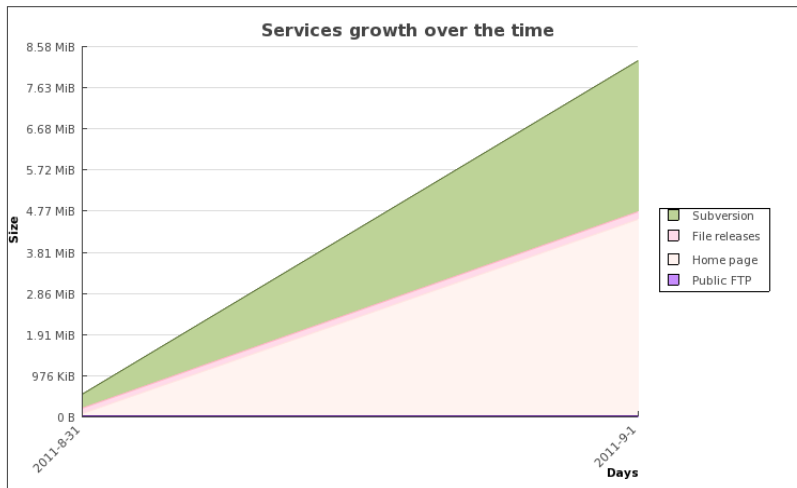
## Croissance des services

Services
Group by
Start date
End date
Options


Subversion  
CVS  
File releases  
Public FTP  
Home page  
Wiki

Relative Y-axis (depend of data set values)

*il y a 8 jours*      *il y a 17 heures*



Service	Start size	End size	Size evolution	Rate evolution
Subversion	340 KiB	6.71 MiB	6.38 MiB	1921.18 %
CVS	0 B	0 B	0 B	-
File releases	128 KiB	192 KiB	64.0 KiB	50.00 %
Public FTP	16.0 KiB	16.0 KiB	0 B	-
Home page	64.0 KiB	9.08 MiB	9.02 MiB	14425.00 %
Wiki	0 B	0 B	0 B	-
Mailman	0 B	0 B	0 B	-
SVN/Webdav	0 B	0 B	0 B	-
Docman	0 B	0 B	0 B	-
Git	0 B	0 B	0 B	-
ForumML	0 B	0 B	0 B	-
<b>Total size</b>	<b>548 KiB</b>	<b>16.0 MiB</b>	<b>15.5 MiB</b>	<b>2888.32 %</b>

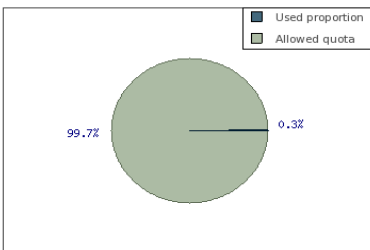
	Titre du document : Compte rendu d'étude
	Référence : L 1.6.1
	Version du 24/09/2011

## b) Utilisation détaillée par projet

Les informations d'utilisation de l'espace disque d'un projet sont disponible aux administrateurs de la forge, mais aussi aux administrateurs du projet concerné.

### Utilisation disque

 Ce projet utilise 16.1 MiB de 5GiB alloué.



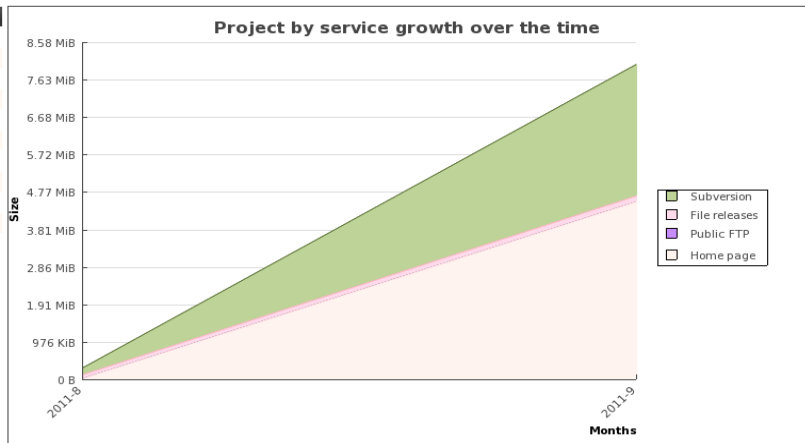
### Utilisation du disque pour la dernière année

[Voir les statistiques pour les 3 derniers mois](#)

- Services
- Subversion
  - CVS
  - File releases
  - Public FTP
  - Home page
  - Wiki

Valider

Service	Start size	End size	Size evolution	Rate evolution
Subversion	176 KiB	6.55 MiB	6.38 MiB	3711.36 %
CVS	0 B	0 B	0 B	-
File releases	116 KiB	180 KiB	64.0 KiB	55.17 %
Public FTP	4.00 KiB	4.00 KiB	0 B	-
Home page	16.0 KiB	9.03 MiB	9.02 MiB	57700.00 %
Wiki	0 B	0 B	0 B	-
Mailman	0 B	0 B	0 B	-
Docman	0 B	0 B	0 B	-
Git	0 B	0 B	0 B	-
ForumML	0 B	0 B	0 B	-
<b>Total size</b>	<b>312 KiB</b>	<b>15.8 MiB</b>	<b>15.5 MiB</b>	<b>5073.08 %</b>



Cette information peut être publiée sur le tableau de bord du projet:

Subversion
Fichiers
Suivi
Intégration Continue

 **Personnaliser**

▼ **Statistiques Du Projet** ✕

Ce projet utilise **16.1 MiB** de **5GiB** alloué.

Il n'y a pas de données a afficher