

# Automatisation de l'administration de 700 serveurs avec Chef

Alain Heinrich  
Christophe Palanché  
12/12/2013

*jres*  
Montpellier 2013

échanges  
aspirationalidad  
insertion  
*perspectives*  
mutualisation  
reussite  
ouverture  
fondation  
CHEMISTRY  
evolution  
biology  
 $E = mc^2$   
RECHERCHE  
SYNERGIES  
CONFÉRENCES  
pi  
TECHNOLOGY  
doctorat  
savoirs  
ENSEIGNEMENT SUPÉRIEUR  
biotechnologies  
axiome  
mécanique  
management  
droit  
excellence  
savoirs  
wissenschaft  
bibliothèques  
médecine  
tesis  
biologie  
gravitation  
idéaux  
connaissances  
langage  
MUTE INTERNATIONAL  
solution  
HEURISTIQUE  
partenariats  
HISTOIRE  
physique  
mécanique quantique  
insertion  
PLURIDISCIPLINARITÉ

- 700 serveurs administrés par la Direction Informatique
- Chiffre en augmentation constante :
  - Augmentation des services proposés aux utilisateurs
  - Virtualisation
- Mise en place de solutions de gestion de configuration et d'automatisation

- Critères de sélection :
  - Maturité, documentation, communauté, etc.
- Maquettage :
  - CFEngine, Puppet et Chef
  - Intégration de plates-formes existantes
- Solution retenue : Chef
  - <http://www.opscode.com/chef>

- Chef-server
  - Centralise les configurations
- Chef-client
  - Agent installé sur les nœuds
- Cookbooks :
  - Éléments de base
  - Contiennent des recettes, attributs et fichiers à transférer
  - Possibilité de gérer des versions

- Recettes :
  - Description des actions à effectuer sur un nœud
  - DSL Ruby
  - Utilisation de ressources
- Exemple :

```
package "nginx" do
  action :install
end

service "nginx" do
  action [:enable, :start]
end
```

- Attributs :
  - Variables manuelles
  - Variables automatiques remontées par Ohai (OS, IP, etc.)
- Fichiers et templates :
  - Fichiers bruts
  - Templates en eRuby

# Présentation de Chef

```
template "/etc/nginx/sites-available/#{sogo_conf['server_name']}" do
  source "nginx-sogo.conf.erb"
  variables(
    :server_name => sogo_conf['server_name']
  )
  owner "root"
  group "root"
  mode "0644"
  notifies :restart, "service[nginx]"
end
```

```
server {
  listen <%= node[:ipaddress] %>:443;
  server_name <%= @server_name %>;
  ...
}
```

- Databags :
  - Stockage centralisé d'informations au format JSON
  - Chiffrement possible
- Rôles :
  - Factorisation de configuration pour un groupe de nœuds
- Environnement :
  - Contraintes sur les versions de cookbooks
- Indexation

- Knife :
  - Outil d'administration principal en ligne de commande
- Exemples :
  - Déployer chef-client sur des nouveaux nœuds :

```
knife bootstrap Serveur
```
  - Gérer les cookbooks :

```
knife cookbook upload Cookbook
```
  - Créer ou modifier les databags, les rôles, les attributs et les run-list :

```
knife data bag edit Bag Item
```
  - Exécuter des commandes sur des nœuds via ssh :

```
knife ssh node:serveur "chef-client"
```

- Plusieurs plates-formes déployées avec Chef
  - Déploiements en masse de services simples :
    - Hôtes de virtualisation
    - Services de base
  - Déploiements complexes :
    - SOGo
    - Messagerie
  - Etc.

- Contraintes :
  - Une nouvelle façon de travailler
  - Un nouveau langage de haut niveau
  - Besoin de formation des équipes
- Bénéfices :
  - Méthode de déploiement unique
  - Déploiement rapide de nouveaux serveurs
  - Gestion centralisée
  - Garantie de résultat

- Bilan positif
- Objectif à court terme :
  - Chef sur tous les nouveaux serveurs
- Perspectives :
  - Travail avec les équipes de développement (intégration continue)
  - Mise en place d'une solution d'orchestration