Configuration Management: Stability in Your Pipeline

sascha bates
sascha bates

blog.brattyredhead.com
Twin Cities Infracoders Meetup
@sascha_d
The Ship Show Podcast
Automation Consultant
sascha@opscode.com
credentials?

In love with CM since 2010

victim and perpetrator of inconsistency over the years
The Problem

Environments Run Amok
Environments Run Amok

configuration management
homogenous environments
consistent process
A tool is just a tool
it’s what you do with it that matters
Environments...
who cares?
This Never Happens

incorrect/incomplete deployment instructions

There has been a failure on the INTERNETS
This Never Happens

wrong database connection string deployed to Production
This Never Happens

smtp server fixed by hand
then forgotten

There has been a failure on the INTERNETS
This Never Happens

test apache server with special configs
This Never Happens

ssh keys pushed by hand
What Happened?
Two Failures
Two Failures

Communication
Two Failures

Communication

Ownership
Communication

people gotta talk
Communication

people gotta talk
Communication

BASIC
HUMANITY

people gotta talk
Technical Ownership

Dev and Test are the Cinderellas of Environments
Environments are complex
Environments

are unloved
Environments

become unloveable
Environments are necessary
So Many Things...

and none of them belong in your code build
So Many Things...

packages and versions

and none of them belong in your code build
So Many Things...

packages and versions
mail servers

and none of them belong in your code build
So Many Things...

packages and versions
mail servers
multiple data centers

and none of them belong in your code build
So Many Things...

packages and versions
mail servers
multiple data centers
database users and passwords

and none of them belong in your code build
So Many Things...

packages and versions
mail servers
multiple data centers
database users and passwords
connection strings

and none of them belong in your code build
So Many Things...

packages and versions
mail servers
multiple data centers
database users and passwords
connection strings
integration URLs

and none of them belong in your code build
So Many Things...

packages and versions
mail servers
multiple data centers
database users and passwords
connection strings
integration URLs
deployments

and none of them belong in your code build
So Many Things...

packages and versions
mail servers
multiple data centers
database users and passwords
connection strings
integration URLs
deployments
batch jobs

and none of them belong in your code build
Configuration Management to the rescue
Eliminate Mistakes
Ensure Consistency
Automate Complexity
Separation of Duties
Eliminate Mistakes

people out of the pipeline
Eliminate Mistakes

people out of the pipeline
Ensure Consistency

one OS build to rule them all
Ensure Consistency

easy

one OS build to rule them all
Ensure Consistency

easy
fast

one OS build to rule them all
Ensure Consistency

easy
fast
reliable

one OS build to rule them all
Ensure Consistency

easy
fast
reliable
server builds

one OS build to rule them all
Ensure Consistency

yum install tomcat

or

download random tomcat; install
Ensure Consistency

yum install tomcat

or

download random tomcat; install

Packages
Ensure Consistency

yum install tomcat

or

download random tomcat, install

Wrong Packages
Ensure Consistency

Packages

```ruby
package 'apache' do
  version '2.2.9'
  action :install
end
```

Packages
Ensure Consistency

```ruby
package 'apache' do
  version node['apache']['version']
  action :install
end
```

Packages
Ensure Consistency

locate environment-specific configurations
outside the codebase

Central Configurations
Automate Complexity

your 5 year old bash scripts not cutting it
Automate Complexity

I can see no way in which this carefully laid plan could ever fail.

stop running around to multiple servers
Automate Complexity

app version 1.2.2

bump 1.3.0

trigger deployment actions across multiple tiers

stop running around to multiple servers
Separation of Duties

What bird?
confidence