How Mozilla draws the line between configuration and data while using configuration management tools

About the authors

Brian Hourigan

bhourigan@mozilla.com

Justin Dow

jdow@mozilla.com

I want to tell you a story…

I want to tell you a story…

A 4GB .iso file was checked into our CM VCS

I want to tell you a story…

A 4GB .iso file was checked into our CM VCS

China puppet master updates from svn on a regular basis

I want to tell you a story…

A 4GB .iso file was checked into our CM VCS

China puppet master updates from svn on a regular basis

Our data center in China isn’t known for it’s robust internet connectivity

I want to tell you a story…

A 4GB .iso file was checked into our CM VCS

China puppet master updates from svn on a regular basis

Our data center in China isn’t known for it’s robust internet connectivity

E-mail based commit reports were… large

I want to tell you a story…

A 4GB .iso file was checked into our CM VCS

China puppet master updates from svn on a regular basis

Our data center in China isn’t known for it’s robust internet connectivity

E-mail based commit reports were… large

Puppet was down in China for a week

That was a catalyst for the separation of configuration and data

What else doesn’t belong in a VCS?

Can we define what belongs in a VCS?

Data classifications

Data classifications

Configuration data

Data classifications

Configuration data

Application data

Characteristics of configuration data

Characteristics of configuration data

Human readable, often line-oriented ASCII

Characteristics of configuration data

Human readable, often line-oriented ASCII

Human writable

Characteristics of configuration data

Human readable, often line-oriented ASCII

Human writable

Typically small file sizes

Characteristics of configuration data

Human readable, often line-oriented ASCII

Human writable

Typically small file sizes

Hopefully, there are comments

Characteristics of application data

Characteristics of application data

Binary

Characteristics of application data

Binary

Machine readable

Characteristics of application data

Binary

Machine readable

Machine writable

Characteristics of application data

Binary

Machine readable

Machine writable

Typically large files

That’s it. Any questions?

Just because you can do it
doesn’t mean you should

It’s healthy to push the limits of technology.. within reason

Let’s take a look into our storage options

Version control systems

Version control systems

Great for configuration data

Version control systems

Great for configuration data

Geared towards a collaborative environment

Version control systems

Great for configuration data

Geared towards a collaborative environment

Often not ideal for binaries (more in a second)

Version control systems

Great for configuration data

Geared towards a collaborative environment

Often not ideal for binaries (more in a second)

Replication is easy

File system based storage

Great for binaries and large files

File system based storage

Great for binaries and large files

Versioning can be hard

File system based storage

Great for binaries and large files

Versioning can be hard

Replication can be hard

File system based storage

Great for binaries and large files

Versioning is hard

Replication is hard

Lots of available tools

File system based storage

Great for binaries and large files

Versioning can be hard

Replication can be hard

Lots of available tools

How Firefox builds are stored

Continuous integration workflow

Developer commits code to hg.mozilla.org

Continuous integration workflow

Developer commits code to hg.mozilla.org

Build bots check out the code, build it, and run automated tests

Continuous integration workflow

Developer commits code to hg.mozilla.org

Build bots check out the code, build it, and run automated tests

Resulting binaries and test data is uploaded to ftp.mozilla.org

Continuous integration workflow

Developer commits code to hg.mozilla.org

Build bots check out the code, build it, and run automated tests

Resulting binaries and test data is uploaded to ftp.mozilla.org

We don’t bother to version builds outside of regular filesystem snapshots

Referencing external data sources using configuration management

```
vcsrepo { "${repo_root}/…":
  ensure   => present,
  provider => "git",
  source   => "git://github.com/mozilla/…",
}
```

Distributing binary data outside VCS

Package data and depend on package managers

Distributing binary data outside VCS

Package data and depend on package managers

Use external tools such as wget, rsync, and lsync

Distributing binary data outside VCS

Package data and depend on package managers

Use external tools such as wget, rsync, and lsync

Distributed file systems

Distributing binary data outside VCS

Package data and depend on package managers

Use external tools such as wget, rsync, and lsync

Distributed file systems

NFS

Distributing binary data outside VCS

Package data and depend on package managers

Use external tools such as wget, rsync, and lsync

Distributed file systems

NFS (nasty)

Some tools that may work for you

Git annex

Some tools that may work for you

Git annex

Git media

Some tools that may work for you

Git annex

Git media

Gitattributes

Some tools that may work for you

Git annex

Git media

Gitattributes

Boar

Questions?

https://github.com/bhourigan/lca2013/