

Le contour

- Motivations
- Le programmation de l'infrastructure
- Exemples
- Les possibilités de recherche
- Motivations
- Infrastructure as Code
- Exemples
- Research Opportunities

2013-06-19

[1]

Aujourd'hui, je vais traiter le sujet de l'infrastructure comme code et montrer comment nous pouvons utiliser des outils comme Chef pour gérer nos systèmes.

Mon sujet et quelque chose qui est très pertinent étant donné que nos infrastructure devient de plus en plus virtualisée et élastique.

L'ingénierie dans les nuages

Engineering your cloud infrastructure using CHEF

Dr. Andrew Forward

forward@gmail.com / @a4word

Le 19 juin, 2013

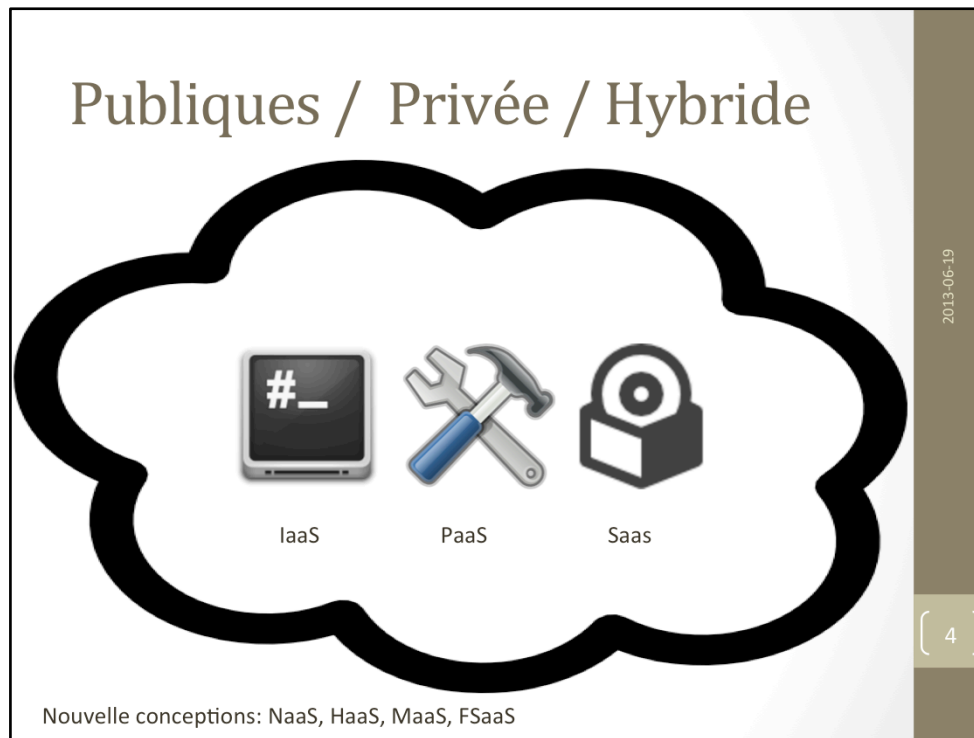
2013-06-19

[2]

En d'autres termes, c'est l'ingénierie dans les nuages.



C'est quoi un nuage en informatique: un nuage se definit par deux choses en principaux:



D'abord, c'est une offre de service virtuel qu'on héberge n'importe où, n'importe quand (e.g. Amazon EC2, ou Google Apps)
Ensuite, la capacité des ressources est dynamique, et on peut l'augmenter ou la diminuer uniquement sur demande

Jusqu'à présent, on connaît 3 modèles d'usage commun

- a - le IAAS - infrastructures comme un service: un service de serveurs virtuels qu'on peut configurer sur demande.
- b - le PAAS - plateforme comme un service: de bases de données, de logiciels, et des API (interface de programmation), (e.g. Heroku)
- c - le SAAS - logiciel comme un service, (e.g. Salesforce, Google Apps)
- d - le NAAS - réseau comme un service, une nouvelle conception qui considère la connectivité du réseau (e.g. VPN or bandwidth-on-demand)
- E – le HAAS – matériel comme un service
- F – le MaaS – métal as a service
- G – FSaaS – scénarios de panne as a service

III En plus, les modèles peuvent être déployés de 3 façons

- a – Nuage privé

Services des nuages

IAAS	PAAS	SAAS
Amazon EC2	AWS Elastic Beanstalk	Petrosoft
Azure Services Platform	Windows Azure Cloud Services	Microsoft Office 365
Google Compute Engine	Google App Engine	Google Apps
Rackspace	Cloud Foundry	Marketo
Digital Ocean	Heroku	Salesforce

2013-06-19

[5]

Voici plusieurs exemples des types de services disponible par les compagnie comme Google, Microsoft et Amazon.

Dans cette exposé nous allons traité uniquement le IAAS ainsi que l'application de la programmation de l'infrastructure en utilisant le logiciel CHEF.

http://en.wikipedia.org/wiki/Infrastructure_as_a_service

Amazon EC2, AirVM, Azure Services Platform, DynDNS, Google Compute Engine, HP Cloud, iland, Joyent, LeaseWeb, Linode, NaviSite, Oracle Infrastructure as a Service, Rackspace, ReadySpace Cloud Services, ReliaCloud, SAVVIS, SingleHop, and Terremark

http://en.wikipedia.org/wiki/Cloud_computing

AWS Elastic Beanstalk, Cloud Foundry, Heroku, Force.com, EngineYard, Mendix, OpenShift, Google App Engine, AppScale, Windows Azure Cloud Services, OrangeScape and Jelastic.

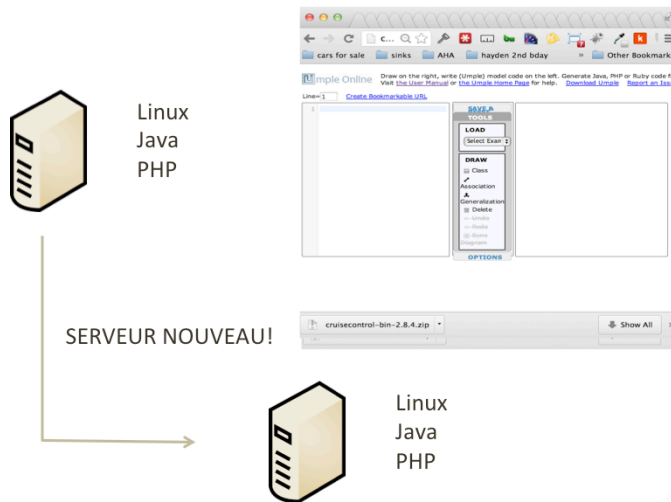
Google Apps, Microsoft Office 365, Petrosoft, Onlive, GT Nexus, Marketo, Casengo, TradeCard, Salesforce and CallidusCloud.



Suivant, je présente des problèmes qui agit comme motivation pour cette séminaire.

Gestion de l'infrastructure

Comment gérons-nous notre serveur / infrastructure du système d'une manière mesurée, contrôlée et cohérente?



2013-06-19

[7]

En premier, la gestion de l'infrastructure. Comment gérons-nous nos serveur d'une manière mesurée, contrôlée et cohérente? Pour elaborer je vais prendre un petit exemple.

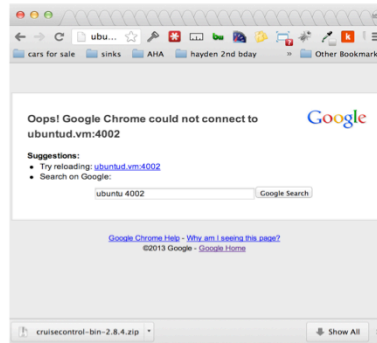
Image que vous avez un serveur Linux avec PHP et Java, et que votre application fonctionne parfaitement. Le département décide qu'il est temps de mettre à niveau et vous donne un nouveau serveur Linux avec PHP et Java. Mais votre application ne fonctionne plus du tout, pourquoi?

Gestion de l'infrastructure (2)



Fedora
Java 1.6
PHP 5.1

Manuel, et sujette à l'erreur, même avec documentation.

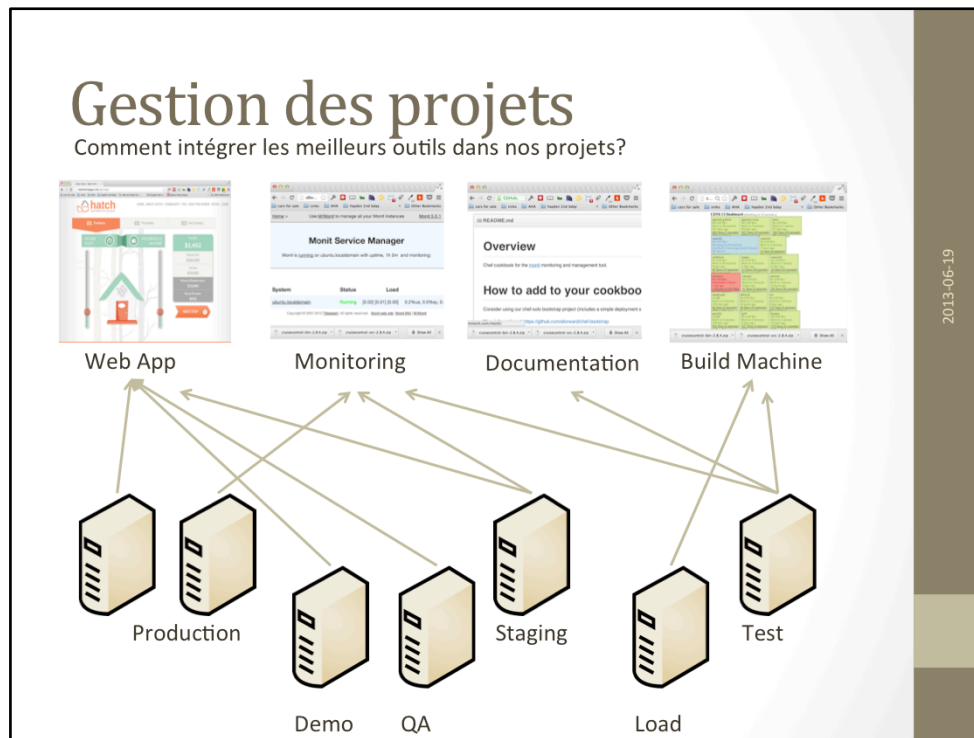


Ubuntu
Java 1.7
PHP

2013-06-19

[8]

Votre serveur a été changé de Fedora pour Ubuntu, a été installé avec Java 1.7 et PHP n'a pas été installé correctement. Le processus manuel de configuration des serveurs, même si vous avez un "script" est sujet aux erreurs, et surtout inconsistent, ce qui conduit à des problèmes. Ou, bien des opportunités pour l'étudier et l'améliorer.



Deuxiement, comment est ce que on peut integrer les outils pour mieux mesurer, controller et surveiller nos logiciels (comme un ingenieur) meme si on travail dans un petit equipe.

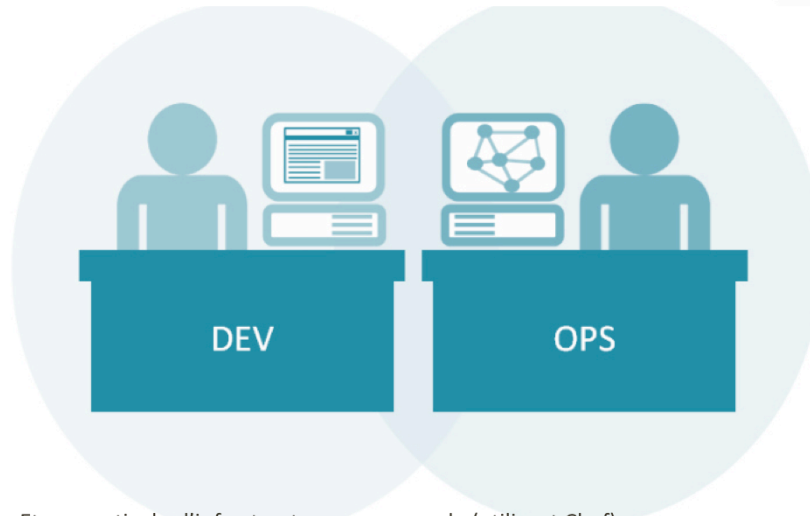
En plus de votre logiciel, votre environnement peut consister des tests automatisés, surveillance, de build-serveur, de machines d'assurance qualité et de documentation (wiki, API, etc).

Et vous pouvez avoir plusieurs environnements comme la production, la mise en scène, test, etc

C'est impratique de configurer individuellement chaque environnement, qui peut avoir plusieurs serveurs, a cause de l'infrastructure qui change fréquemment, avec nouveau version des logicielles, des configurations, etc.

C'est possible, oui, mais c'est pas pratique et ce n'est pas l'ingénierie.

Donc, on utilise DevOps



Et, en particulier l'infrastructure comme code (utilisant Chef).

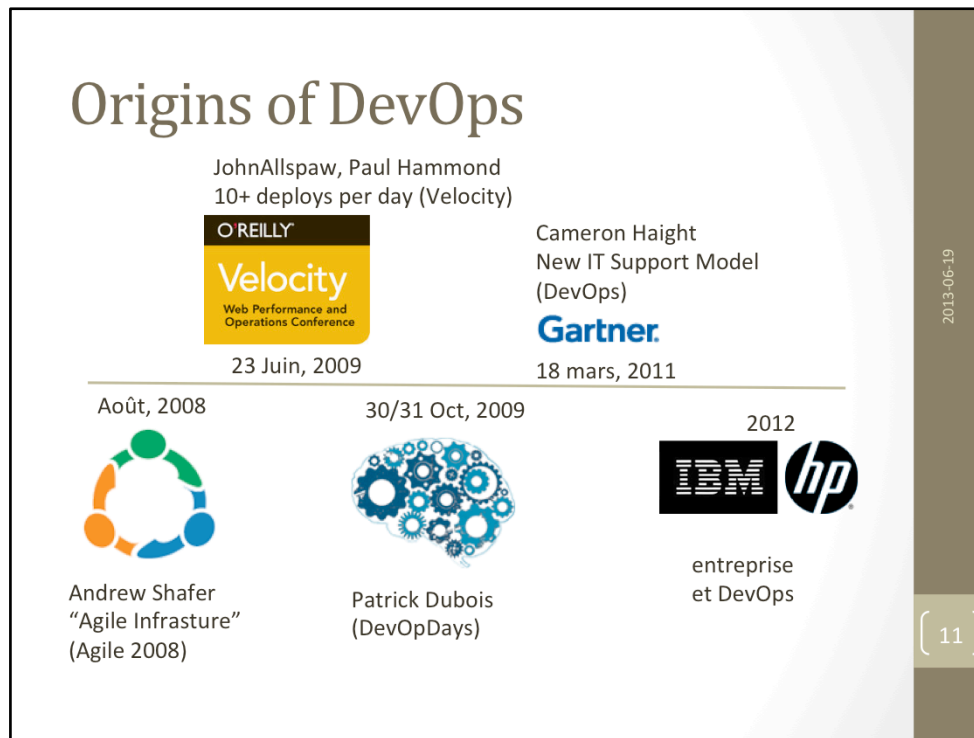
2013-06-19

[10]

DevOps nous permet d'avoir nouveaux langages de programmation pour faciliter les opérations des systems, suivant les principes CAMS -- communauté, de l'automatisation, de mesure et de partage.

Ref: <http://techli.com/collabnet-UC4-software>

<http://www.opscode.com/blog/2010/07/16/what-devops-means-to-me/>



Le concept a pris forme en 2009 avec les interactions entre Andrew Shaffer et Patrick Dubois, ainsi qu'un séminaire donné par John Allspaw et Paul Hammond. La première conférence de DevOpsDays était en Octobre 2009. Momentum a continué à twitter et en ligne et en 2011 un analyste de Gartner, Cameron Haight, a prédit une participation de 20% dans les «DevOps» en 2020. Peu de temps après, DevOps paru dans l'entreprise.

REFERENCES

<http://www.devopsdays.org/>
<http://agile2008.agilealliance.org/>
<http://www.slideshare.net/jallspaw/10-deploys-per-day-dev-and-ops-cooperation-at-flickr>
http://a4.mzstatic.com/us/r30/Podcasts/v4/bd/86/50/bd8650bb-51a4-5dfe-9366-0c561ada02ee/mza_3135987482758484457.170x170-75.jpg
<http://na2.www.gartner.com/imagesrv/images/gartner136.gif;pvab88126431201058>
<http://www-03.ibm.com/ibm/history/exhibits/logo/images/920911.jpg>
http://www.www8-hp.com/us/en/images/THP_S_K_RGB_150_LG_65x65_Ctcm2451096198_Ttcm245108559832_F.png



Nous allons maintenant discuter Chef, un langage de programmation qui facilite DevOps. Mais, il ya beaucoup plus d'options disponibles

<http://www.opscode.com/chef/>

Chef is an open source systems integration framework built to bring the benefits of configuration management to your entire infrastructure. You write source code to describe how you want each part of your infrastructure to be built, then apply those descriptions to your servers. The result is a fully automated infrastructure: when a new server comes on line, the only thing you have to do is tell Chef what role it should play in your architecture.

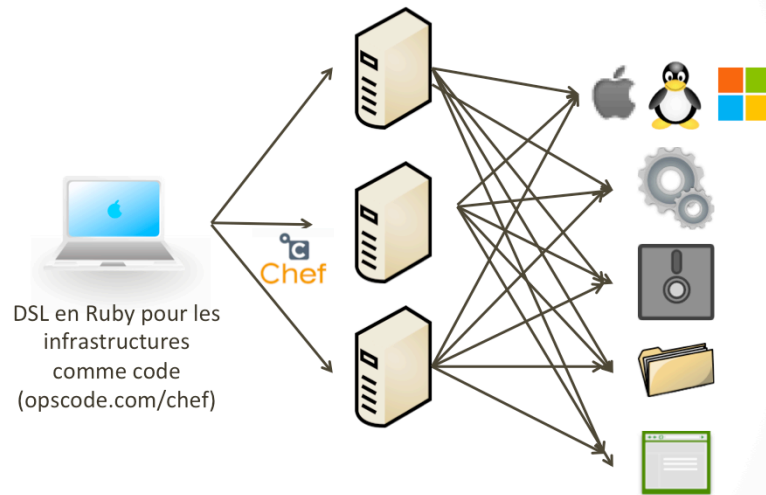
<http://saltstack.com/community.html>

Salt is a powerful remote execution manager that can be used to administer and provision servers in a fast and efficient way. Salt allows commands to be executed across large groups of servers. This means systems can be easily managed, but data can also be easily gathered. Quick introspection into running systems becomes a reality. Remote execution is usually used to set up a certain state on a remote system. Salt addresses this problem as well, the salt state system uses salt state files

<http://cfengine.com/>

Cfengine, the world technology leader in datacenter automation, based on state-of-

Intro à Chef

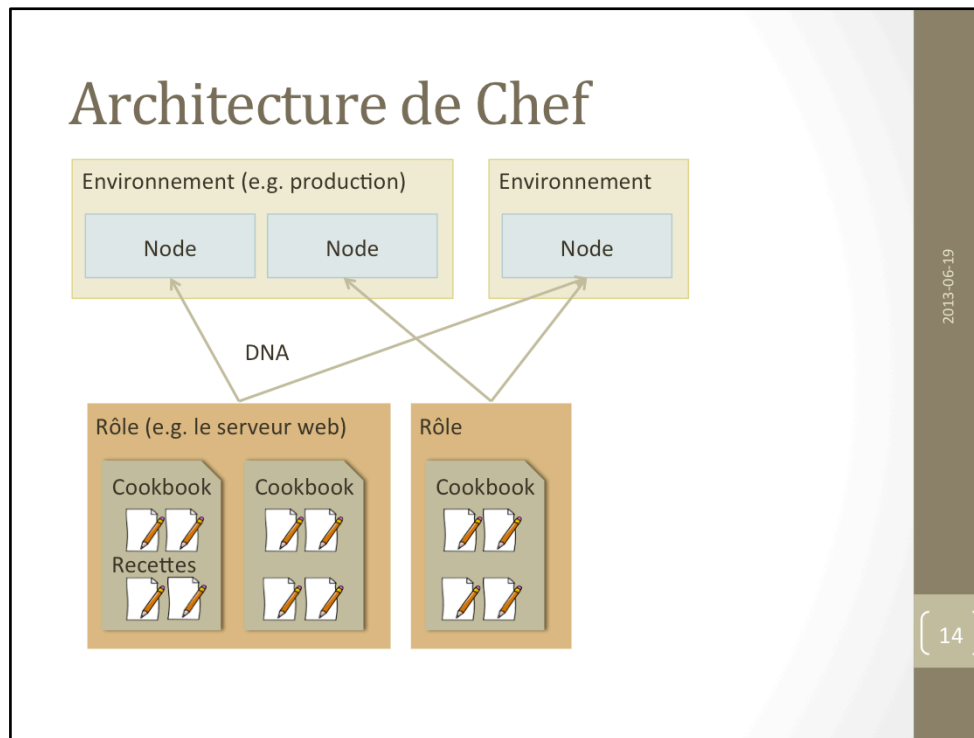


2013-06-19

[13]

Chef est un langage écrit en Ruby, et avec, nous pouvons:

- Installer des systèmes d'exploitation
- Installez les logiciels, les fichiers et les configurations
- Démarrer et Arrêter les services
- En fin, on peut configurer plusieurs fois (un processus idempotentes)



Les ressources sont configurées à l'aide des recettes. Dans ces recettes, on peut créer des fichiers et des documents, puis installer les logiciels, ou bien configurer une connection à un base de données. Une ou plusieurs recettes font partie d'un livre de recettes (cookbook).

Un rôle c'est un groupes de recettes utilisées ensembles (e.g. une combinaison de recettes pour installer dans serveur LAMP). Le DNA est une combinaison de recettes / rôle qui sert pour définir la configuration d'un serveur, qui est l'équivalent d'un node. En fin, Les nodes peuvent être regroupées par environnement, par exemple production ou mise en scène.

Inspired by and thanks to:

Adam Jacob (@adamhjk), released in 2009 under Apache License, Version 2.0.

<http://learnchef.getharvest.com/introduction.html#mapping>

Structures de fichiers de Chef

▼ cookbooks

▼ helloworld

- ▶ attributes
- ▶ definitions
- ▶ files
- ▶ libraries
- ▶ providers
- ▶ recipes
- ▶ resources
- ▶ templates

CHANGELOG.md

metadata.rb

README.md

```
{
  "monit": {
    "version": "5.5.1",
    "address": "monit.vm",
    "port": "4001",
    "allow": [ "admin:admin" ],
    "logfile": "/etc/monit/monit.log",
    "monitrc_file": "/etc/monitrc"
  },
  "run_list": [ "role[bare]", "recipe[monit]" ]
}
```

DNA

Ressources

2013-06-19

[15]

Une autre façon de visualiser Chef est de la structure des fichiers d'un livre de recettes.

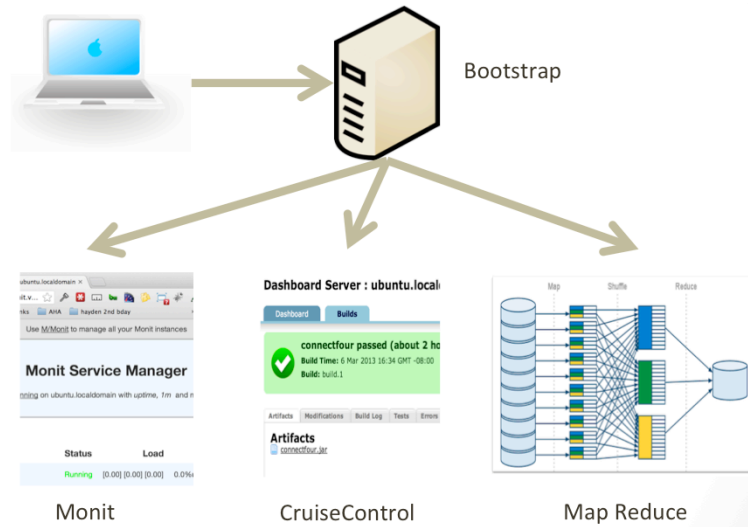
Ici, nous voyons que les recettes peuvent comporter des attributs, des définitions, des fichiers, des modèles, des recettes, etc.

Comme le temps le permet, nous allons prendre plusieurs exemples.

Images:

<https://github.com/json-schema>

Exemples avec Chef



2013-06-19

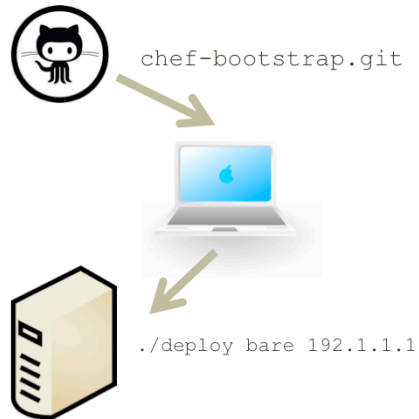
[16]

D'abord, bootstrapping CHEF. Chef est un logiciel complexe. Bootstrap est un project qui nous permet d'installer Chef dans notre serveur. Cette configuration de bootstrap est utiliser avec chef-solo et on n'a pas besoin d'un serveur specifique a Chef.

En suite, nous pourrions installer un service de surveillance Monit, ou un build-serveur.

En fin, et l'exemple qu'on va discuter le plus et un exemples de MapReduce.

Examples: Bootstrapping Chef



Update README.md

 **aforward** authored 16 hours ago

 .chef

 bootstrap

 chef

 .gitignore

 Capfile

 README.md

 denlv

2013-06-19

[17]

<https://github.com/aforward/chef-bootstrap>

Chef Examples - execute

```
dirname = "monit-#{node[:monit][:version]}"
filename = "#{dirname}-linux-x64.tar.gz"

cookbook_file "/tmp/#{filename}" do
  source filename
  owner 'root'
  group 'root'
  mode '0644'
end

execute "tar xzfv #{filename}" do
  cwd "/tmp"
  not_if { File.exists?("/tmp/#{dirname}") }
end

execute "cp #{dirname}/bin/monit #{node[:monit][:bin_file]}" do
  cwd "/tmp"
  user 'root'
  group 'root'
  only_if { `monit -V | grep #{node[:monit][:version]}`.empty? }
end
```

2013-06-19

[18]

Chef Examples - upstart

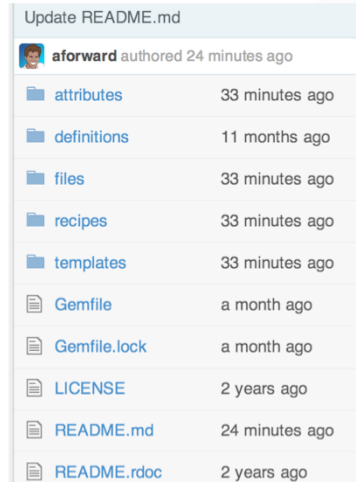
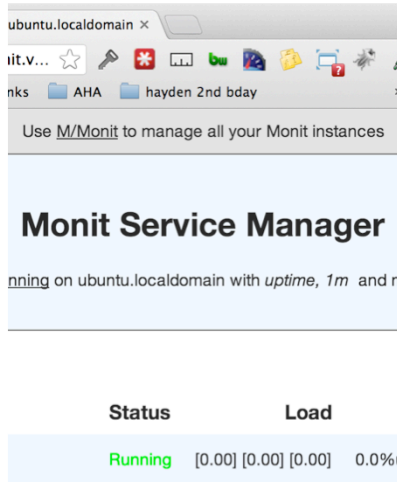
```
template "/etc/init/monit.conf" do
  owner "root"
  group "root"
  mode 0700
  source 'monit.conf.erb'
end

service "monit" do
  provider Chef::Provider::Service::Upstart
  supports :status => true, :restart =>
true, :reload => true
  action [ :enable ]
end
```

2013-06-19

[19]

Examples: Monit



<https://github.com/aforward/chef-monit>

Il y a un livre de menu à source ouvert pour cette recettes, chef-monit.

<https://github.com/aforward/chef-monit>

Je vais vous presenter certain exemples en fonction du temps qui m'est alloué

Examples: Build Server

Dashboard Server : ubuntu.local

Dashboard

Builds



connectfour passed (about 2 h
Build Time: 6 Mar 2013 16:34 GMT -08:00
Build: build.1

Artifacts

Modifications

Build Log

Tests

Errors

Artifacts
 [connectfour.jar](#)

 **aforward** authored 2 hours ago

attributes

files

recipes

templates

Gemfile

Gemfile.lock

LICENSE

README.md

2013-06-19

[21]

<https://github.com/aforward/chef-cruisecontrol>

MapReduce

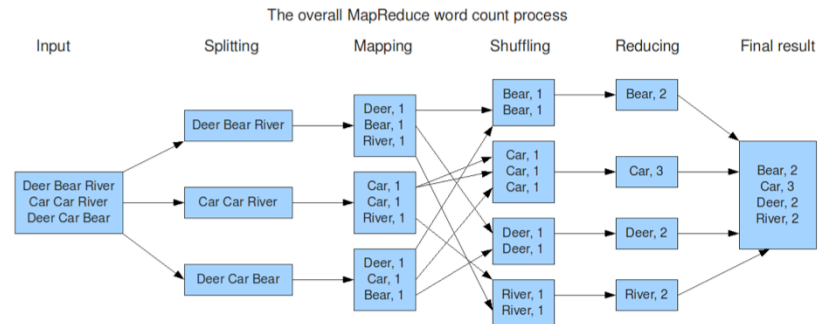
- GFS (Google File System) and MapReduce in 2004
- HFS and Hadoop open sourced under Apache
 - Parallel processing on hundreds of nodes
- BigTable in 2006, and Hbase was born
 - Store data in massive tables (billion rows / million columns)
 - Retrieve key/value pairs in real-time
- Google later released
 - Sawzall (query language) in 2005
 - Pig & Hive (batch queries) in 2008
 - Spanner (online queries like joins / transactions) in 2012

2013-06-19

[22]

<http://classpattern.com/what-is-hadoop.html>

Map, Shuffle, Reduce



Both Map and Reduce are stateless – so can be parallelized with ease (the MapReduce algorithms manage the distribution of the processing parts and the consolidation of the results)

2013-06-19

(23)

<http://blog.jteam.nl/wp-content/uploads/2009/08/MapReduceWordCountOverview1-300x139.png>

Example: Top Collaborators

- Analyze author collaborations, e.g.

Omar Badreddin, Andrew Forward: Model Oriented Programming: An Empirical Study of Comprehension. CASCON 2012

Lethbridge, T., Mussbacher G., and Badreddin, O, (2011) "Teaching UML Using Umple: Applying Model-Oriented Programming in the Classroom", CSEE&T 2011, pp. 421-428.

Lethbridge, T.C., Singer, J and Forward, A., (2003) "How software engineers use documentation: the state of the practice", IEEE Software special issue: The State of the Practice of Software Engineering, Nov/Dec 2003, pp 35-39.

Badreddin, O and Lethbridge, T. (2012) 'Combining Experiments and Grounded Theory to Evaluate a Research Prototype: Lessons from the Umple Model-Oriented Programmin

- For simplification, we will refer to the authors as
 - A : Andrew Forward
 - O : Omar Badreddin
 - T : Timothy C. Lethbridge
 - G : Gunther Mussbacher
 - J : Janice Singer

2013-06-19

[24]

Group Authors (Map)

- First, map the list of authors to each other

O: (O A)	T: (T G O)	T: (T J A)	O: (O T)
A: (O A)	G: (T G O)	J: (T J A)	T: (O T)
	O: (T G O)	A: (T J A)	

- Which is grouped / shuffled as

O: (O A) (T G O) (O T)
A: (O A) (T J A)
T: (T G O) (T J A) (O T)
G: (T G O)
J: (T J A)

Count Collaborations (Reduce)

- Identify all collaborations (union the lists)

O	:	(	O	A	G	T	)	
A	:	(	A	O	T	J	)	
T	:	(	O	A	T	G	J	)
G	:	(	T	G	O	)		
J	:	(	T	J	A	)		

- Or, count the collaborations between authors

O	:	(	O	3	)	(	A	1	)	(	T	2	)	(	G	1	)				
A	:	(	O	1	)	(	A	2	)	(	T	1	)	(	J	1	)				
T	:	(	T	3	)	(	G	1	)	(	A	1	)	(	O	2	)	(	J	1	)
G	:	(	T	1	)	(	G	1	)	(	O	1	)								
J	:	(	T	1	)	(	J	1	)	(	A	1	)								

Common Collaborators (Chain)

- What if we wanted to know collaborators in commun?
- Map the reduction for more results

(O O) : (O A G T)	(G T) : (T G O)
(A O) : (O A G T)	(G G) : (T G O)
(G O) : (O A G T)	(O G) : (T G O)
(O T) : (O A G T)	
(A A) : (A O T J)	(J T) : (T J A)
(A O) : (A O T J)	(J J) : (T J A)
(A T) : (A O T J)	(A J) : (T J A)
(A J) : (A O T J)	
(O T) : (O A T G J)	
(A T) : (O A T G J)	
(T T) : (O A T G J)	
(G T) : (O A T G J)	
(J T) : (O A T G J)	

Common Collaborators (con't)

- Join collaboration pairs (shuffle)

(O O) : (O A G T)	(A A) : (A O T J)
(A O) : (O A G T) (A O T J)	(A T) : (A O T J) (O A T J)
(O T) : (O A G T) (O A T J)	(G T) : (T G O) (O A T J)
(J T) : (T J A) (O A T J)	(G G) : (T G O)
(A J) : (T J A) (A O T J)	(G O) : (T G O)
(J J) : (T J A)	(T T) : (O A T G J)

2013-06-19

[28]

Common Collaborators (reduce)

- Reduce by taking the union of the lists, and then removing the collaborators

(O O) : (A T)	(A A) : (O T J)
(A O) : (T)	(A T) : (O J)
(O T) : (A)	(G T) : (O)
(J T) : (A)	(G G) : (T O)
(A J) : (T)	(O G) : (T)
(J J) : (T A)	(T T) : (O A G J)

- So, if A visits T's research profile, he would see that they have both collaborated with O and J

Chef + MapReduce

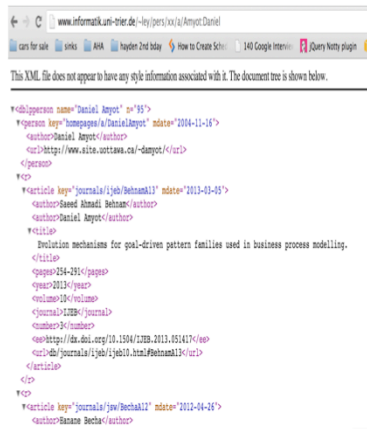
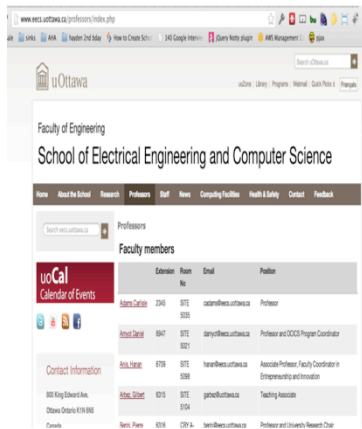
- Manage the installation of your hadoop / HFS deployment
- Configure Single-Node servers for algorithm testing, but Multi-Node for production
- Enable dynamic / elastic provisioning

```
iex(3)> HashDict.fetch(c,"Daniel Amyot")
{:ok,["Saeed Ahmadi Behnam","Daniel Amyot","Hanane Be
eiss","Jo6#227;o Ara6#250;jo 0001","Ana Moreira","Jas
","Edna Braun","Nick Cartwright","Mira Vrbaski","Dord
niele Barone","Lei Jiang","Eric S. K. Yu","St6#233;ph
eddiine Hassine","Marconi Lanna","S6#233;bastien Mosse
i","Liam Peyton","Flavio Rizzolo","Alberto Siena","Ar
"Dragon Gasevic","Marek Hatala","Nicolas Genon","Ali
iex(4)> HashDict.fetch(c,"Timothy Lethbridge")
{:ok,["Omar Bahy Badreddin","Timothy Lethbridge","His
","Janice Singer","Abdelwahab Hamou-Lhadj","Edna Braun
rie Thompson","Joanne M. Atlee","Ann E. Kelley Sobel"
","Lianjiang Fu","John Impagliazzo","Esther A. Hughes
Anquetil","Mordechai Ben-Menachem","Thomas B. Hilburn
abad","Stan Matwin","Arie van Deursen","Perdita Steve
"Douglas R. Skuce","Colin Ware"]}
iex(5)> HashDict.fetch(c,"Dwight Deugo")
{:ok,["Andrew Forward","Timothy Lethbridge","Dwight D
iex(6)>
```

2013-06-19

[30]

MapReduce in Elixir



<https://github.com/aforward/chef-elixir-mapreduce>

<https://github.com/aforward/chef-elixir>

<https://github.com/aforward/collabs>

<http://elixir-lang.org/>

2013-06-19

[31]

<http://www.eecs.uottawa.ca/professors>

<http://www.informatik.uni-trier.de/~ley/pers/ho/l/Lethbridge:Timothy.html>

Elixir MapReduce

```
defmodule JointAuthors do
  def map(list), do: map(list, list, HashDict.new [])
  def map([], _, hash), do: hash
  def map([head|tail], list, hash) do
    map(tail, list, HashDict.put(hash, head, list))
  end
  def reduce(list), do: reduce(list, [])
  def reduce([], reduced), do: Enum.uniq(reduced)
  def reduce([head|tail], reduced) do
    reduce(tail, reduced ++ head)
  end
end
```

<https://github.com/afoward/chef-elixir-mapreduce>
<https://github.com/afoward/collabs>

2013-06-19

[32]

```
c = Collabs.CLI.run(["-o"])
Dict.fetch(c, "Timothy Lethbridge")
Dict.keys(c)
```

Sample Collaborations

```
iex(31)> Dict.fetch(output,"Daniel Amyot")
{:ok,["Saeed Ahmadi Behnam","Daniel Amyot","Hanane
Becha","Alireza Pourshahid","Azalia Shamsaei","Gunter
Mussbacher","Michael Weiss","Jo&#227;o Ara&#250;jo
0001","Ana Moreira","Jason Kealey","Etienne
Tremblay","Andrew Miga","Mohammad Alhaj","Rasha
Tawhid","Edna Braun","Nick Cartwright","Mira
Vrbaski","Dorina C. Petriu","Jennifer
Horkoff","Alexander Borgida","John Mylopoulos","Daniele
Barone","Lei Jiang","Eric S. K. Yu","St&#233;phane
Leblanc","J&#246;rg Kienzle","Mario Saucier","Gregory
Richards","Jameleddine Hassine","Marconi
Lanna","S&#233;bastien Mosser","Mireille Blay-
Fornarino","Patrick Heymans","Hao Luo","Sepideh
Ghanavati","Liam Peyton","Flavio Rizzolo","Alberto
Siena","Anna Perini","Angelo Susi","Jon Whittle","Marko
Boskovic","Ebrahim Bagheri","Dragan Gasevic","Marek
Hatala","Nicolas Genon","Ali Echihabi","Yong
He","Pengfei Chen","Alan J. Forster",...]}
```

2013-06-19

[33]

Les possibilités de recherche

Research avenues

2013-06-19

[34]

Je vous ai parlé de la conception futuriste des nuages en informatique et de l'infrastructure comme code en utilisant Chef. C'est un projet de grande ampleur que je compte développer au sein du département.

À mon avis, les avenues de recherche comprennent:

La research avec la programmation de l'infrastructure. On peut étudier, expérimenter, et d'améliorer la façon dont nous gérons les services de cloud computing. Cela comprend des activités de recherche tels que la construction d'une taxonomie pour les catégories des logiciels de DevOps, l'analyse et l'expérimentation de logiciels disponibles ainsi que la recherche à coorelations entre le développement d'applications et le développement de l'infrastructure.

En plus, la voit concernant la research dans les nuages en generale où on utilise un logiciel comme Chef pour faciliter la configuration de notre nuages. Cela nous laisse libre de se concentrer sur nos problèmes au Big Data. Par exemple, fournir de meilleurs outils pour l'examen de la littérature en analysant les auteurs et des références dedans les publications.

Questions?

Dr. Andrew Forward

forward@gmail.com / @a4word

Le 19 juin, 2013

2013-06-19

[35]

Je suis a vous pour repondres a vos questions
Je vous remercie mesdame et mousiers pour votre attention.

Why work at university?

J'aime la recherche, et j'aime transmettre mon savoir, et pour travailler en
collaboration avec l'industrie privés.

J'aime faire de la recherche en informatiques et en fin j'aime travailler en collaboration
avec le secteur industriel dans le caudre de recherche et development.

J'aime travailler à l'universite parce que j'aime trav

Resources

- Presentation Code
 - <https://github.com/aforward/chef-bootstrap>
 - <https://github.com/aforward/chef-monit>
 - <https://github.com/aforward/chef-runas>
- Cloud Definitions
 - <http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf>
- DevOps
 - <http://itrevolution.com/the-history-of-devops/>
 - <http://www.slideshare.net/KrisBuytaert/7-tools-for-your-devops-stack>
 - <http://itrevolution.com/the-history-of-devops/>
 - <http://www.devopsdays.org/images/devopsdays-banner.png>
 - <http://agile2008.agilealliance.org/images/headerLogo.png>
- Chef
 - <http://www.opscode.com/chef/>
 - <http://community.opscode.com/cookbooks>
 - <http://docs.opscode.com/ohai.html>
- Books
 - Test-Driven Infrastructure with Chef, Stephen Nelson-Smith, 2011

More References

- MapReduce
 - <https://developers.google.com/appengine/docs/python/dataprocessing/>
 - <http://classpattern.com/what-is-hadoop.html>
 - <http://stevekrenzel.com/finding-friends-with-mapreduce>
 - http://hadoop.apache.org/docs/stable/mapred_tutorial.html
 - http://hadoop.apache.org/docs/stable/single_node_setup.html
 - <http://hadoop.apache.org/releases.html>
 - <https://github.com/derekchiang/Elixir-MapReduce>
 - <http://xiaochongzhang.me/blog/?p=338>
 - <http://www.cs.berkeley.edu/~ballard/cs267.sp11/hw0/results/htmls/Muzaffar.html>

Image References

- <http://www.clker.com/>
- <http://www.cablefax.com/tech/deployment/casestudies/21910.html>
- <http://techli.com/collabnet-UC4-software>
- <http://i.s-microsoft.com/global/ImageStore/PublishingImages/logos/hp/logo-lg-2x.png>
- <http://www.apple.com/favicon.ico>
- https://developers.google.com/appengine/docs/python/images/mapreduce_mapshuffle.png
- <http://www.cs.berkeley.edu/~ballard/cs267.sp11/hw0/results/htmls/images/mapreduce.png>
- <http://blog.jteam.nl/wp-content/uploads/2009/08/MapReduceWordCountOverview1-300x139.png>

Related Work

Roberto Di Cosmo, Stefano Zacchiroli, and Gianluigi Zavattaro discuss a formal component model for managing infrastructure in the cloud. [1]

J. Weinman quantifies the benefits of cloud computing and defines a mechanism to axiomatically define and analyze cloud benefits called Cludonomics [2].

Gunawi [3] introduced Failure Scenario's as a server (FSaaS), probably most known in "Chaos Monkey" [4] a product open sourced by Netflix. Faraz Faghri [5] developed FSaaS for Hadoop Clusters.

2013-06-19

[39]

[1] In particular to deal with elastic infrastructure where you must manage the upgrading of multiple servers, called Aeolus. Di Cosmo formalize through state machines the core scenarios for infrastructure management including: Package installation; Services, Redundancy / Capacity Planning / Conflicts; and creating and destroying resources.

Related Work (cont)

- [1] Towards behavior driven operations (BDOps)
Gohil, Komal ; Alapati, Nagalakshmi ; Jogekar, Sunil
Advances in Recent Technologies in Communication and Computing (ARTCom 2011), 3rd International Conference on
Digital Object Identifier: 10.1049/ic.2011.0095
Publication Year: 2011 , Page(s): 262 - 264
- [2] J. Weinman. Cloudonomics: A rigorous approach to cloud benefit quantification. The Journal of Software Technology, 14:10-18, October 2011.
- [3] H. S. Gunawi, T. Do, J. M. Hellerstein, I. Stoica, D. Borthakur, and J. Robbins. Failure as a Service (FaaS): A cloud service for large-scale, online failure drills. Technical Report UCB/EECS-2011-87, EECS Department, University of California, Berkeley, Jul 2011.
- [4] ChaosMonkey. <http://techblog.netflix.com/2010/12/5-lessons-weve-learned-using-aws.html>.
- [5] Failure Scenario as a Service (FSaaS) for Hadoop Clusters
- [6] X. Zhang, S. Dworkadas, G. Folkmanis, and K. Shen. Processor hardware counter statistics as a first-class system resource. In Proceedings of the 11th USENIX workshop on Hot topics in operating systems, HOTOS'07, pages 14-1-14-6, Berkeley, CA, USA, 2007. USENIX Association.
- [7] M. Isard, V. Prabhakaran, J. Currey, U. Wieder, K. Talwar, and A. Goldberg. Quincy: fair scheduling for distributed computing clusters. In Proceedings of the ACM SIGOPS 22nd symposium on Operating systems principles, SOSP '09, pages 261-276, New York, NY, USA, 2009. ACM.
- [8] Devops: A software revolution in the making? Cutter IT Journal, 24(8), 2011. Special Issue.
- [9] S. McIntosh, B. Adams, Y. Kamei, T. Nguyen, and A. E. Hassan. "An empirical study of build maintenance effort." In Proc. of Intl. Conf. on Software Engineering (ICSE), 2011, pp. 141-150.
- [10] J. A. Whittaker, J. Arbon, and J. Carollo. How Google Tests Software. Addison-Wesley Professional, April 2012.
- [11] R. DeLine. "Avoiding packaging mismatch with flexible packaging." in Proc. of Intl. Conf. on Software Engineering (ICSE), 1999, pp. 97-106.
- [12] A. van der Hoek and A. L. Wolf. "Software release management for component-based software." *Softw. Pract. Exper.*, vol. 33, pp. 77-98, January 2003.
- [13] J. Humble and D. Farley, *Continuous Delivery*, 1st ed. Addison Wesley, August 2010.
- [14] T. Fitz. "Continuous deployment at IMVU: Doing the impossible fifty times a day," <http://goo.gl/qPT6>, February 2009.
- [15] S. Shankland. "Google ethos speeds up Chrome release cycle," <http://goo.gl/vWwlr>, July 2010.
- [16] F. Khomh, T. Othawal, Y. Zou, and B. Adams. "Do faster releases improve software quality? an empirical case study of mozilla firefox," in Proc. of the Working Conf. on Mining Software Repositories (MSR), 2012.
- [17] M. Armbrust et al. Above the clouds: A berkeley view of cloud computing. In Tech. Rep. UCB/EECS-2009-28, EECS Department, University of California, Berkeley, 2009.
- [18] Don't Install Software by Hand
Spinnells, D.
Software, IEEE
Volume: 29 , Issue: 4
Digital Object Identifier: 10.1109/IMS.2012.85
Publication Year: 2012 , Page(s): 86 - 87
- [19] Fast Development Platforms and Methods for Cloud Applications
Hosono, S. ; Jiafu He ; Xuemei Liu ; Lin Li ; He Huang ; Yoshino, S.
Services Computing Conference (APSCC), 2011 IEEE Asia-Pacific
Digital Object Identifier: 10.1109/APSCC.2011.75
Publication Year: 2011 , Page(s): 94 - 101
- [20] Building IaaS Clouds and the art of virtual machine management
Montero, R.S.
High Performance Computing and Simulation (HPCS), 2012 International Conference on
Digital Object Identifier: 10.1109/HPCS.2012.6266975
Publication Year: 2012 , Page(s): 573

More Chef Details

2013-06-19

[41]

Les bases Chef

On peut...

- Installer des systèmes d'exploitation
- Installez les logiciels
- Démarrer / Arrêter les services
- Configurer plusieurs fois (idempotentes)

En utilisant...

- Ohai
- Chef-client
- Chef-server
- Chef-solo
- Knife
- Chef



<http://www.opscode.com/chef/>

2013-06-19

[42]

Chef comprend 4 outils.

D'abord Ohai, c'est un system qui nous permet de connaitre toutes la composition du system: le logiciel, versionne, etc.

Ensuite, le client qui installe et configure un system et le serveur qui coordonne les activités de tous les serveurs, et chef-solo qui permet de jouer le rôle d'un system de distribution

Puis, Knife qui gère toutes les commandes d'installation de chef

Et enfin, Chef avec un S et non pas un C, est un programme utilisé pour corriger les fautes qui surgissent

Revenant à Chef avec un grand C

Ohai

System profiler / asset collector

Knife Can...

- Create cookbook templates
- Download global cookbooks
- Upload cookbooks (chef-server only)
- Edit nodes, environments, and roles (chef-server only)

2013-06-19

[43]

Ohai Can...

Attribute	Description
node['platform']	The platform on which a node is running.
node['platform_version']	The version of the platform.
node['ipaddress']	The IP address for a node. If the node has a default route, this is the IPV4 address for the interface.
node['macaddress']	The MAC address for a node.
node['fqdn']	The fully qualified domain name for a node.
node['hostname']	The host name for the node.
node['domain']	The domain for the node.
node['recipes']	A list of recipes associated with a node (and part of that node's run-list).
node['roles']	A list of roles associated with a node (and part of that node's run-list).
node['ohai_time']	The time at which Ohai was last run.

2013-06-19

[44]

\$ ohai

```
Andrew-MacBook-Pro:~ aforward$ ohai
[2013-06-12T08:27:47-06:00] INFO: [inet6] no default interface, picking the first ipaddress
[2013-06-12T08:27:47-06:00] INFO: ipaddress and ipaddress are set from different interfaces (en0 & lo0), macaddress has been set using the ipaddress interface
{
  "languages": {
    "ruby": {
      "platform": "x86_64-darwin12.1.1",
      "version": "1.9.2",
      "release_date": "2011-07-09",
      "target": "x86_64-apple-darwin12.1.1",
      "target_cpu": "x86_64",
      "target_vendor": "apple",
      "target_os": "darwin12.1.1",
      "host": "1386-apple-darwin12.1.1",
      "host_cpu": "x86_64",
      "host_os": "darwin12.1.1",
      "host_vendor": "apple",
      "bin_dir": "/Users/afoward/.rbenv/versions/1.9.2-p290/bin",
      "ruby_bin": "/Users/afoward/.rbenv/versions/1.9.2-p290/bin/ruby",
      "gem_dir": "/Users/afoward/.rbenv/versions/1.9.2-p290/lib/ruby/gems/1.9.1",
      "gem_bin": "/Users/afoward/.rbenv/versions/1.9.2-p290/bin/gem"
    },
    "erlang": {
      "version": "5.9.2",
      "options": {
        "SMP": "ASYNC_THREADS",
        "HYPE": ""
      },
      "emulator": "BEAM"
    },
    "java": {
      "version": "1.7.0_10",
      "runtime": {
        "name": "Java(TM) SE Runtime Environment",
        "build": "1.7.0_10-b18"
      },
      "hotspot": {
        "name": "Java HotSpot(TM) 64-Bit Server VM",
        "build": "23.6-bbb, mixed mode"
      }
    },
    "nodejs": {
      "version": "0.6.13"
    },
    "perl": {
      "version": "5.12.4",
      "archname": "darwin-thread-multi-2level"
    },
    "php": {
      "version": "5.3.15",
      "builddate": "1.11 (built: Aug"
    },
    "python": {
      "version": "2.7.2",
      "builddate": "Oct 11 2012, 20:14:37"
    },
    "kernel": {

```

2013-06-19

[45]

Try it on the command line

Common Chef Resources

- Package (e.g. yum, apt-get)
- File
- Directory
- Template (using ERB)
- Service (e.g. Upstart)
- Execute (i.e. `tar -zxfv ...`)
- Cron
- Git
- Group
- Mount
- User

2013-06-19

[46]

Example Chef Roles

- Load Balancer
- Database Master / Slave
- File / Media Server
- Web Server
- Build Server
- Application Specific Server

2013-06-19

[47]

Example Chef Environments

- Development
- Staging
- User Acceptance
- Production

2013-06-19

[48]

Chef Examples - execute

```
dirname = "monit-#{node[:monit][:version]}"
filename = "#{dirname}-linux-x64.tar.gz"

cookbook_file "/tmp/#{filename}" do
  source filename
  owner 'root'
  group 'root'
  mode '0644'
end

execute "tar xzfv #{filename}" do
  cwd "/tmp"
  not_if { File.exists?("/tmp/#{dirname}") }
end

execute "cp #{dirname}/bin/monit #{node[:monit][:bin_file]}" do
  cwd "/tmp"
  user 'root'
  group 'root'
  only_if { `monit -V | grep #{node[:monit][:version]}`.empty? }
end
```

2013-06-19

[49]

Chef Examples - upstart

```
template "/etc/init/monit.conf" do
  owner "root"
  group "root"
  mode 0700
  source 'monit.conf.erb'
end

service "monit" do
  provider Chef::Provider::Service::Upstart
  supports :status => true, :restart =>
true, :reload => true
  action [ :enable ]
end
```

2013-06-19

[50]

DevOps

- CAMS
 - Culture
 - Automtion
 - Measurement
 - Sharing
- (Damon Edwards & John Willis)
- Deployment
 - Fai, kickstart, preseed, cobbler
- Configuration
 - Chef, puppet, etc
- Build
 - Jenkins, maven, ant, cruisecontrol, hudson

2013-06-19

[51]

CAMS - Damon