

Le contour

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

L'ingénierie dans les nuages

Engineering your cloud infrastructure using CHEF

Dr. Andrew Forward

aforward@gmail.com / @a4word

Le 19 juin, 2013

Les nuages informatiques

Cloud Definitions

Publiques / Privée / Hybride



IaaS



PaaS



SaaS

Nouvelle conceptions: NaaS, HaaS, MaaS, FSaaS

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

~~Les problèmes~~

Les opportunités

Apportez-moi des solutions, pas des problèmes

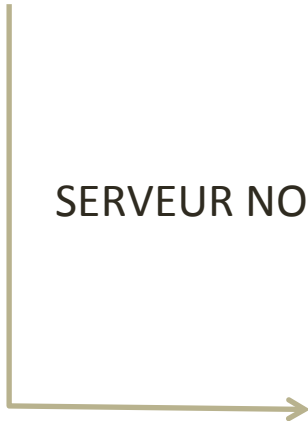
Gestion de l'infrastructure

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

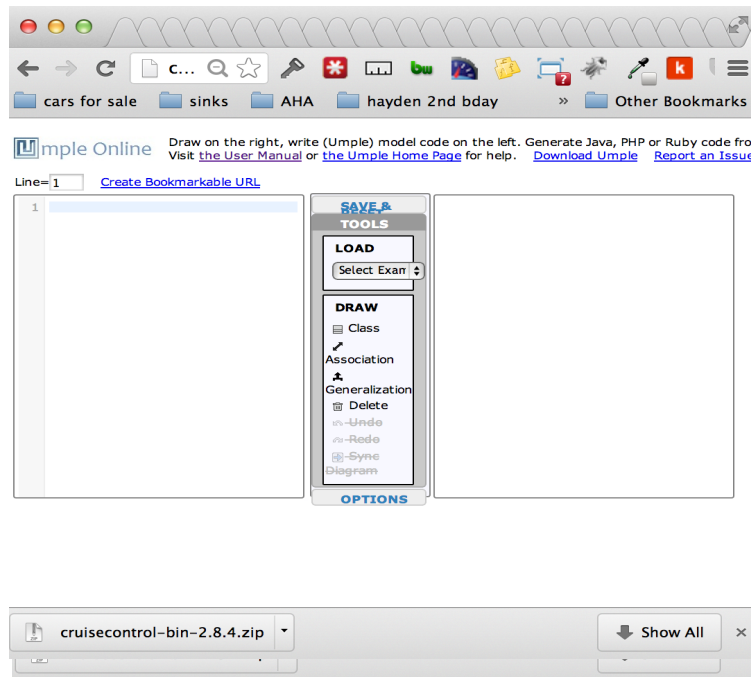


Linux
Java
PHP

SERVEUR NOUVEAU!



Linux
Java
PHP

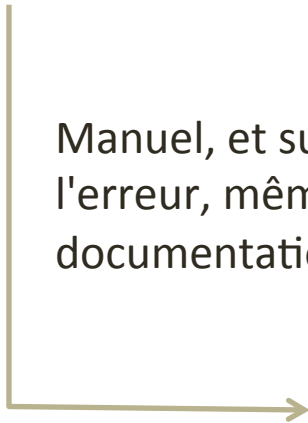


Gestion de l'infrastructure (2)

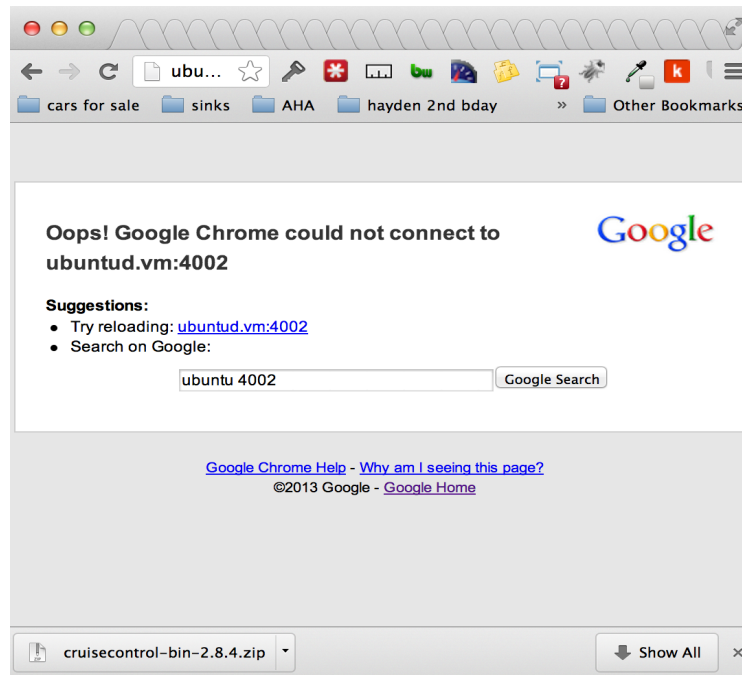


Fedora
Java 1.6
PHP 5.1

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

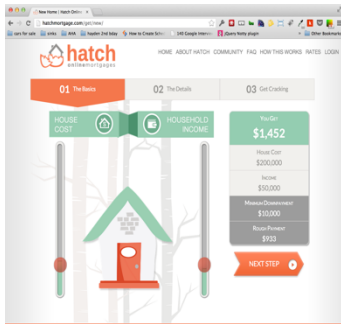


Ubuntu
Java 1.7
PHP

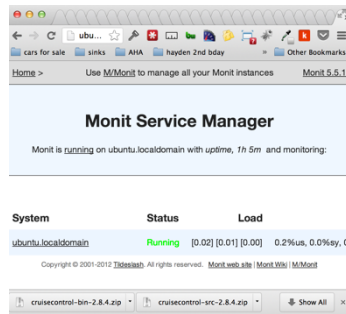


Gestion des projets

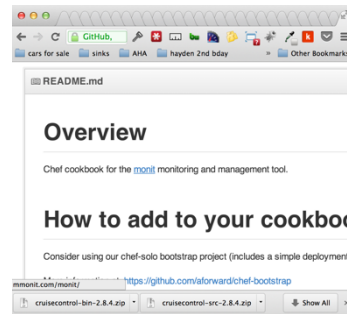
Comment intégrer les meilleurs outils dans nos projets?



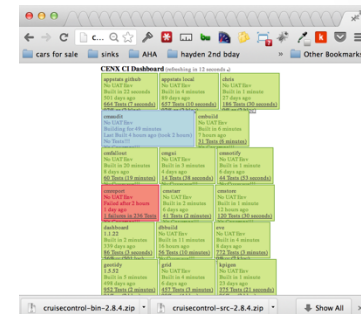
Web App



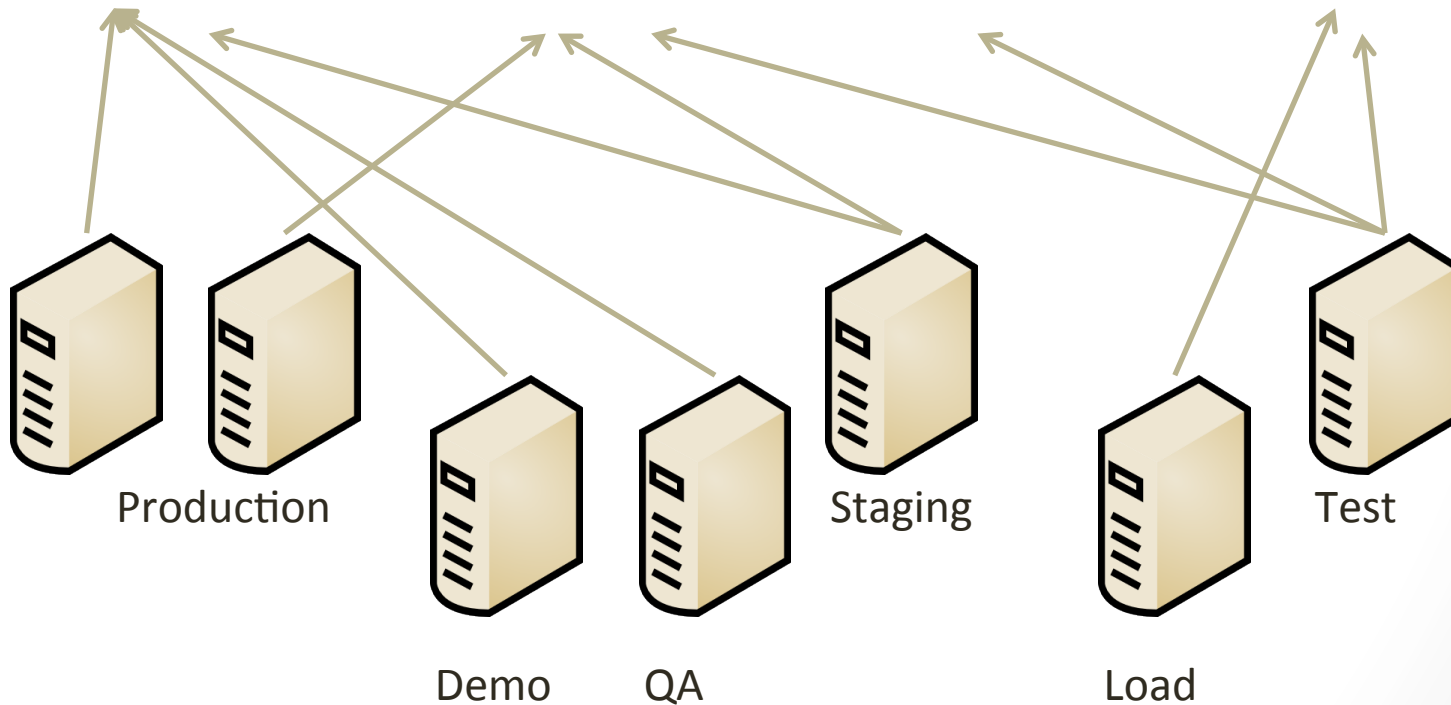
Monitoring



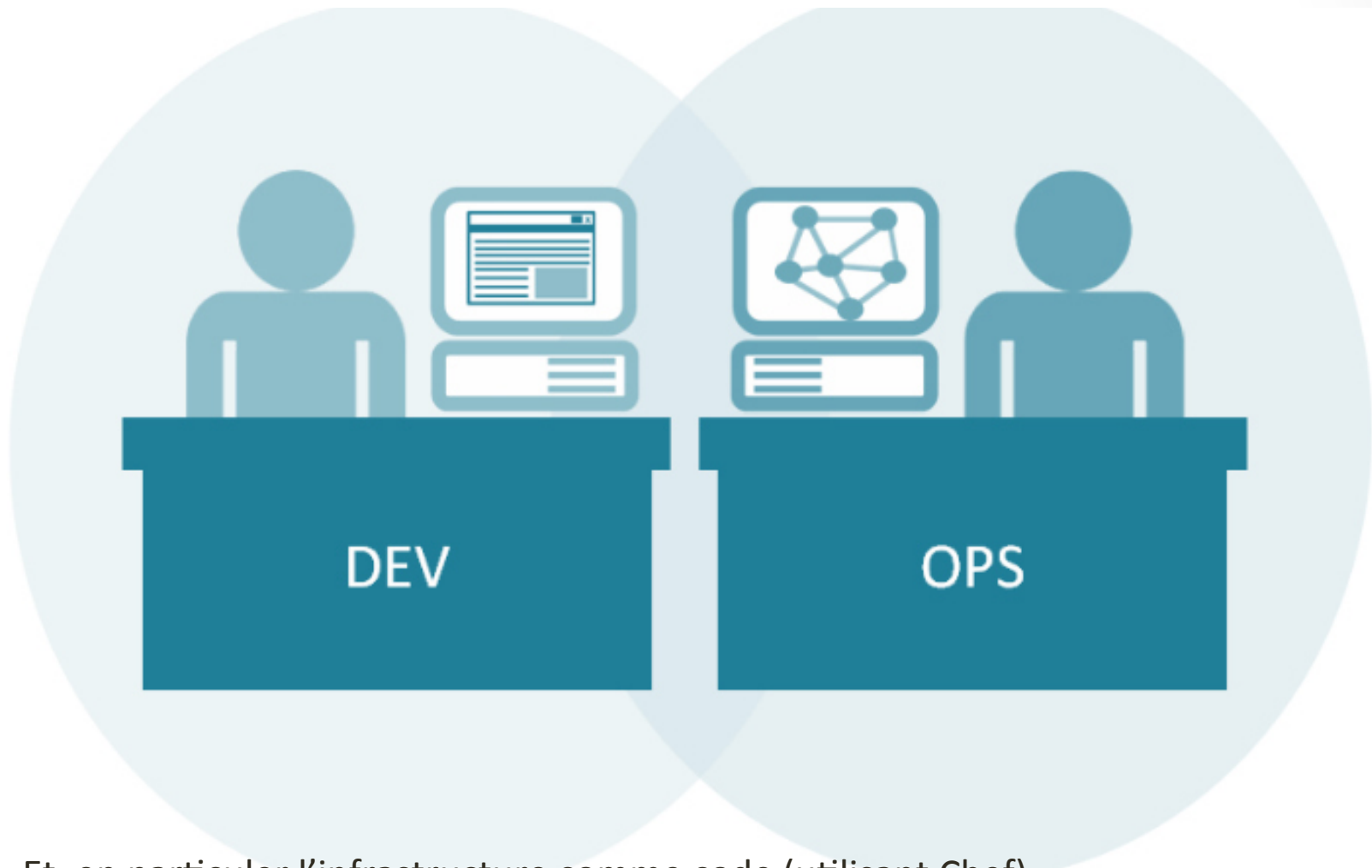
Documentation



Build Machine



Donc, on utilise DevOps



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

Origins of DevOps

JohnAllspaw, Paul Hammond
10+ deploys per day (Velocity)



23 Juin, 2009

Cameron Haight
New IT Support Model
(DevOps)

Gartner.

18 mars, 2011

Août, 2008



Andrew Shafer
"Agile Infrastructure"
(Agile 2008)

30/31 Oct, 2009

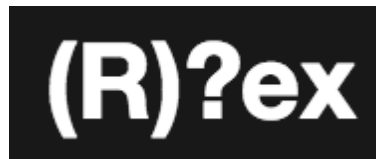


Patrick Dubois
(DevOpsDays)

2012



entreprise
et DevOps

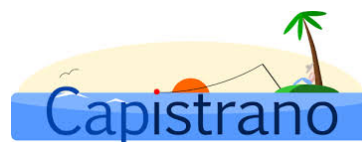


Fabric

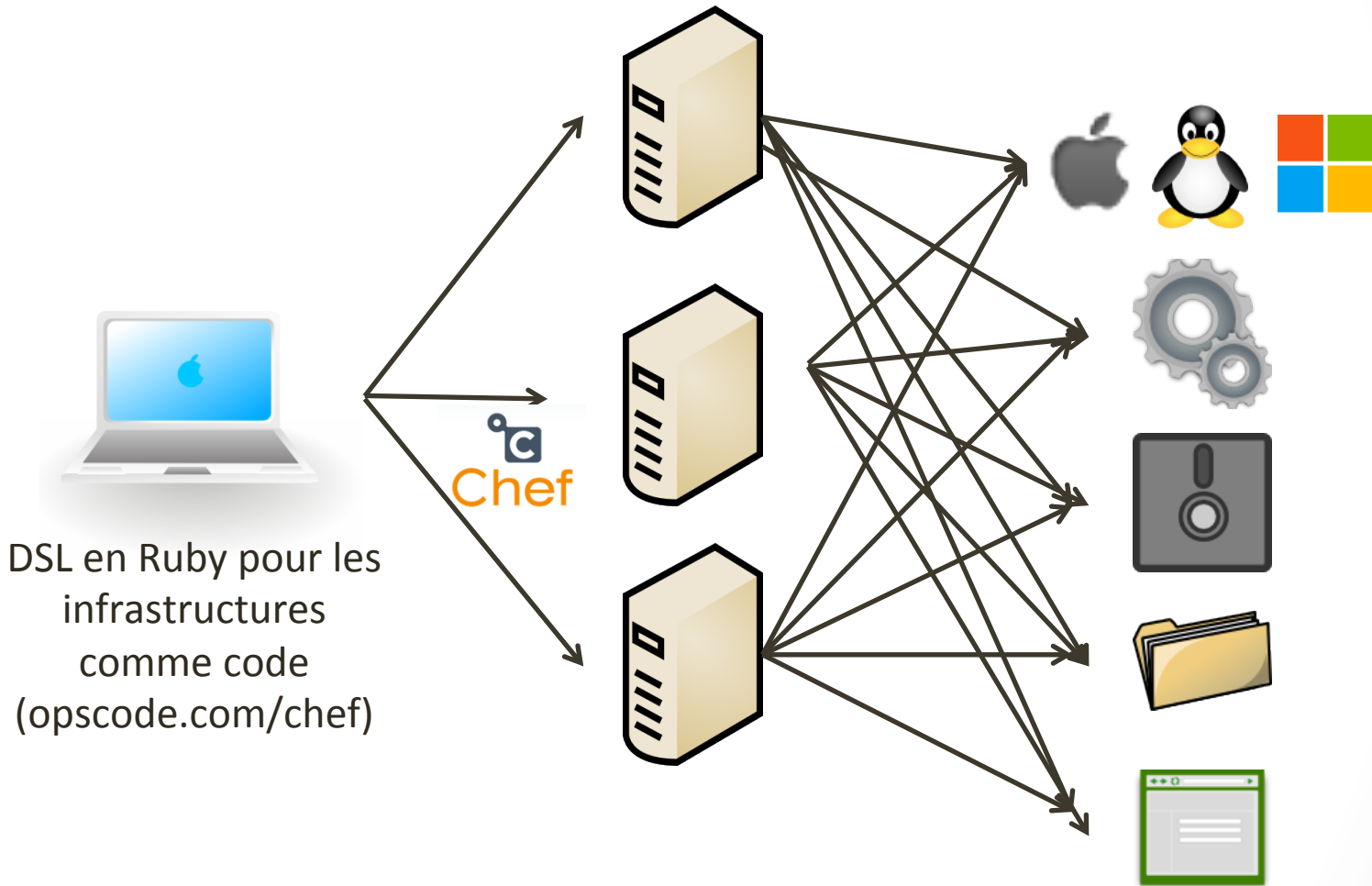
Crowbar 2.0



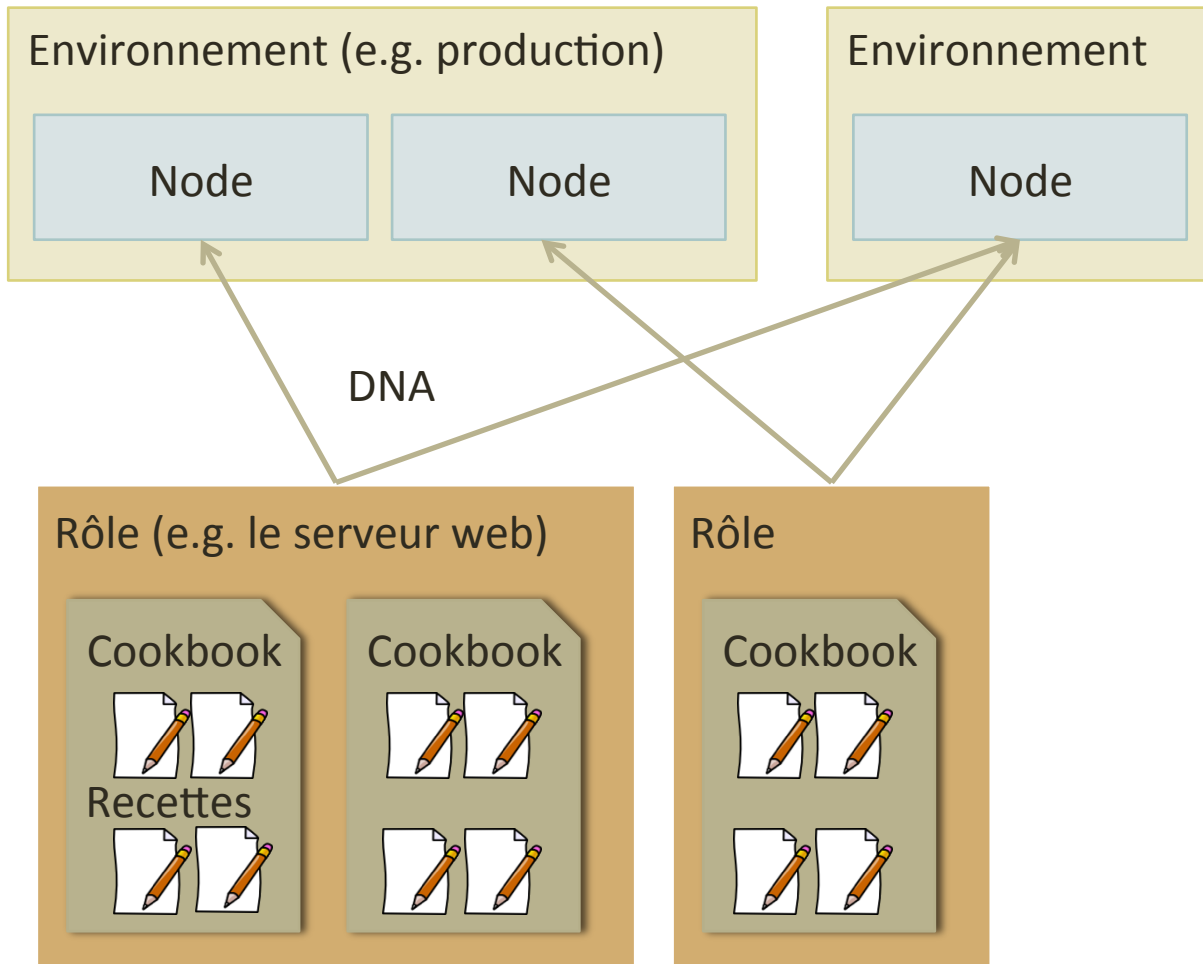
Sprinkle



Intro à Chef



Architecture de Chef



Structures de fichiers de Chef

▼ cookbooks

▼ helloworld

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

CHANGELOG.md

metadata.rb

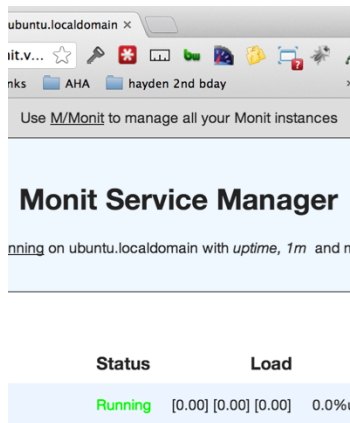
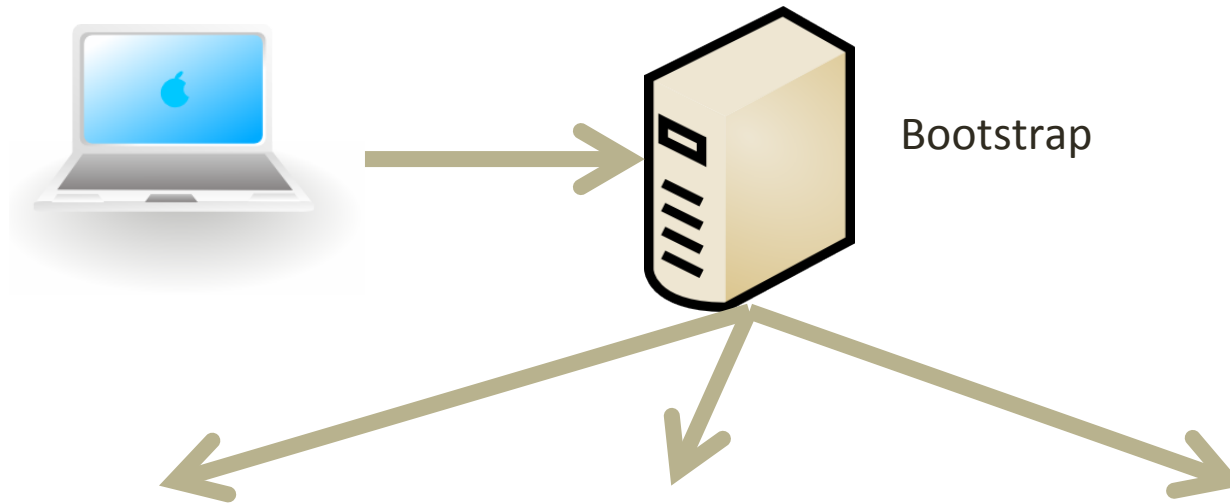
README.md

```
{  
  "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]" ]  
}
```

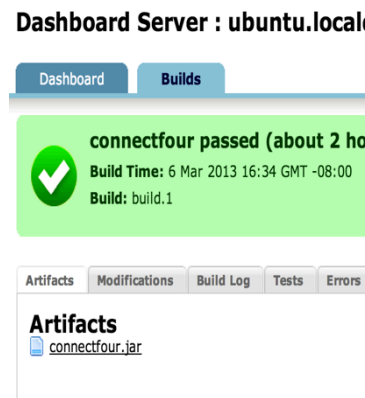
DNA

Ressources

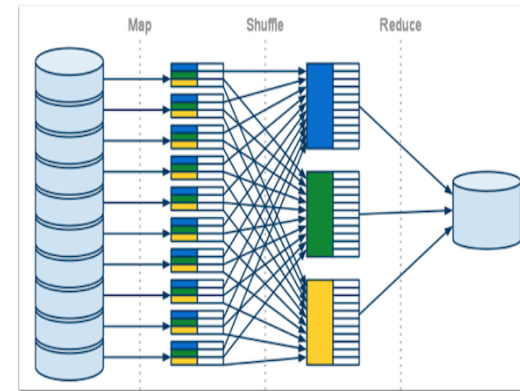
Exemples avec Chef



Monit

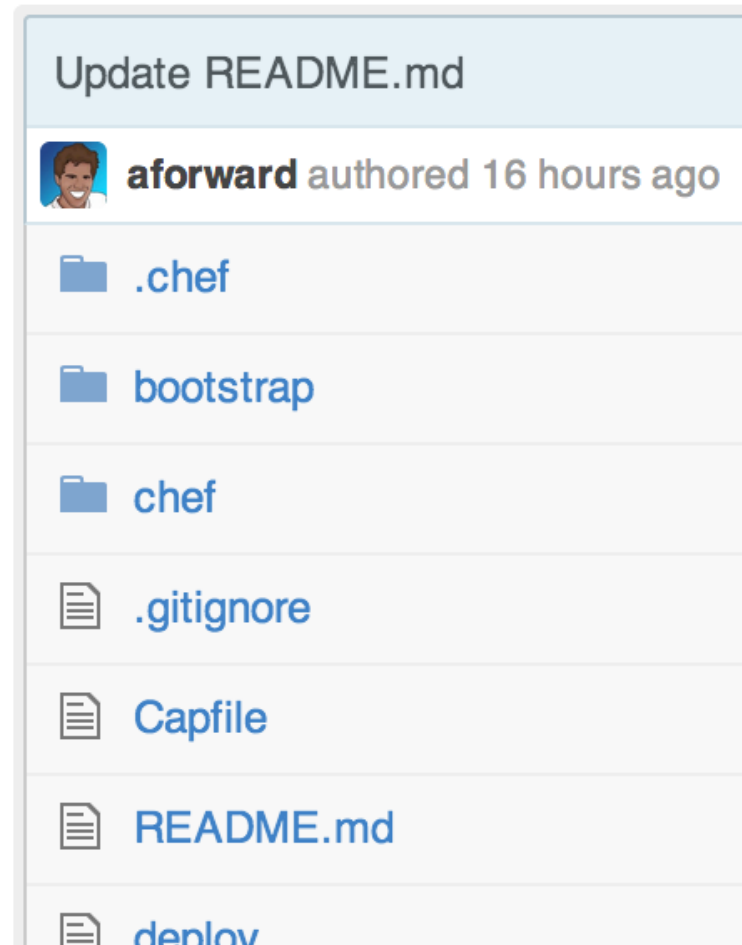
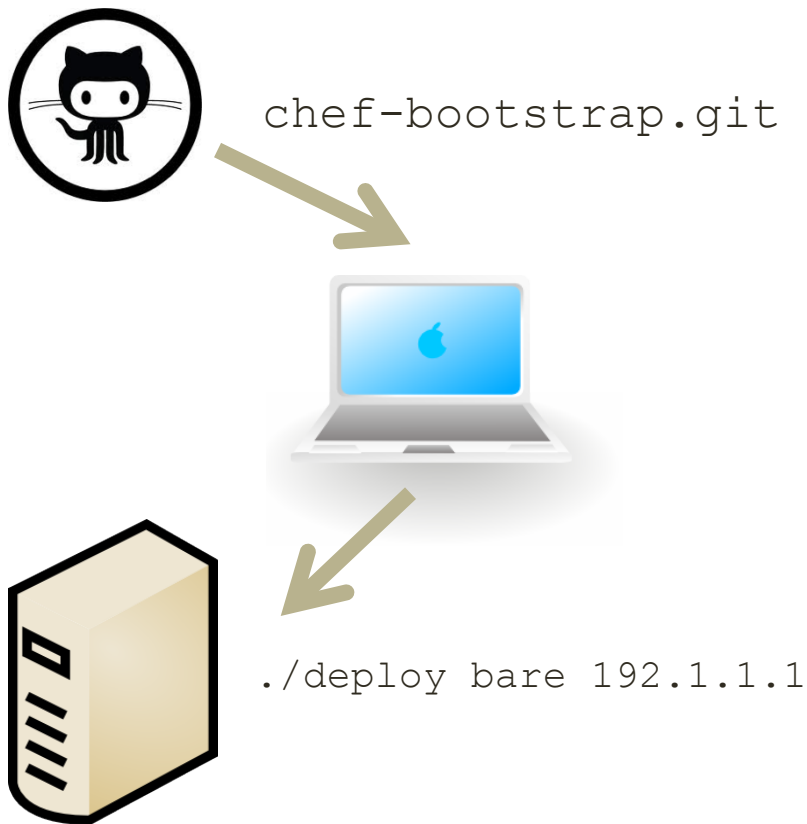


CruiseControl



Map Reduce

Examples: Bootstrapping Chef



<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 zxfv #{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
```

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
```

Examples: Monit

ubuntu.localdomain x

nit.v... ☆ 🔑 ❄️ 📊 bw 📡 📁 ? 🐛

inks 📁 AHA 📁 hayden 2nd bday >

Use [M/Monit](#) to manage all your Monit instances

Monit Service Manager

unning on ubuntu.localdomain with *uptime*, *1m* and n

Status	Load
Running	[0.00] [0.00] [0.00] 0.0%

Update README.md

 **aforward** authored 24 minutes ago

📁 attributes	33 minutes ago
📁 definitions	11 months ago
📁 files	33 minutes ago
📁 recipes	33 minutes ago
📁 templates	33 minutes ago
📄 Gemfile	a month ago
📄 Gemfile.lock	a month ago
📄 LICENSE	2 years ago
📄 README.md	24 minutes ago
📄 README.rdoc	2 years ago

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

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

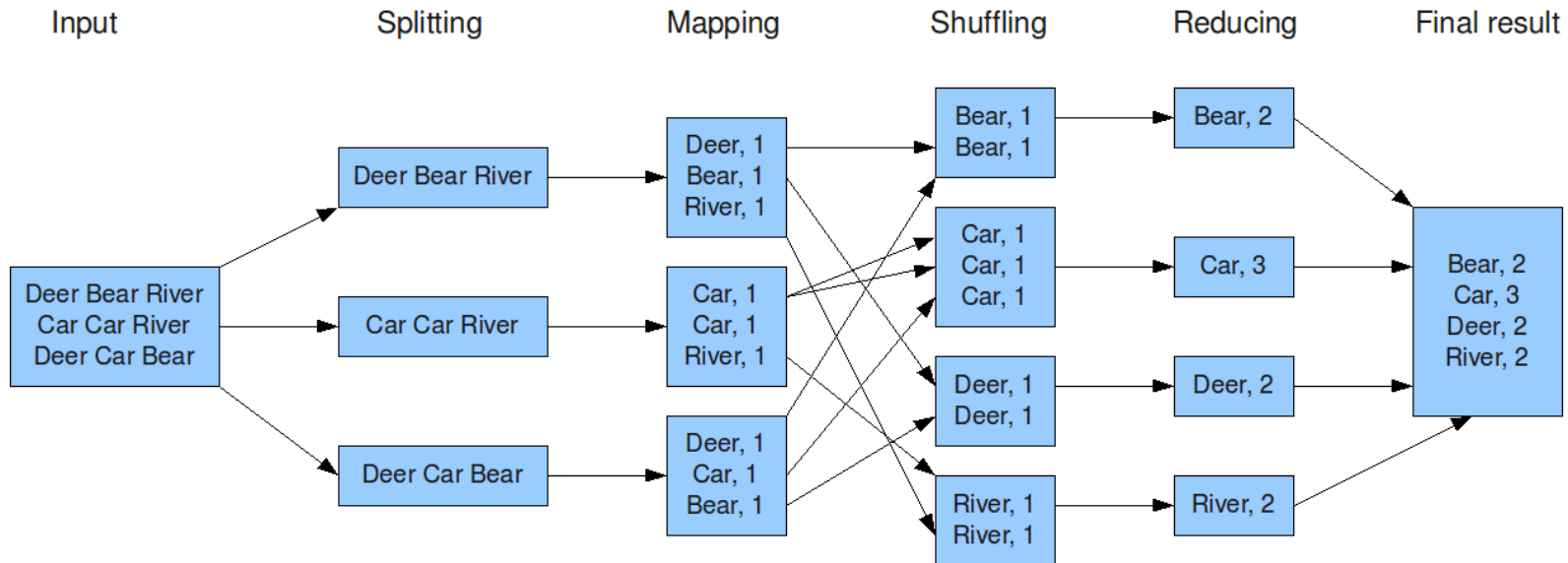
<https://github.com/aforward/chef-cruisecontrol>

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

Map, Shuffle, Reduce

The overall MapReduce word count process



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)

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

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

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)

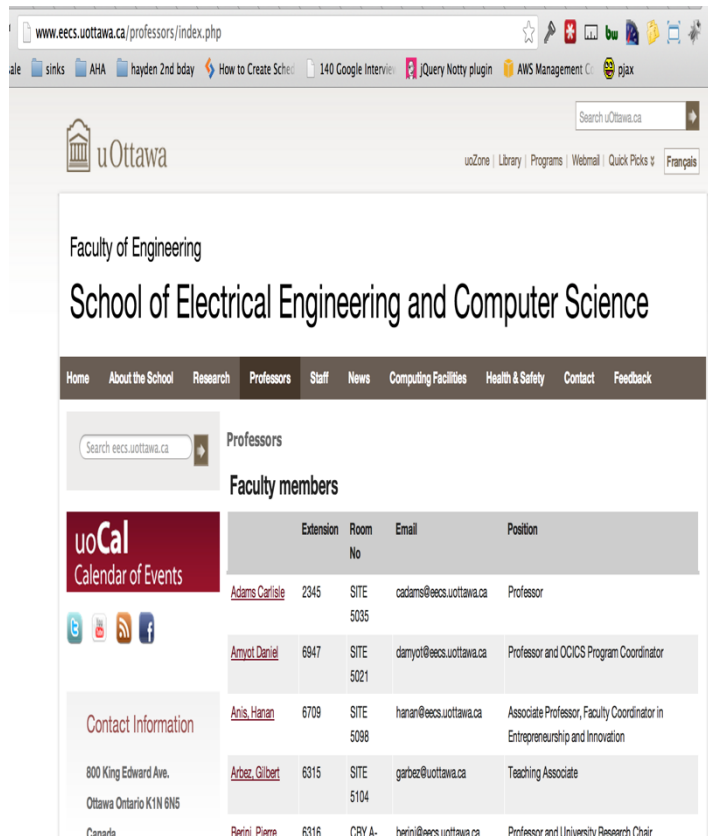
- So, if A visits T's research profile, he would see that they have both collaborated with O and J

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

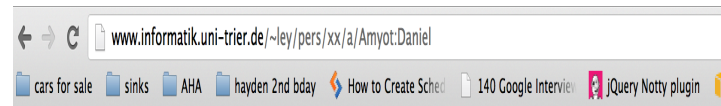
```
iex(3)> HashDict.fetch(c,"Daniel Amyot")
{:ok,["Saeed Ahmadi Behnam","Daniel Amyot","Hanane Be
eiss","Jo&#227;o Ara&#250;jo 0001","Ana Moreira","Jas
","Edna Braun","Nick Cartwright","Mira Vrbaski","Dori
niele Barone","Lei Jiang","Eric S. K. Yu","St&#233;ph
eddine Hassine","Marconi Lanna","S&#233;bastien Mosse
i","Liam Peyton","Flavio Rizzolo","Alberto Siena","An
"Dragan 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)>
```

MapReduce in Elixir



The screenshot shows the website of the Faculty of Engineering, School of Electrical Engineering and Computer Science at the University of Ottawa. The page includes a search bar, navigation links (Home, About the School, Research, Professors, Staff, News, Computing Facilities, Health & Safety, Contact, Feedback), and a section for Professors. The Professors section lists several faculty members with their contact information.

Extension	Room No	Email	Position
Adams, Carlisle	2345 SITE 5035	cadams@eecs.uottawa.ca	Professor
Amyot, Daniel	6947 SITE 5021	damyot@eecs.uottawa.ca	Professor and OCICS Program Coordinator
Anis, Hanan	6709 SITE 5098	hanan@eecs.uottawa.ca	Associate Professor, Faculty Coordinator in Entrepreneurship and Innovation
Arbez, Gilbert	6315 SITE 5104	garbez@uottawa.ca	Teaching Associate
Barin, Pierre	6316 CRY A-	barin@eecs.uottawa.ca	Professor and University Research Chair



The screenshot shows a web browser displaying an XML file. The address bar shows the URL: www.informatik.uni-trier.de/~ley/pers/xx/a/Amyot:Daniel. The page content shows a message: "This XML file does not appear to have any style information associated with it. The document tree is shown below."

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<?xml version="1.0" encoding="UTF-8" ?>
<dblp person name="Daniel Amyot" n="95">
  <person key="homepages/a/DanielAmyot" mdate="2004-11-16">
    <author>Daniel Amyot</author>
    <url>http://www.site.uottawa.ca/~damyot/</url>
  </person>
</dblp>
<?xml version="1.0" encoding="UTF-8" ?>
<article key="journals/ijeb/BehnamA13" mdate="2013-03-05">
  <author>Saeed Ahmadi Behnam</author>
  <author>Daniel Amyot</author>
  <title>
    Evolution mechanisms for goal-driven pattern families used in business process modelling.
  </title>
  <pages>254-291</pages>
  <year>2013</year>
  <volume>10</volume>
  <journal>IJEB</journal>
  <number>3</number>
  <ee>http://dx.doi.org/10.1504/IJEB.2013.051417</ee>
  <url>db/journals/ijeb/ijeb10.html#BehnamA13</url>
</article>
<?xml version="1.0" encoding="UTF-8" ?>
<article key="journals/jsw/BechaA12" mdate="2012-04-26">
  <author>Hanane Becha</author>
```

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

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

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

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

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

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;j
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",...]}
```

Les possibilités de recherche

Research avenues

Questions?

Dr. Andrew Forward

forward@gmail.com / @a4word

Le 19 juin, 2013

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 service (FSaaS), probably most known in "Chaos Monkey" [4] a product open sourced by Netflix. Faraz Faghri [5] developed FSaaS for Hadoop Clusters.

Related Work (cont)

- [1] Towards behavior driven operations (BDOps)
Gohil, Komal ; Alapati, Nagalakshmi ; Joglekar, 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. Dwarkadas, 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/vNvIr>, July 2010.
- [16] F. Khomh, T. Dhaliwal, 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
Spinellis, D.
Software, IEEE
Volume: 29 , Issue: 4
Digital Object Identifier: 10.1109/MS.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/HPCSim.2012.6266975
Publication Year: 2012 , Page(s): 573

More Chef Details

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



Knife Can...

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

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.

\$ ohai

```
Andrews-MacBook-Pro:~ aforward$ ohai
[2013-06-12T00:37:07-04:00] INFO: [inet6] no default interface, picking the first ipaddress
[2013-06-12T00:37:07-04:00] INFO: ipaddress and ip6address 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": "i386-apple-darwin12.1.1",
      "host_cpu": "i386",
      "host_os": "darwin12.1.1",
      "host_vendor": "apple",
      "bin_dir": "/Users/aforward/.rbenv/versions/1.9.2-p290/bin",
      "ruby_bin": "/Users/aforward/.rbenv/versions/1.9.2-p290/bin/ruby",
      "gems_dir": "/Users/aforward/.rbenv/versions/1.9.2-p290/lib/ruby/gems/1.9.1",
      "gem_bin": "/Users/aforward/.rbenv/versions/1.9.2-p290/bin/gem"
    },
    "erlang": {
      "version": "5.9.2",
      "options": [
        "SMP",
        "ASYNC_THREADS",
        "HIPE"
      ],
      "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-b04, mixed mode"
      }
    },
    "nodejs": {
      "version": "0.8.17"
    },
    "perl": {
      "version": "5.12.4",
      "archname": "darwin-thread-multi-2level"
    },
    "php": {
      "version": "5.3.15",
      "builddate": "(cli) (built: Aug"
    },
    "python": {
      "version": "2.7.2",
      "builddate": "Oct 11 2012, 20:14:37"
    }
  },
  "kernel": {
```

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

Example Chef Roles

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

Example Chef Environments

- Development
- Staging
- User Acceptance
- Production

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 zxfv #{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
```

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
```

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