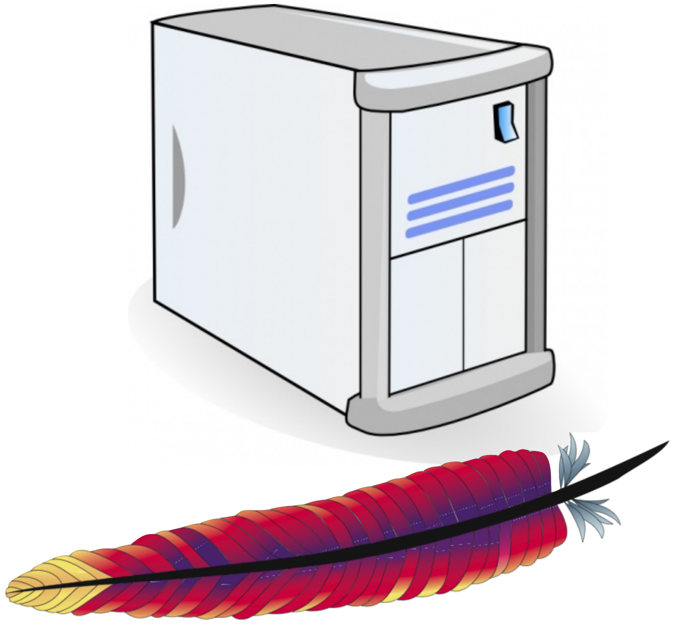
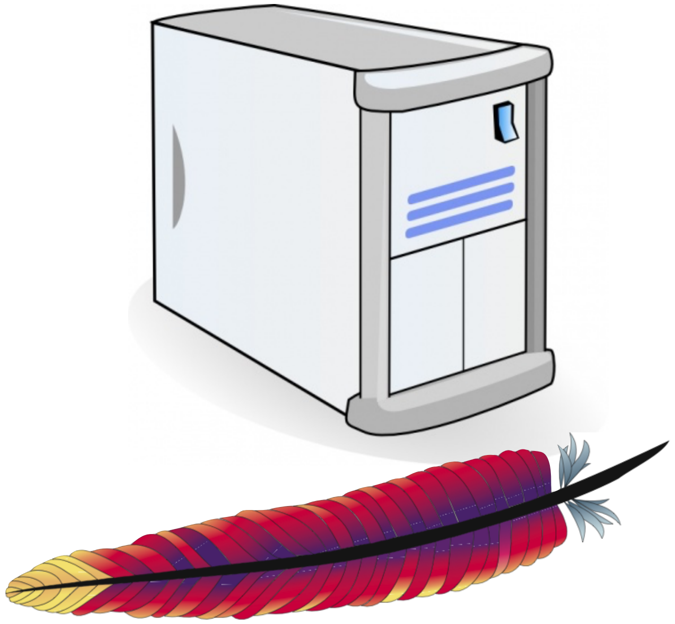
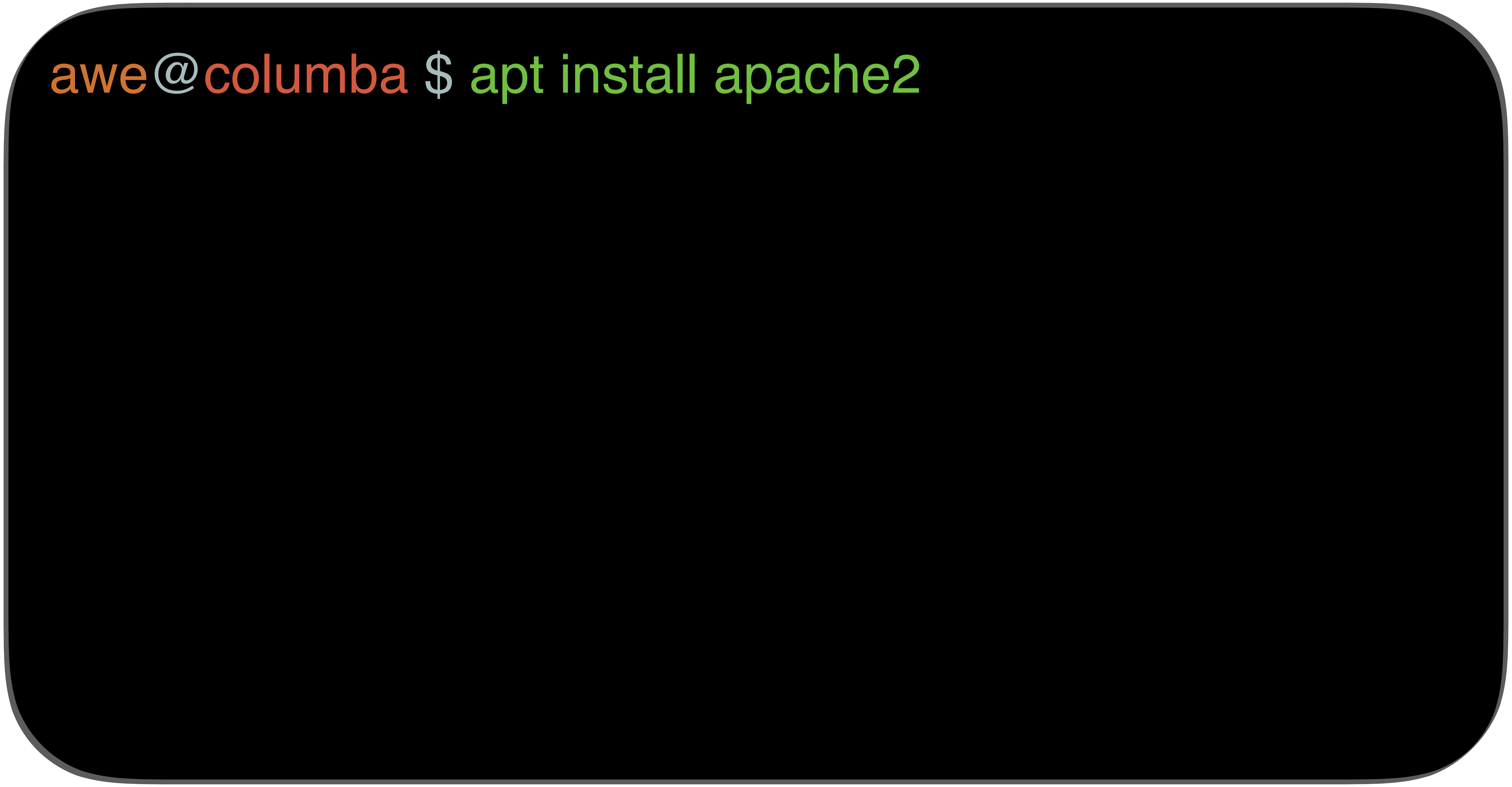
TORTOISE: IMPERATIVE SYSTEM CONFIGURATION REPAIR

Aaron Weiss, Arjun Guha, Yuriy Brun

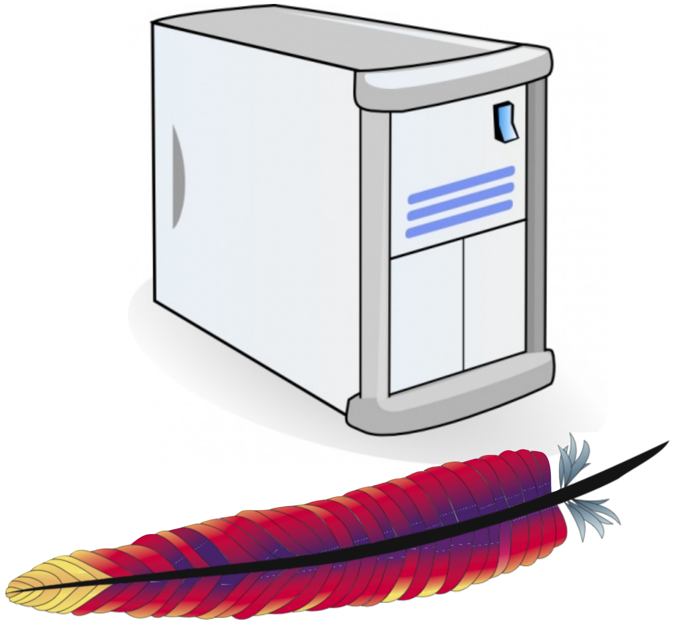
Northeastern University and University of Massachusetts

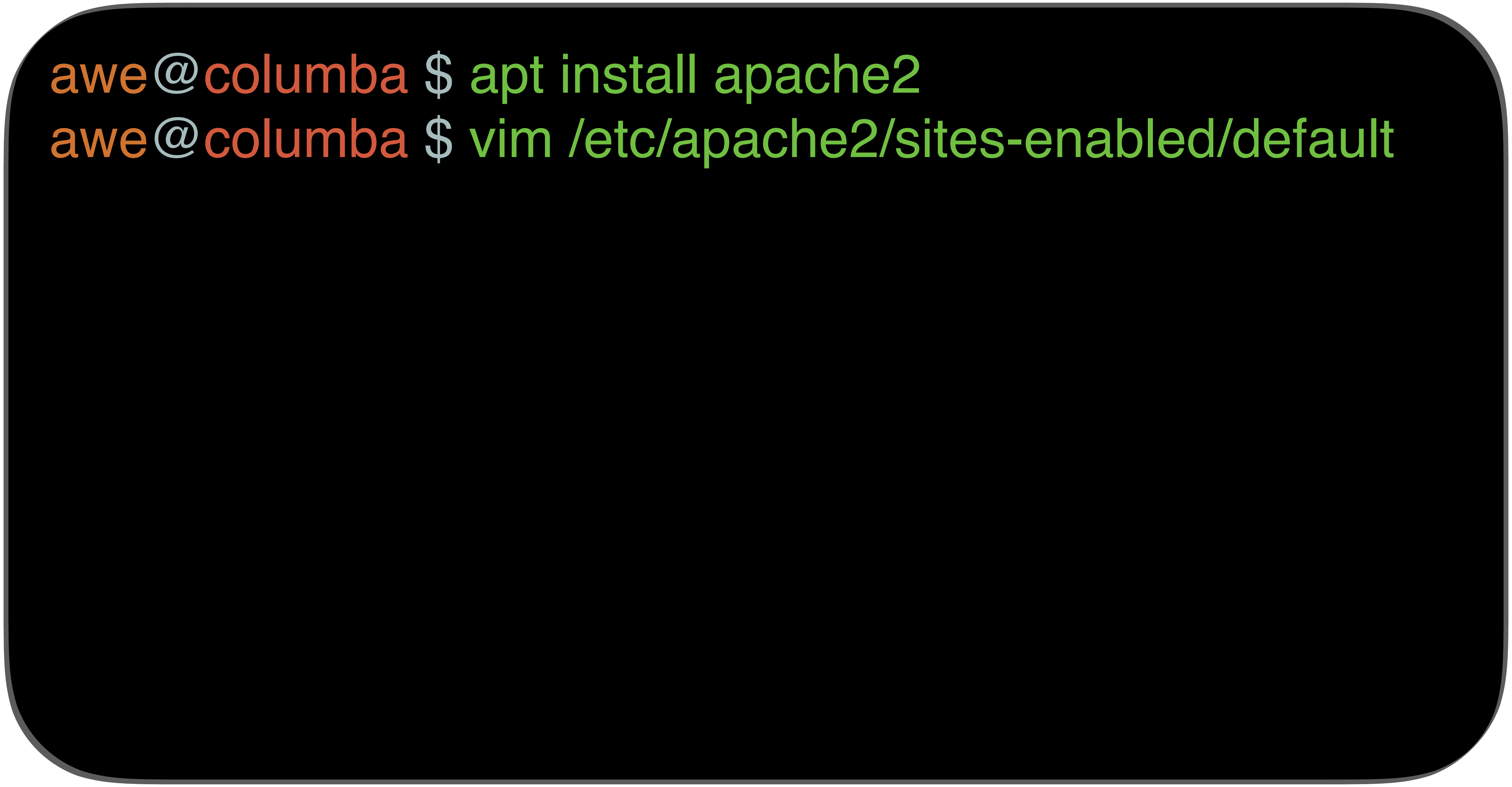




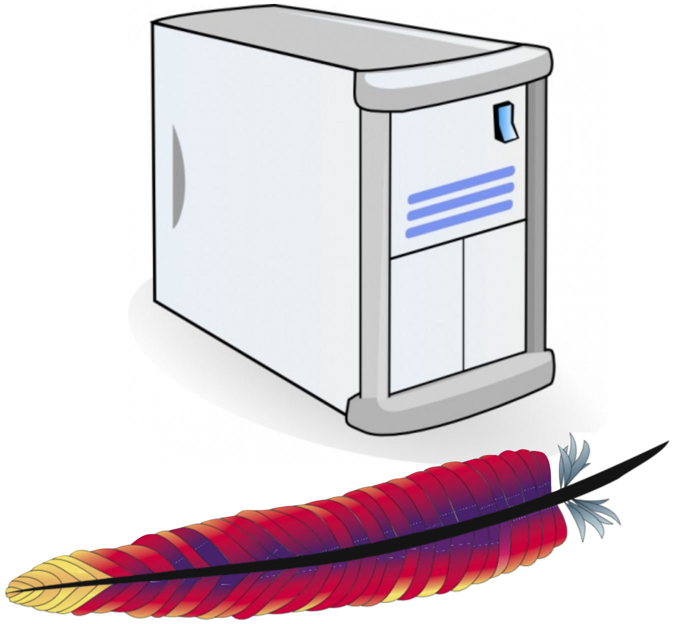


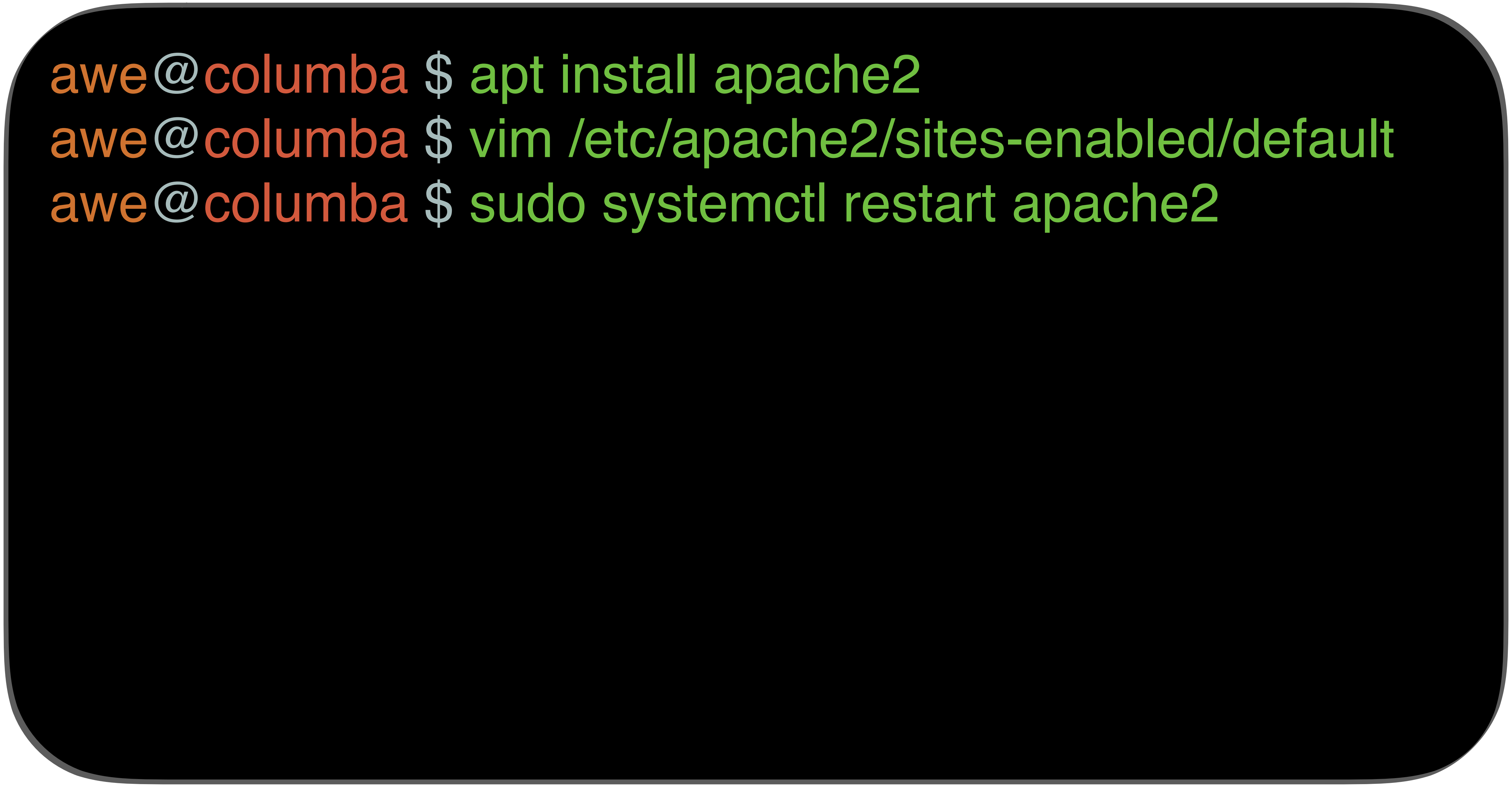
```
awe@columba $ apt install apache2
```



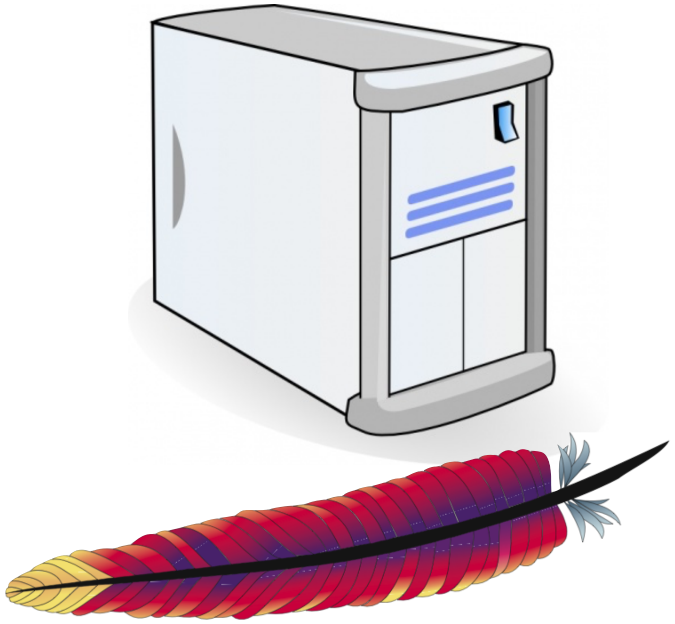


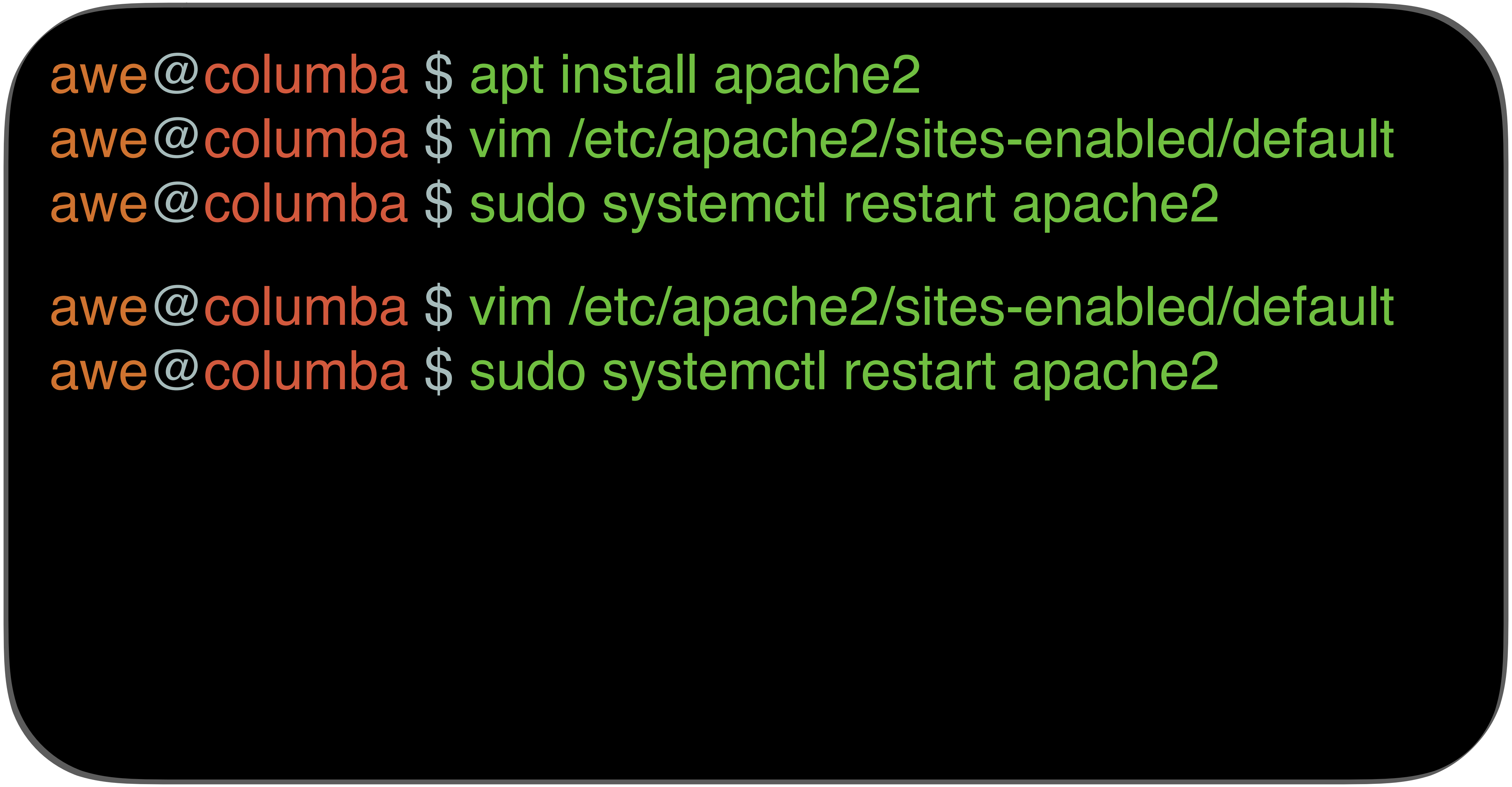
```
awe@columba $ apt install apache2
awe@columba $ vim /etc/apache2/sites-enabled/default
```



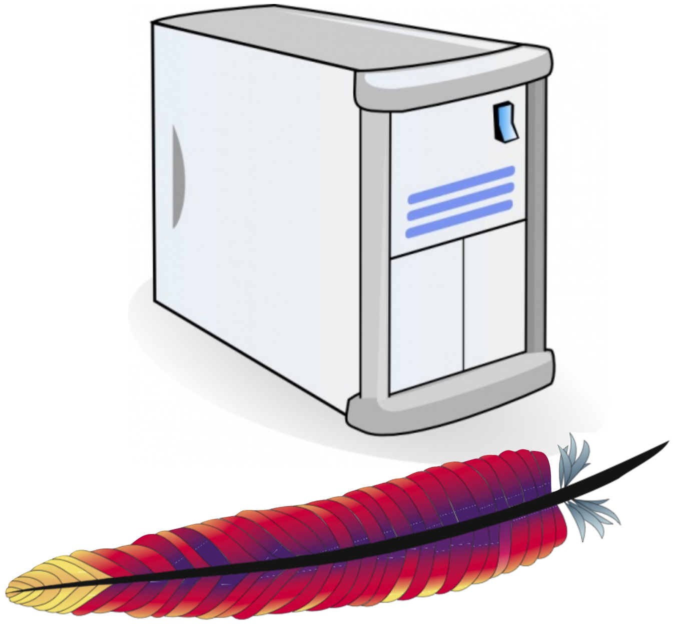


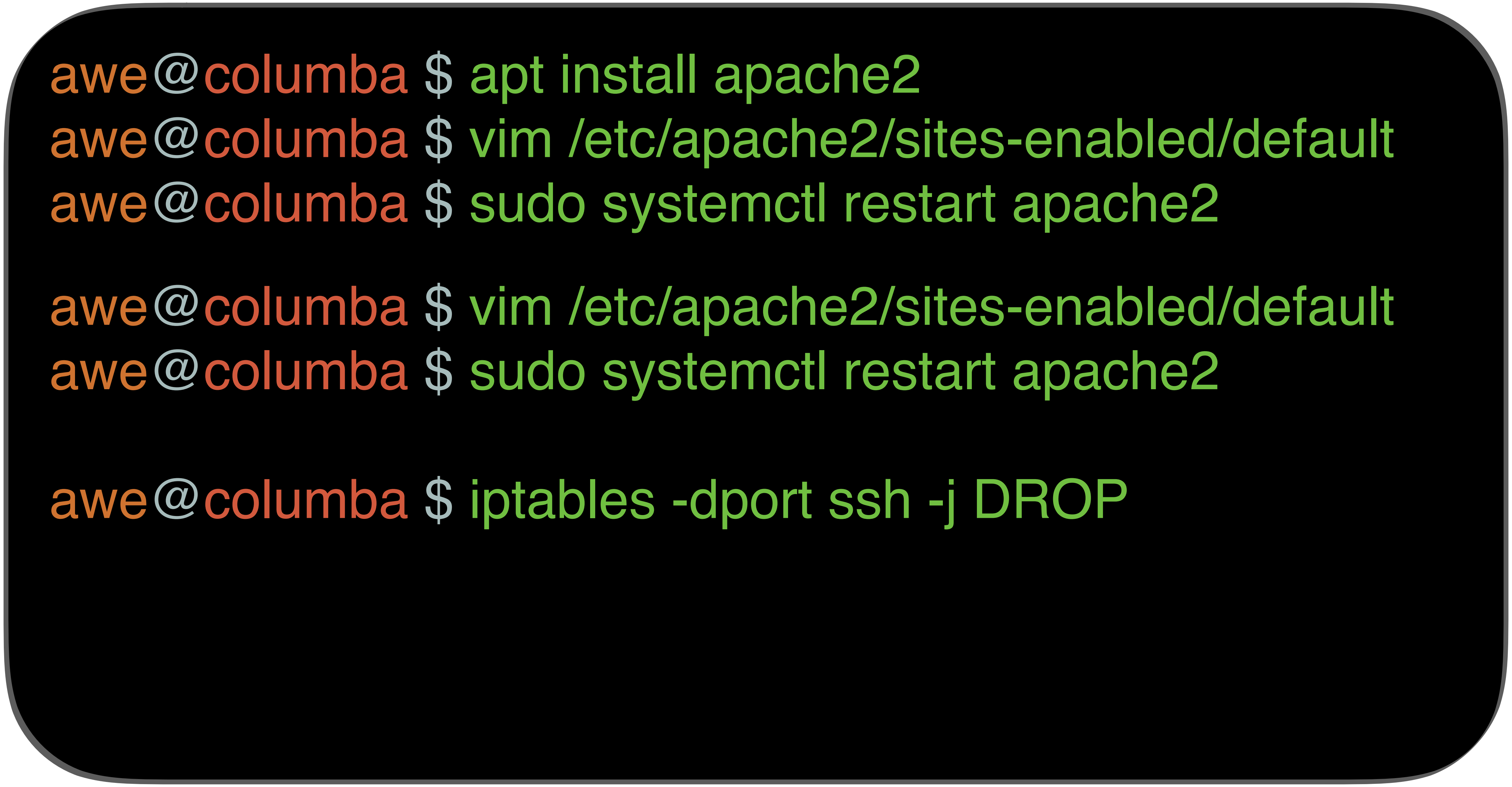
```
awe@columba $ apt install apache2
awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ sudo systemctl restart apache2
```



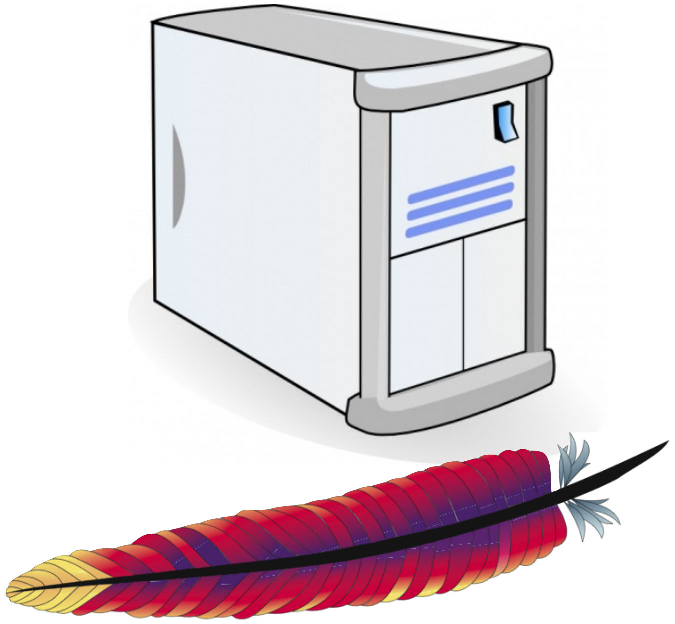


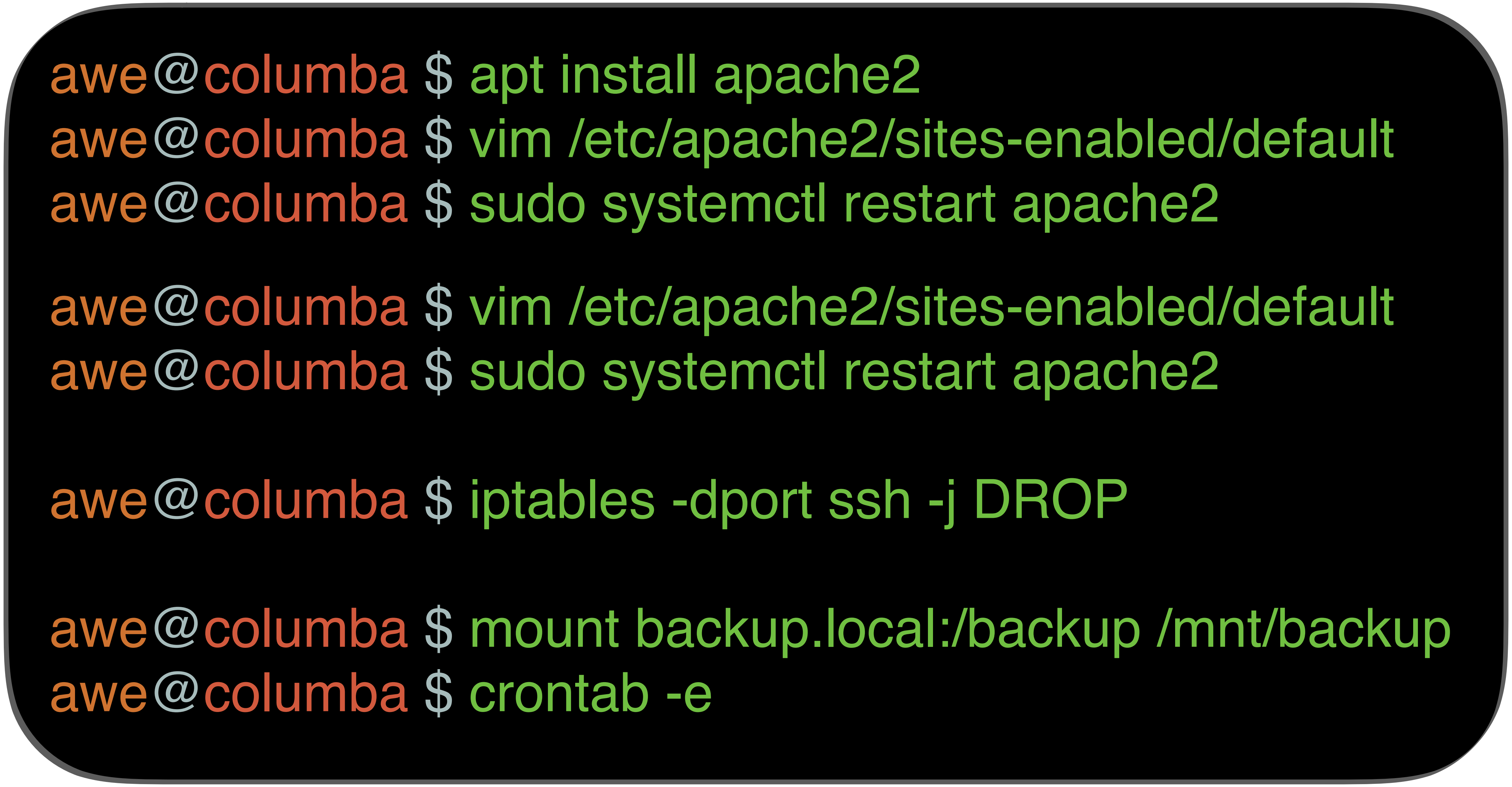
```
awe@columba $ apt install apache2
awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ sudo systemctl restart apache2

awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ sudo systemctl restart apache2
```



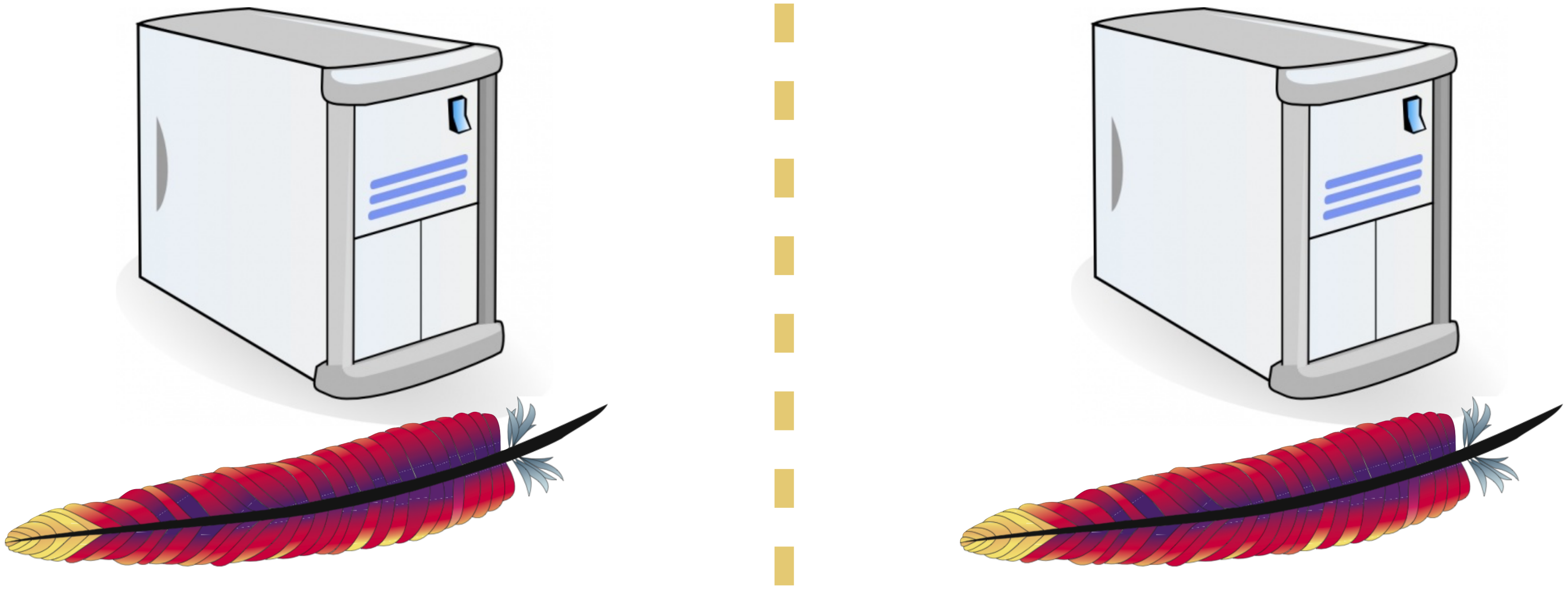


```
awe@columba $ apt install apache2
awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ sudo systemctl restart apache2

awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ sudo systemctl restart apache2


awe@columba $ iptables -dport ssh -j DROP
```

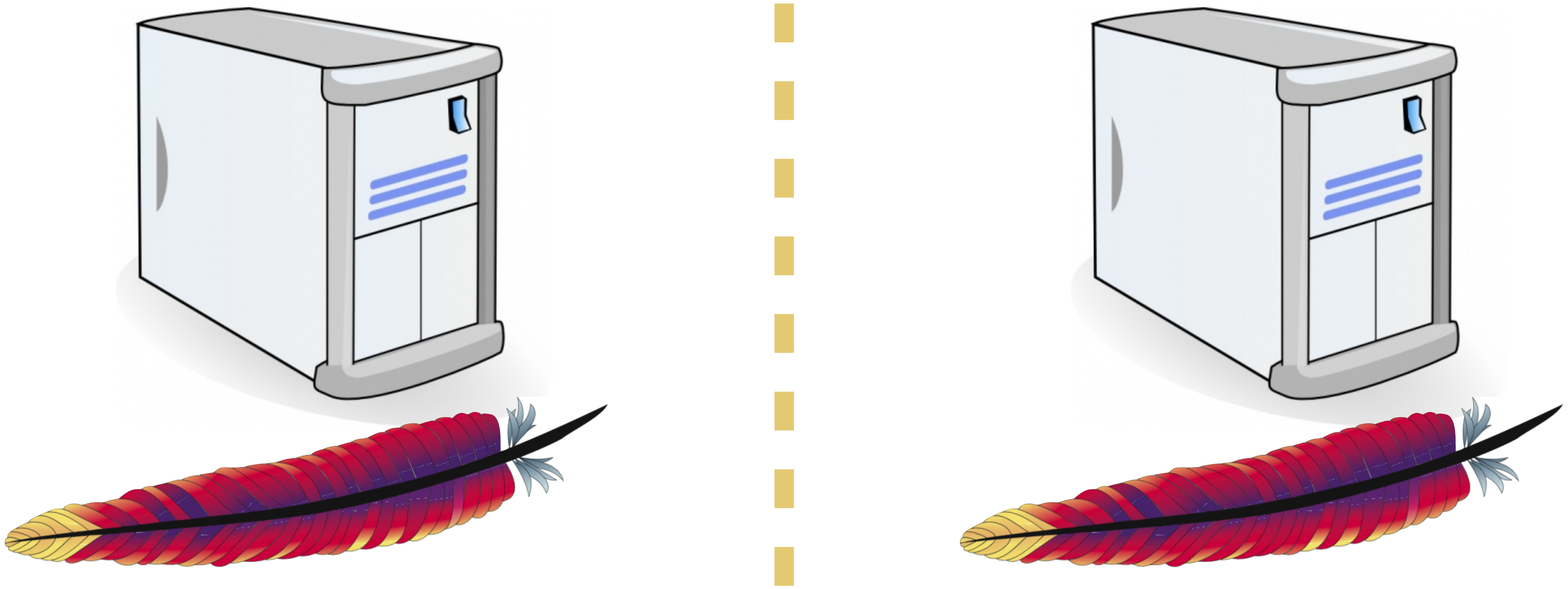


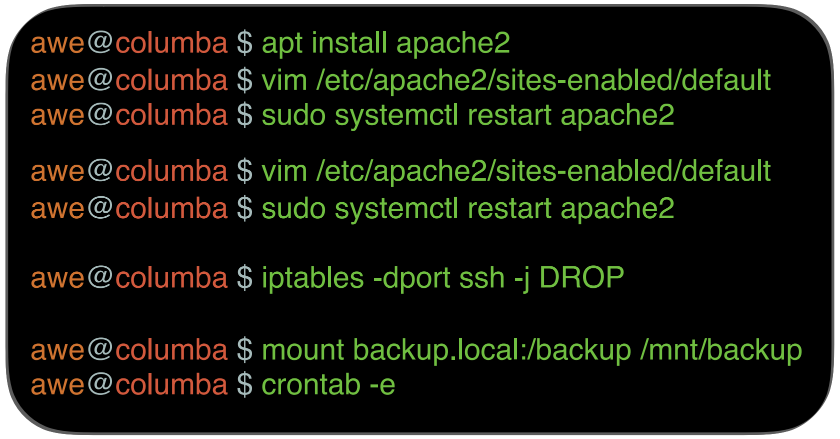


```
awe@columba $ apt install apache2
awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ sudo systemctl restart apache2

awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ sudo systemctl restart apache2


awe@columba $ iptables -dport ssh -j DROP


awe@columba $ mount backup.local:/backup /mnt/backup
awe@columba $ crontab -e
```

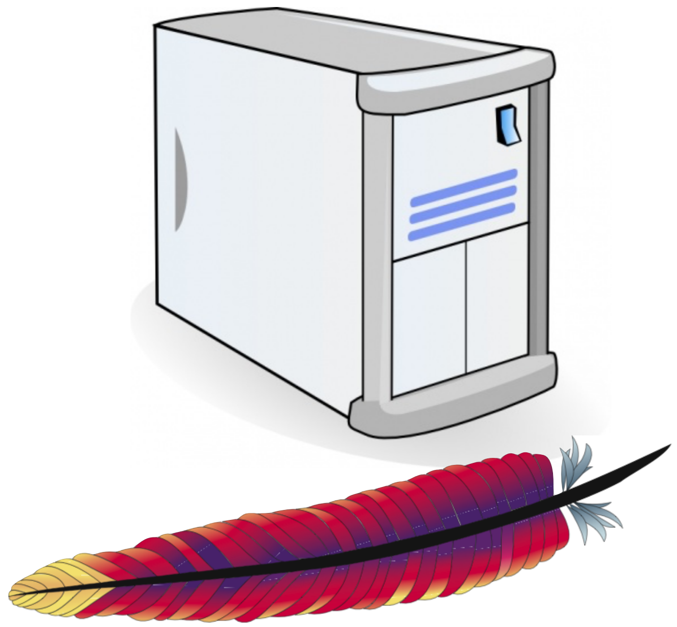


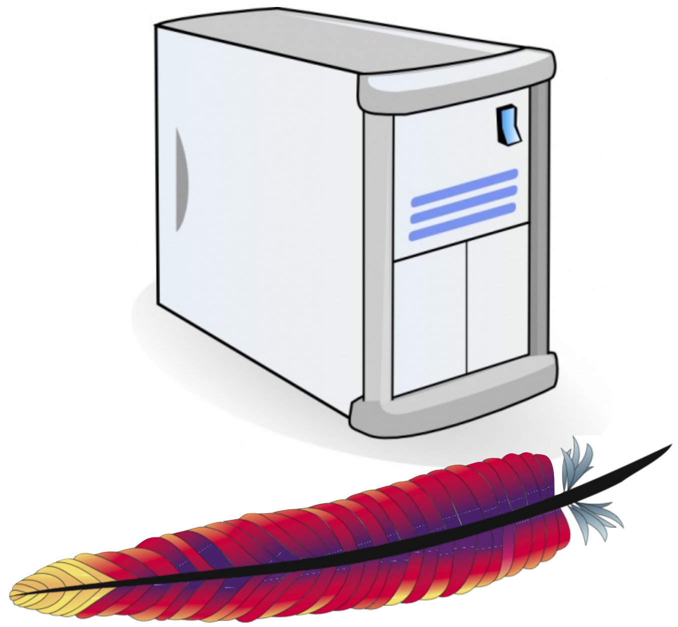
```
awe@columba $ apt install apache2
awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ sudo systemctl restart apache2

awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ sudo systemctl restart apache2


awe@columba $ iptables -dport ssh -j DROP


awe@columba $ mount backup.local:/backup /mnt/backup
awe@columba $ crontab -e
```





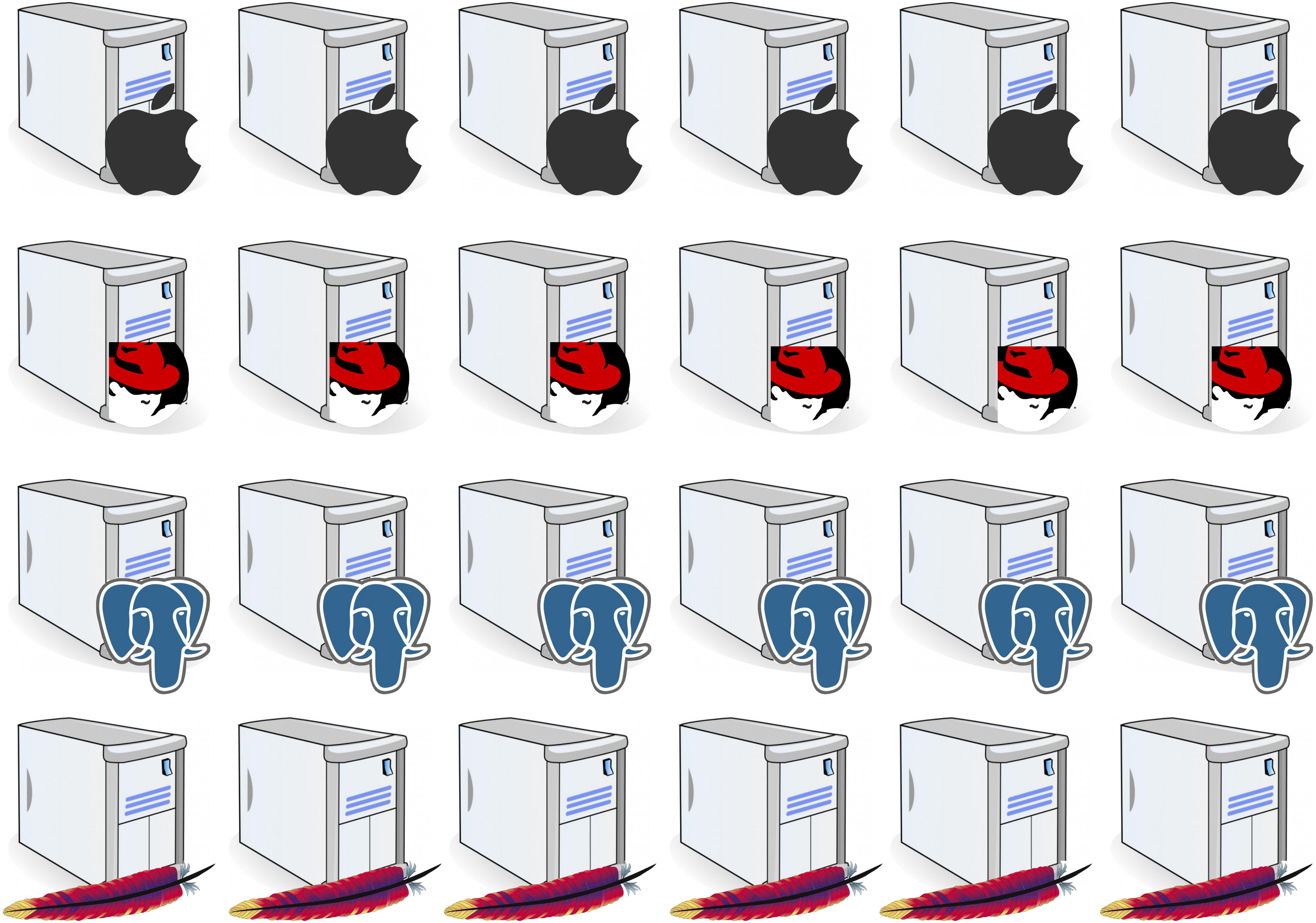
```
awe@columba $ apt install apache2
awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ sudo systemctl restart apache2

awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ sudo systemctl restart apache2

awe@columba $ iptables -dport ssh -j DROP

awe@columba $ mount backup.local:/backup /mnt/backup
awe@columba $ crontab -e
```





```
awe@columba $ apt-get install apache2
awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ sudo systemctl restart apache2

awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ sudo systemctl restart apache2

awe@columba $ iptables -dport ssh -j DROP

awe@columba $ mount backup.local:/backup /mnt/backup
awe@columba $ crontab -e
```

```
awe@columba $ apt install apache2
awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ sudo systemctl restart apache2

awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ sudo systemctl restart apache2

awe@columba $ iptables -dport ssh -j DROP

awe@columba $ mount backup.local:/backup /mnt/backup
awe@columba $ crontab -e
```



CONFIGURATION MANAGEMENT TOOLS

puppet

ANSIBLE

CHEF™

SALTSTACK

CFEngine

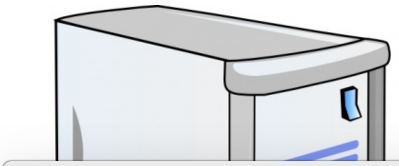
LCFG Large Scale UNIX Configuration System

```
package {"apache2": ensure => present }
service {"apache2": ensure => running }
define website($title, $root) {
  file {"/etc/apache2/sites-enabled/$title.conf":
    content => "<VirtualHost $title:80>
    DocumentRoot /var/sites/$root
    </VirtualHost>" }
  file {"/var/sites/$root":
    ensure => directory,
    source => "puppet://sites/$root",
    owner => "root",
    mode => 0700,
    recurse => "remote"}
}
website {"aaronweiss.us": root => "aaronweiss" }
website {"pdgn.co": root => "pdgn" }
```

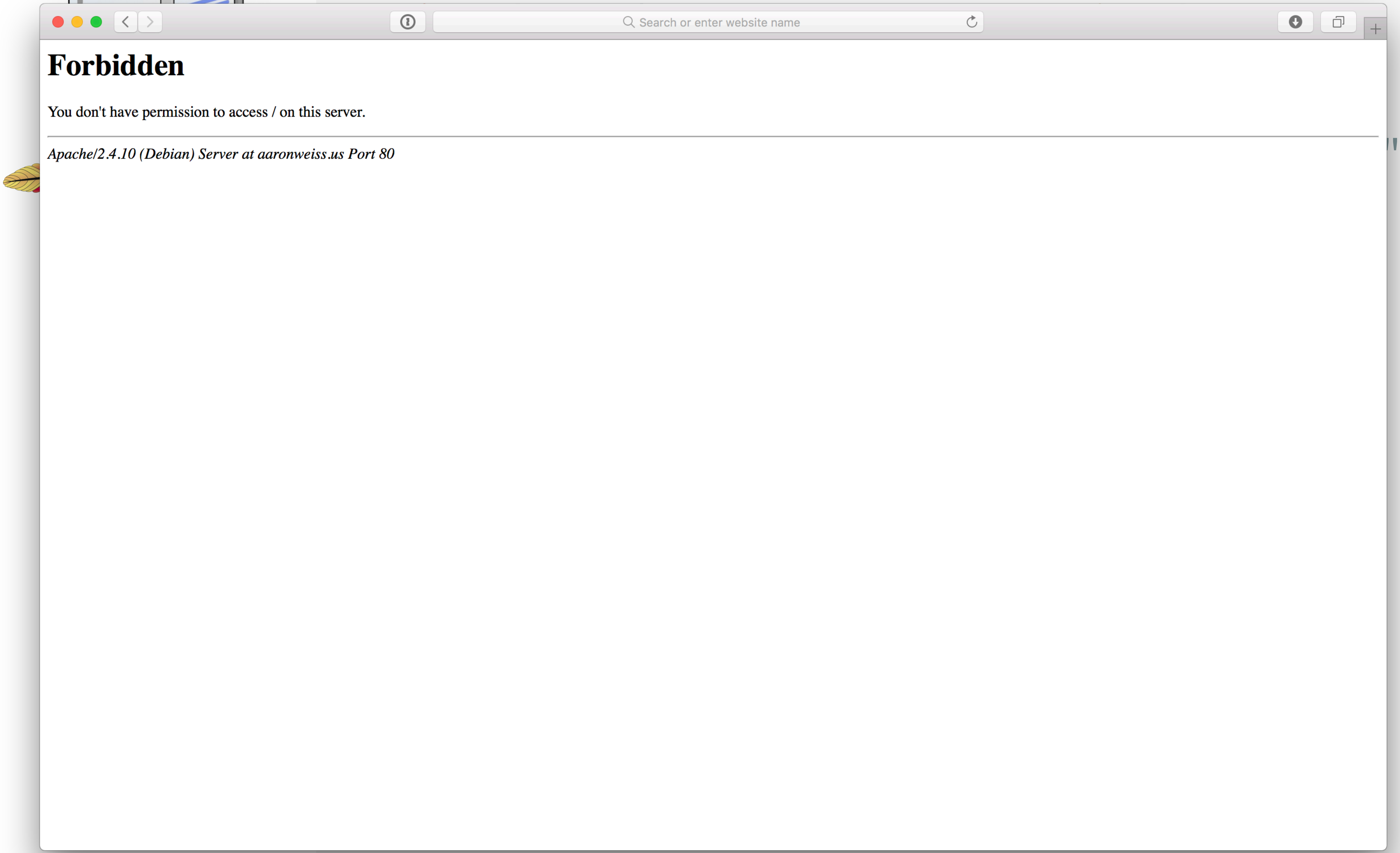
```
package {"apache2": ensure => present }
service {"apache2": ensure => running }
define website($title, $root) {
  file {"/etc/apache2/sites-enabled/$title.conf":
    content => "<VirtualHost $title:80>
    DocumentRoot /var/sites/$root
    </VirtualHost>" }
  file {"/var/sites/$root":
    ensure => directory,
    source => "puppet://sites/$root",
    owner => "root",
    mode => 0700,
    recurse => "remote"}
}
website {"aaronweiss.us": root => "aaronweiss" }
website {"pdgn.co": root => "pdgn" }
```

CONFIGURATION MANAGEMENT ISN'T PERFECT

```
package {"apache2": ensure => present }
service {"apache2": ensure => running }
define website($title, $root) {
  file {"/etc/apache2/sites-enabled/$title.conf":
    content => "<VirtualHost $title:80>
    DocumentRoot /var/sites/$root
    </VirtualHost>" }
  file {"/var/sites/$root":
    ensure => directory,
    source => "puppet://sites/$root",
    owner => "root",
    mode => 0700,
    recurse => "remote" }
}
website {"aaronweiss.us": root => "aaronweiss" }
website {"pdgn.co": root => "pdgn" }
```



```
package {"apache2": ensure => present }
```



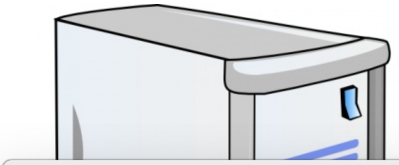
Forbidden

You don't have permission to access / on this server.

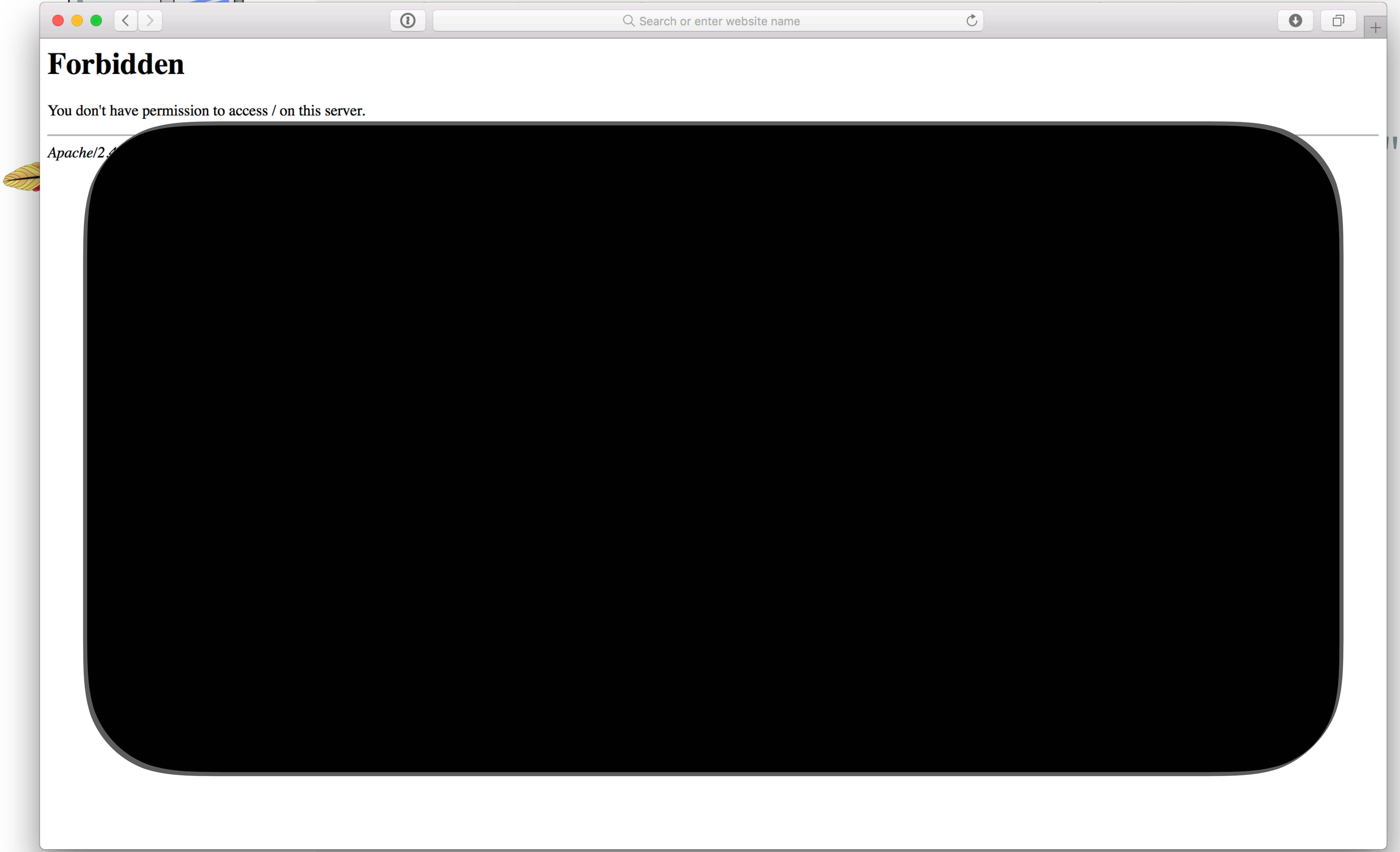
Apache/2.4.10 (Debian) Server at aaronweiss.us Port 80

```
":
```

```
website {"aaronweiss.us": root => "aaronweiss" }

website {"pdgn.co": root => "pdgn" }
```



```
package {"apache2": ensure => present }
```



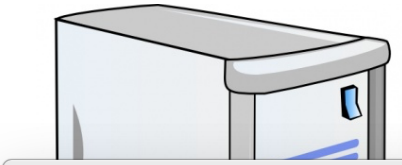
Forbidden

You don't have permission to access / on this server.

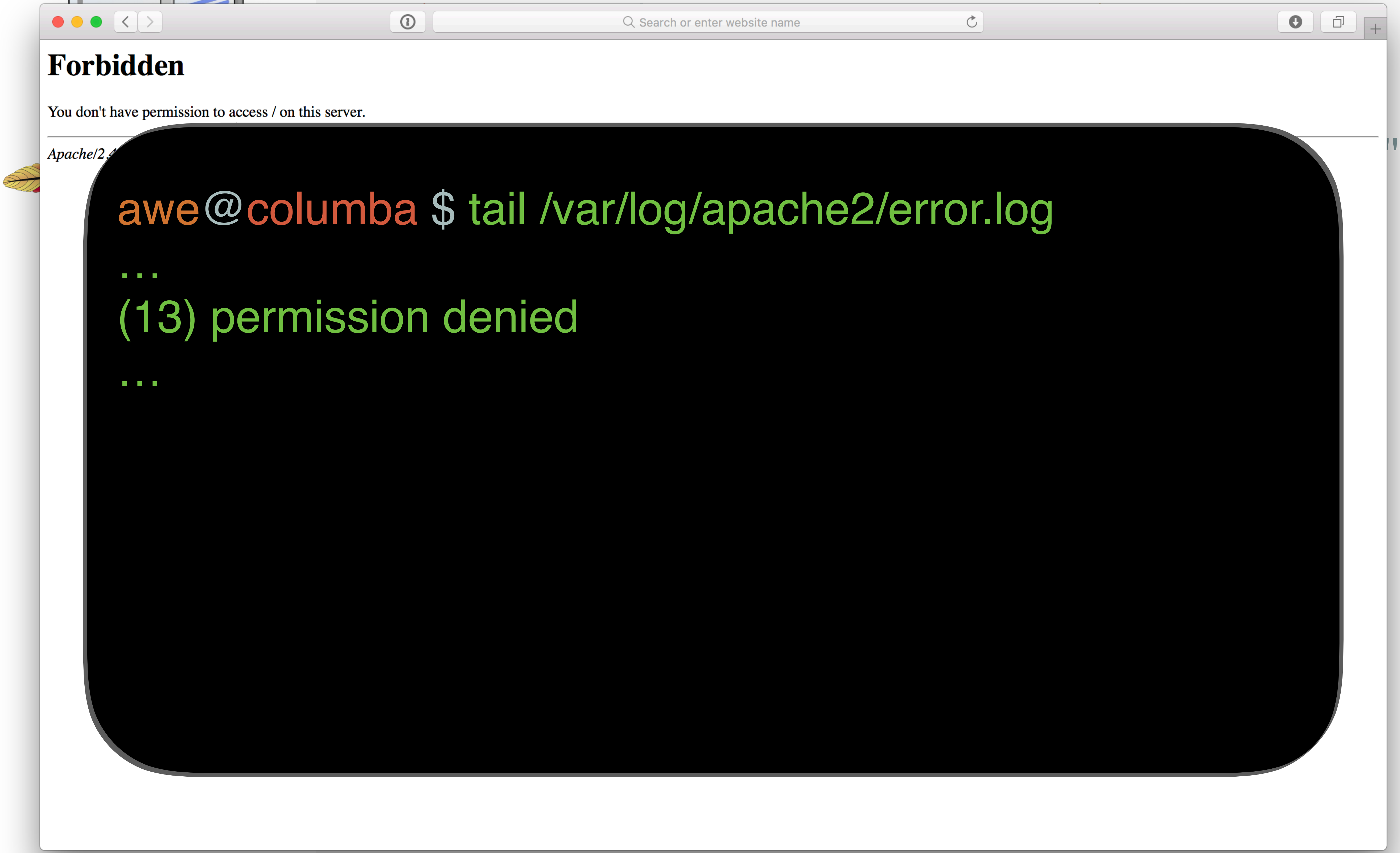
Apache/2.4 [redacted]

```
":
```

```
website {"aaronweiss.us": root => "aaronweiss" }
website {"pdgn.co": root => "pdgn" }
```



```
package {"apache2": ensure => present }
```



Forbidden

You don't have permission to access / on this server.

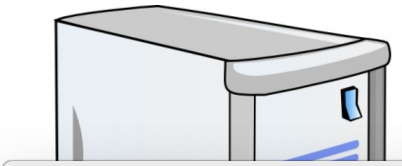
Apache/2.4

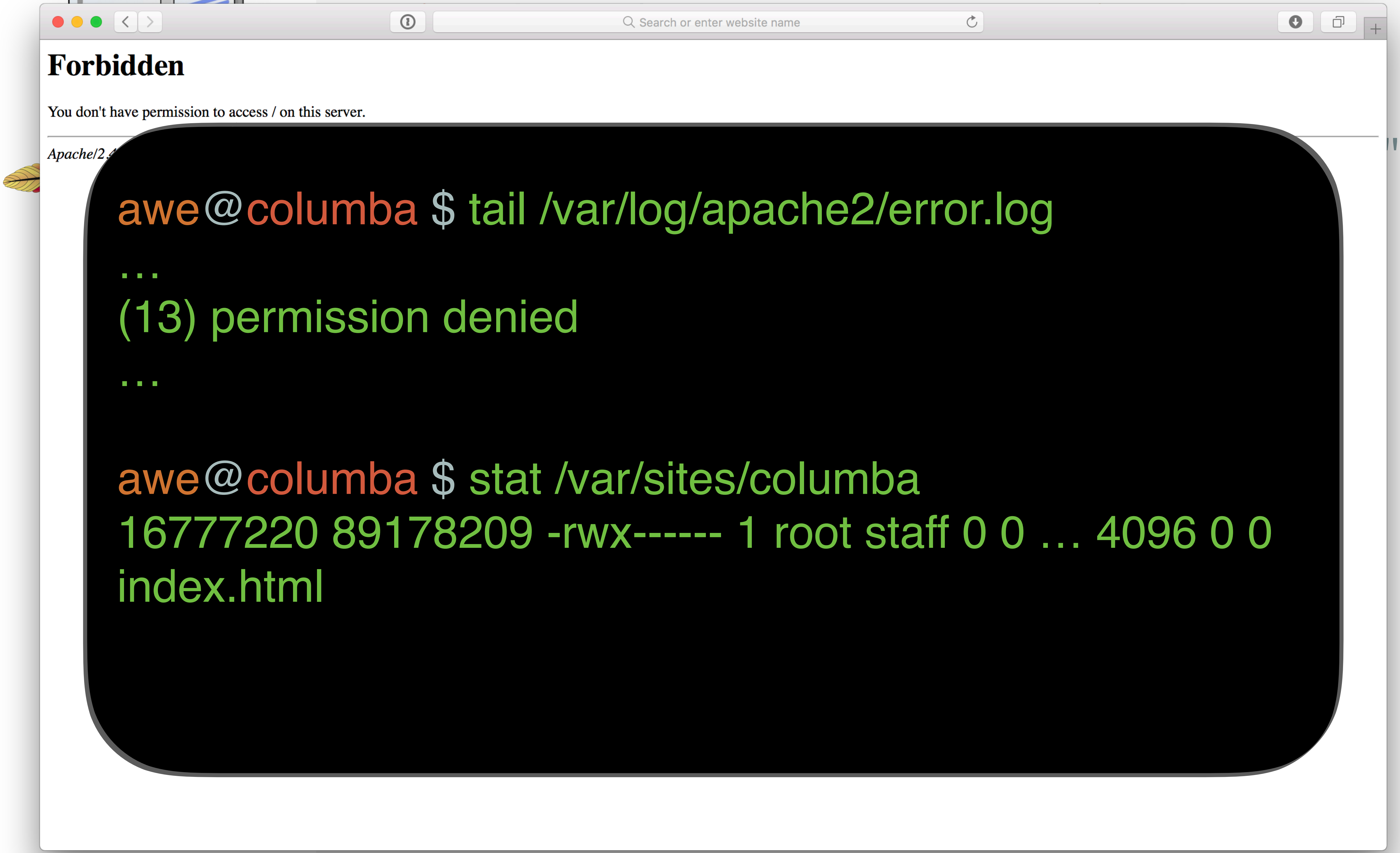
```
awe@columba $ tail /var/log/apache2/error.log
...
(13) permission denied
...
```

```
":
```

```
website {"aaronweiss.us": root => "aaronweiss" }
website {"pdgn.co": root => "pdgn" }
```

```
package {"apache2": ensure => present }
```





Forbidden

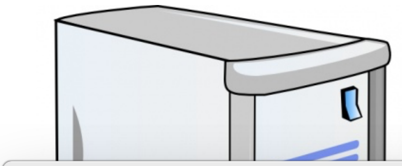
You don't have permission to access / on this server.

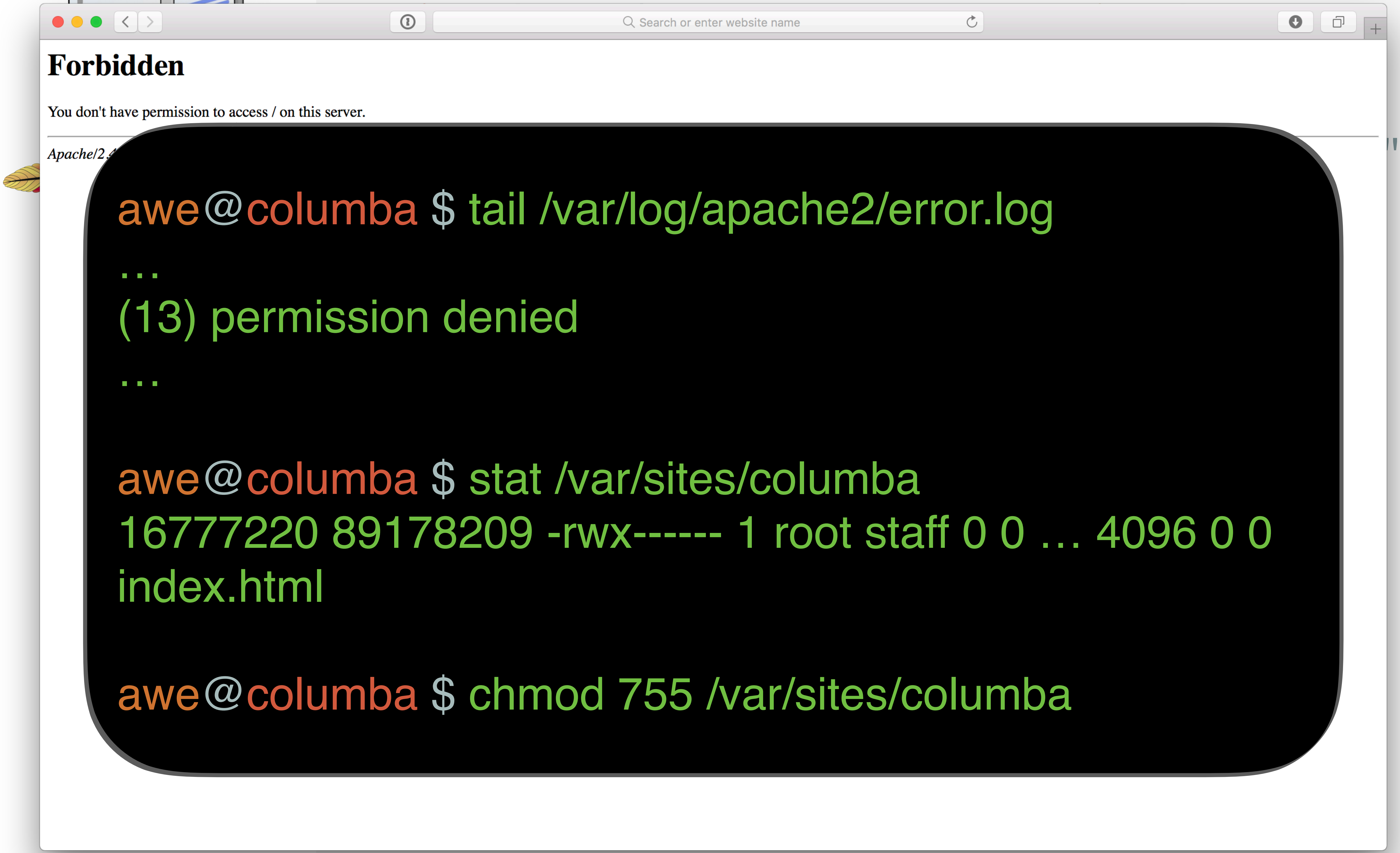
Apache/2.4

```
awe@columba $ tail /var/log/apache2/error.log
...
(13) permission denied
...

awe@columba $ stat /var/sites/columba
16777220 89178209 -rwx------ 1 root staff 0 0 … 4096 0 0
index.html
```

```
website {"aaronweiss.us": root => "aaronweiss" }
website {"pdgn.co": root => "pdgn" }
```

```
package {"apache2": ensure => present }
```





Forbidden

You don't have permission to access / on this server.

Apache/2.4

```
awe@columba $ tail /var/log/apache2/error.log
...
(13) permission denied
...

awe@columba $ stat /var/sites/columba
16777220 89178209 -rwx------ 1 root staff 0 0 … 4096 0 0
index.html

awe@columba $ chmod 755 /var/sites/columba
```

```
website {"aaronweiss.us": root => "aaronweiss" }
website {"pdgn.co": root => "pdgn" }
```

New York Stock Exchange:

"a software update went out [...] it returned an error. [...] There was clearly a difference in the configuration going into production [from the test environment]"

New York Stock Exchange:

"a software update went out [...] it returned an error. [...] There was clearly a difference in the configuration going into production [from the test environment]"

GitLab backup failure: “When we went to look for […] backups we found out they were not there. […] The backup procedure was using pg_dump 9.2, while our database is running PostgreSQL 9.6. […] The difference is the result of how our Omnibus package works.”

GitLab

New York Stock Exchange:

"a software update went out [...] it returned an error. [...] There was clearly a difference in the configuration going into production [from the test environment]"

GitLab backup failure: “When we went to look for […] backups we found out they were not there. […] The backup procedure was using pg_dump 9.2, while our database is running PostgreSQL 9.6. […] The difference is the result of how our Omnibus package works.”

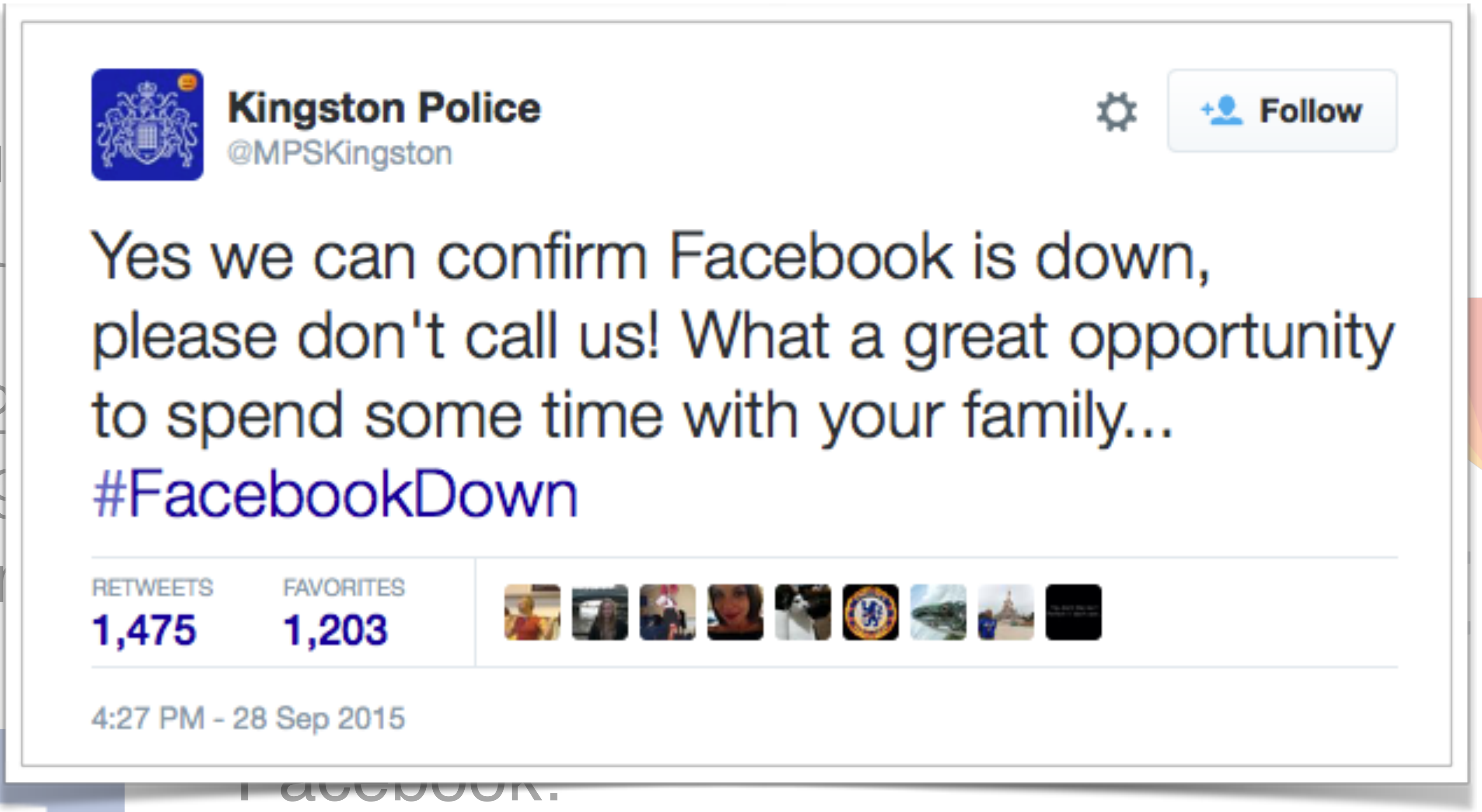
Facebook:

"Facebook was down or unreachable for many of you for approximately 2.5 hours. [...] An automated system for verifying configuration values ended up causing much more damage than it fixed."

New York Stock Exchange:

"a software update went out [...] it returned an error. [...] There was clearly a difference in the configuration going into production [from the test environment]"

GitLab backu
for [...] backu
there. [...] Th
pg_dump 9.2
PostgreSQL 9
of how our O



Kingston Police
@MPSKingston

Yes we can confirm Facebook is down, please don't call us! What a great opportunity to spend some time with your family... #FacebookDown

RETWEETS 1,475 FAVORITES 1,203

4:27 PM - 28 Sep 2015

Facebook:

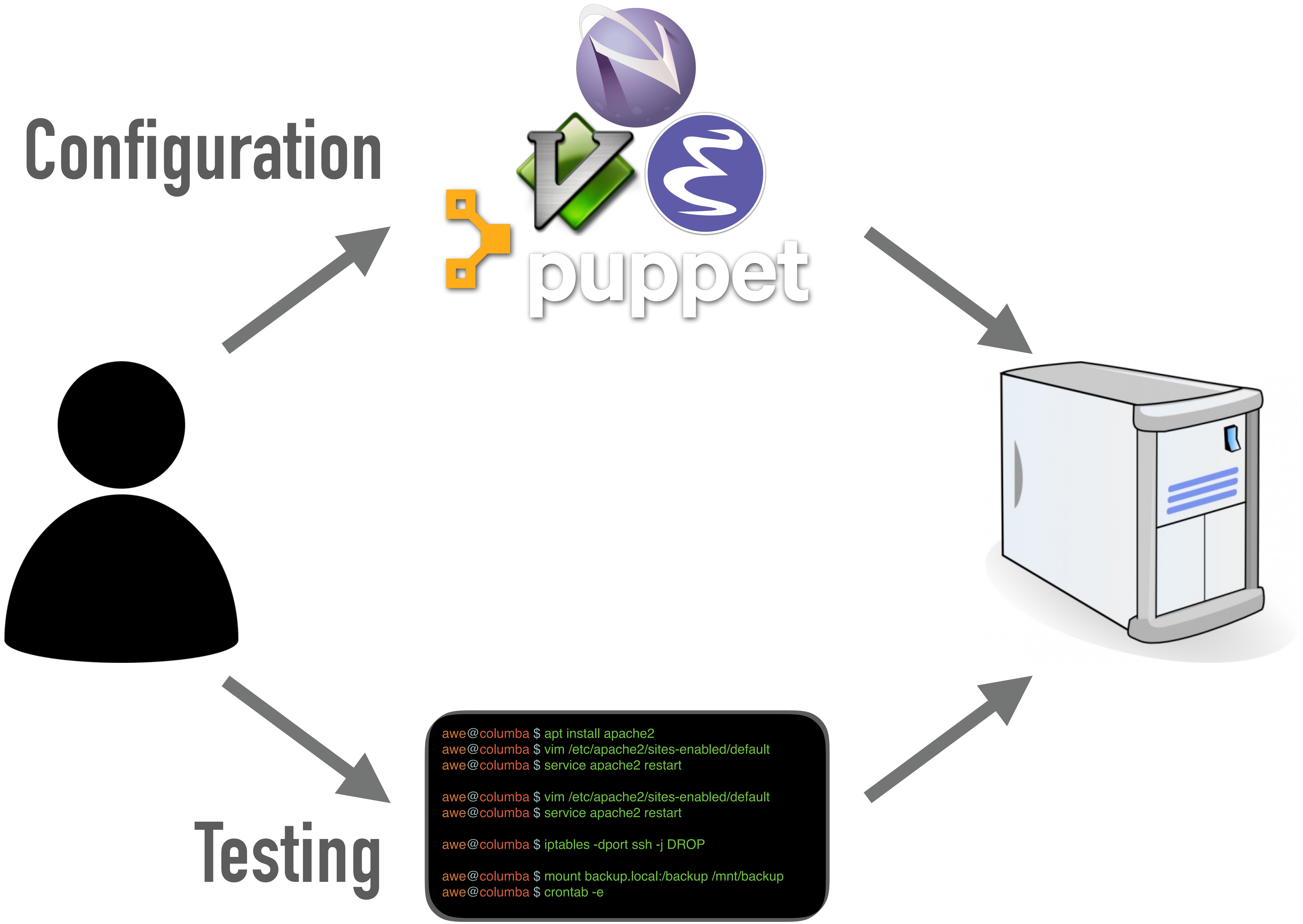
"Facebook was down or unreachable for many of you for approximately 2.5 hours. [...] An automated system for verifying configuration values ended up causing much more damage than it fixed."

WHY DO CONFIGURATION ERRORS HAPPEN?

CONFIGURATION TOOLS ARE STILL NEW

UNLIKE THE SHELL…

CONFIGURATION AND TESTING ARE TOO SEPARATED



Configuration

Testing

```
awe@columba $ apt install apache2
awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ iptables -dport ssh -j DROP

awe@columba $ mount backup.local:/backup /mnt/backup
awe@columba $ crontab -e
```

CONVENTIONAL PROGRAM REPAIR?

CONVENTIONAL PROGRAM REPAIR?

- Test-based repair
  - GenProg (Le Goues et al. TSE 2012)
  - SemFix (Nguyen et al. ICSE 2013)
  - Angelix (Mechtaev et al. ICSE 2016)
  - Staged Program Repair (Long and Rinard. FSE 2015)

GenProg

Evolutionary Program Repair

Angelix

CONVENTIONAL PROGRAM REPAIR?

GenProg

Evolutionary Program Repair

Angelix

- Test-based repair
  - GenProg (Le Goues et al. TSE 2012)
  - SemFix (Nguyen et al. ICSE 2013)
  - Angelix (Mechtaev et al. ICSE 2016)
  - Staged Program Repair (Long and Rinard. FSE 2015)
- Specification-based repair
  - EML and MPy (Singh et al. PLDI 2013)
  - Specification Repair with Alloy (Gopinath et al. TACAS 2011)

alloy

WHY NOT USE THE SHELL DIRECTLY?

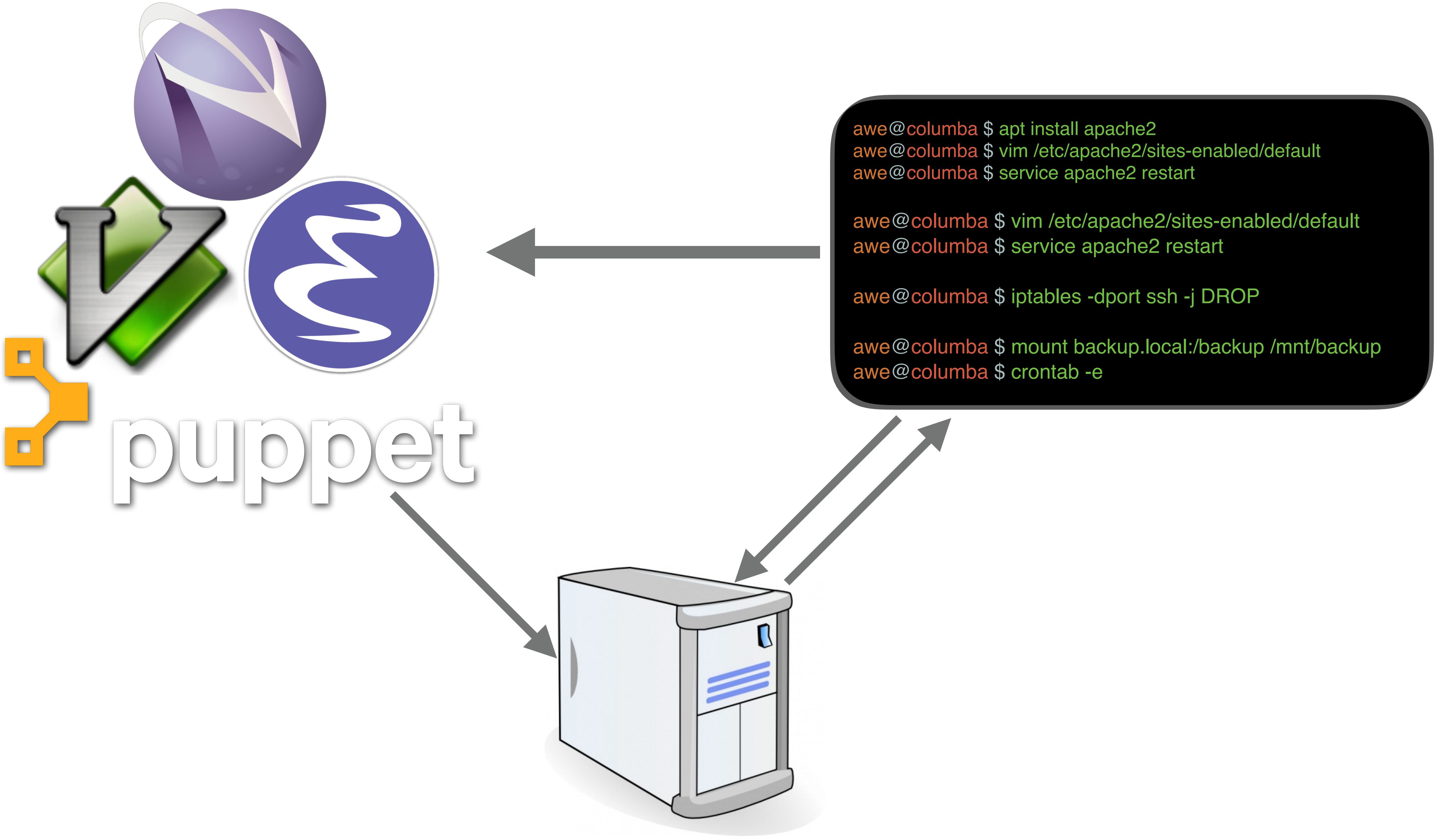
WHY NOT USE THE SHELL DIRECTLY?

Configuration drift!

SHELL-BASED CONFIGURATION REPAIR?

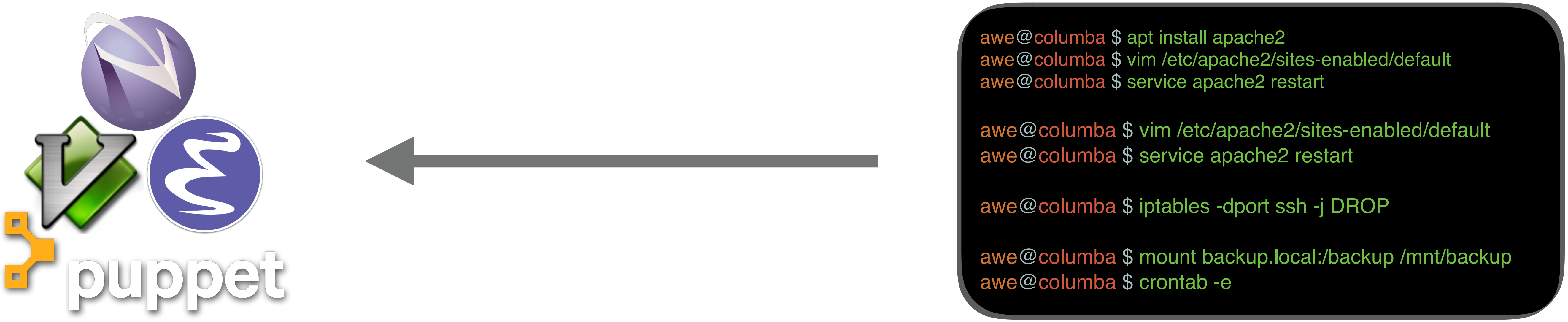
IMPERATIVE CONFIGURATION REPAIR

Live Programming for Configuration Management



```
awe@columba $ apt install apache2
awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ iptables -dport ssh -j DROP

awe@columba $ mount backup.local:/backup /mnt/backup
awe@columba $ crontab -e
```

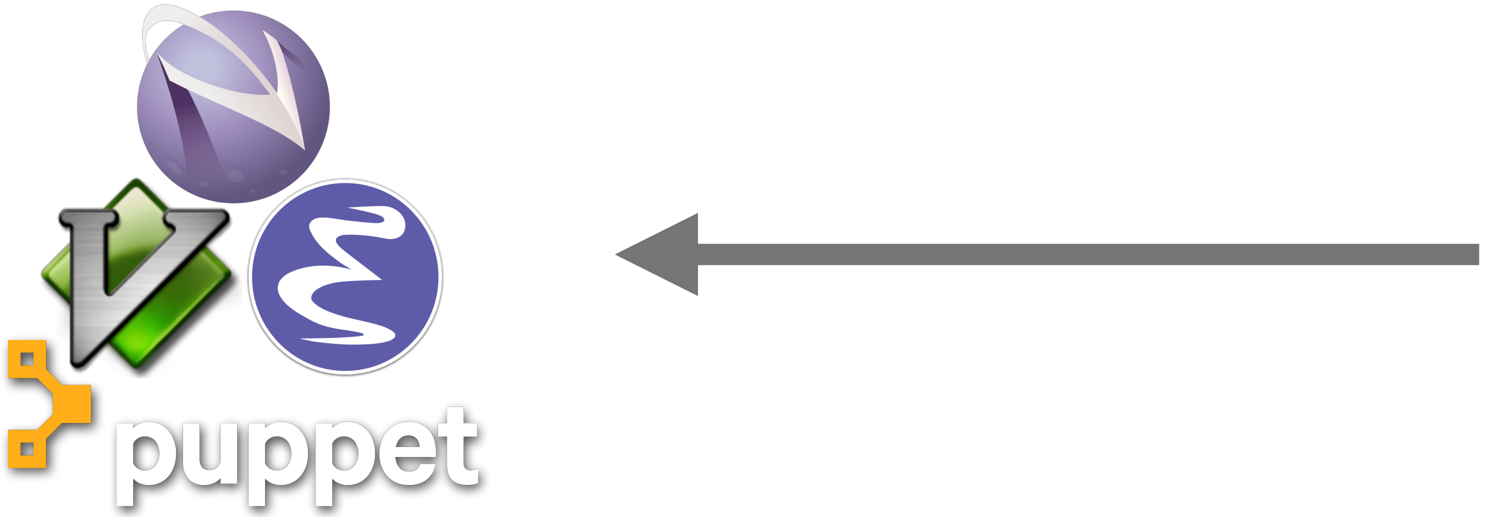
IMPERATIVE CONFIGURATION REPAIR

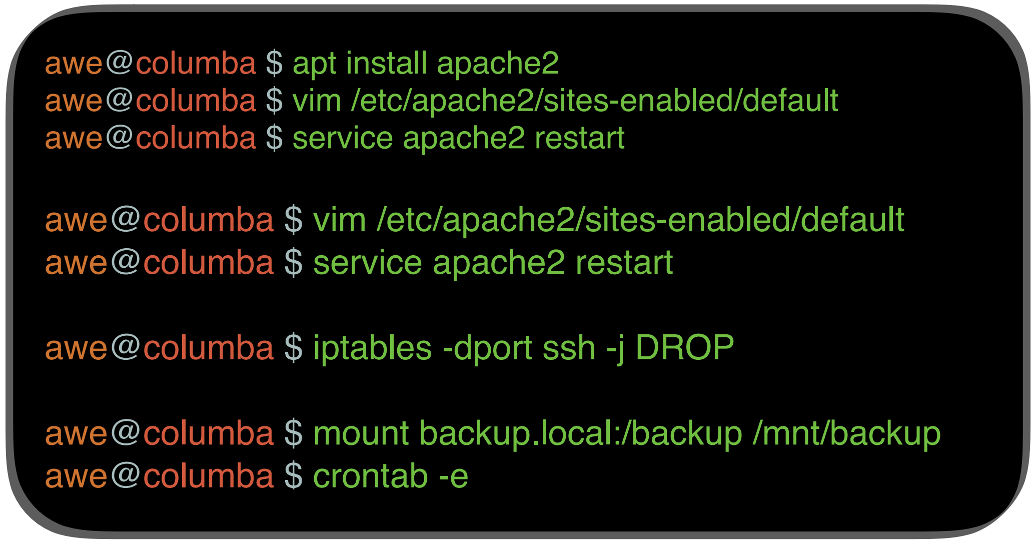


```
awe@columba $ apt install apache2
awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ iptables -dport ssh -j DROP

awe@columba $ mount backup.local:/backup /mnt/backup
awe@columba $ crontab -e
```

IMPERATIVE CONFIGURATION REPAIR

- Propagates changes made via the shell back to the original manifest!

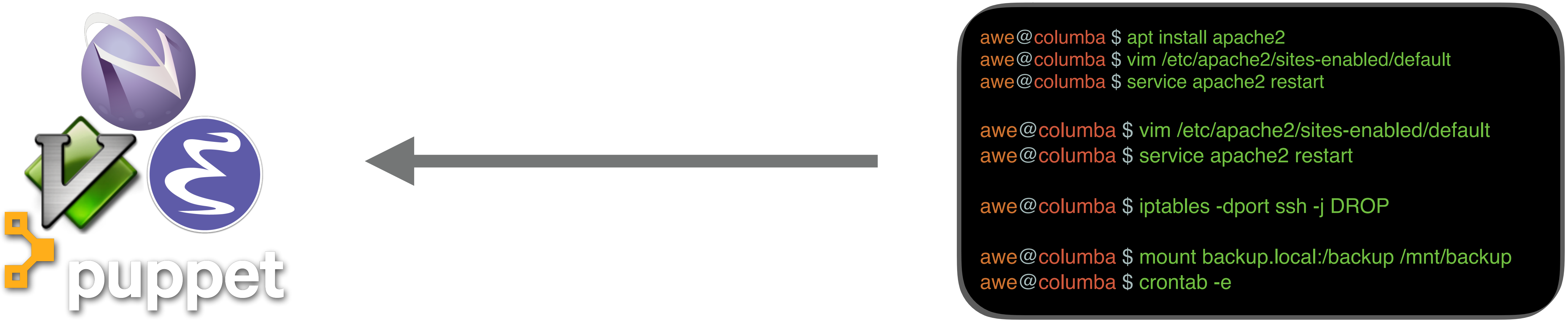




```
awe@columba $ apt install apache2
awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ iptables -dport ssh -j DROP

awe@columba $ mount backup.local:/backup /mnt/backup
awe@columba $ crontab -e
```

IMPERATIVE CONFIGURATION REPAIR

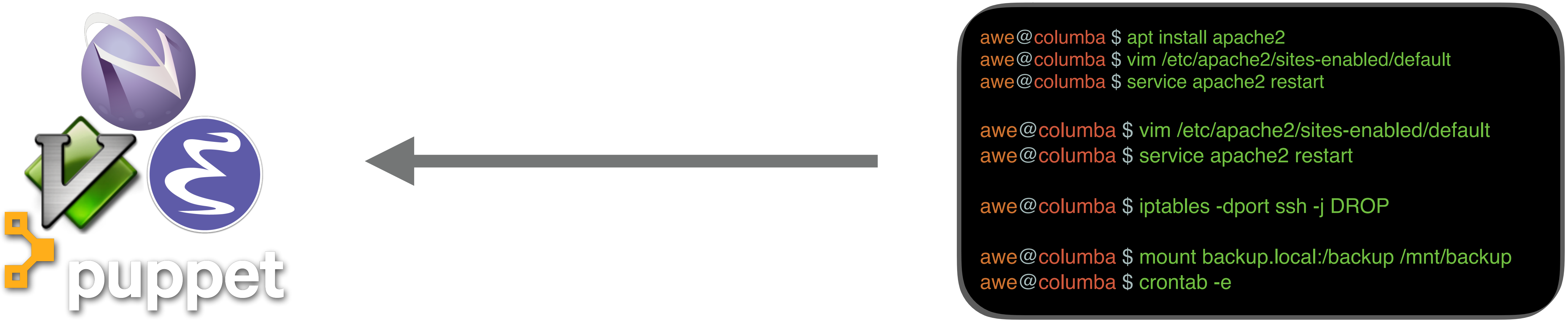
- Propagates changes made via the shell back to the original manifest!
  - Consistent: All changes made via the shell are preserved



```
awe@columba $ apt install apache2
awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ iptables -dport ssh -j DROP

awe@columba $ mount backup.local:/backup /mnt/backup
awe@columba $ crontab -e
```

IMPERATIVE CONFIGURATION REPAIR

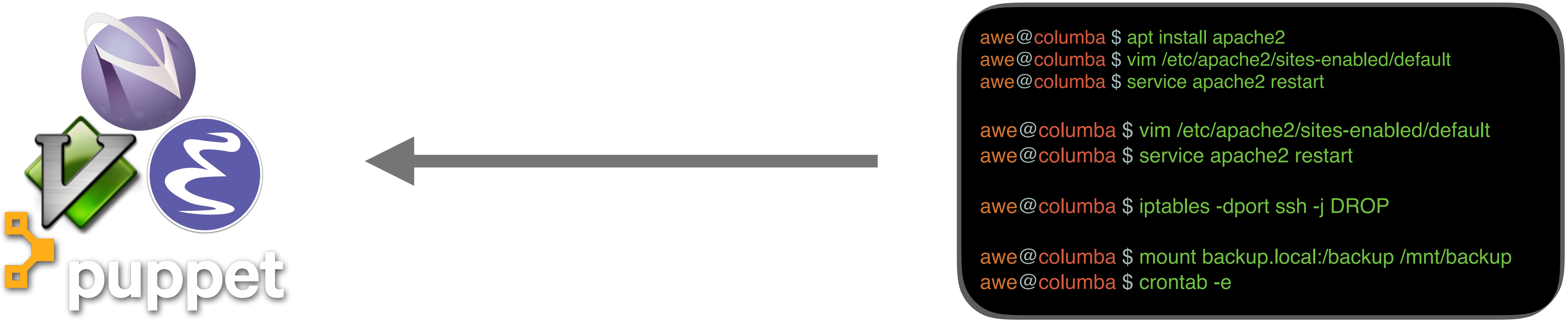
- Propagates changes made via the shell back to the original manifest!
  - Consistent: All changes made via the shell are preserved
  - Maintainable: Structure and abstraction is preserved



```
awe@columba $ apt install apache2
awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ iptables -dport ssh -j DROP

awe@columba $ mount backup.local:/backup /mnt/backup
awe@columba $ crontab -e
```

IMPERATIVE CONFIGURATION REPAIR

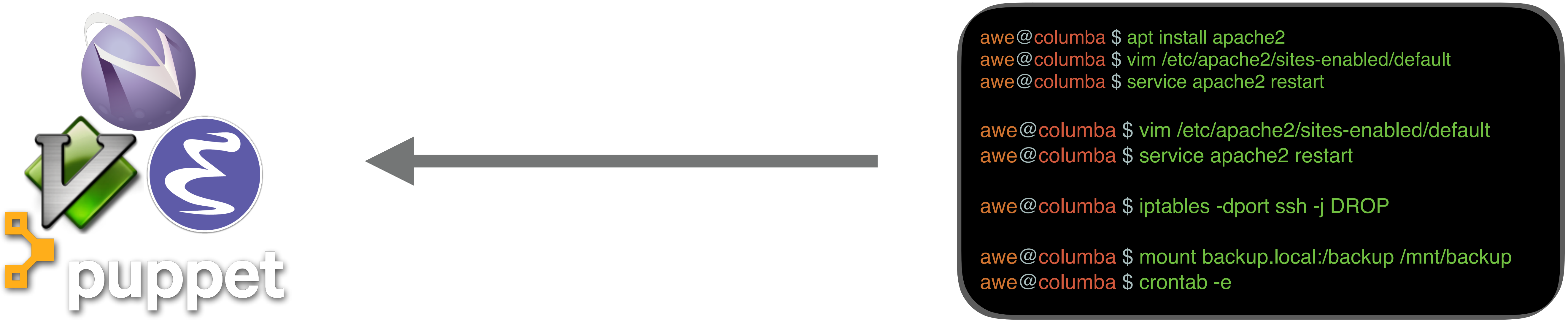
- Propagates changes made via the shell back to the original manifest!
  - Consistent: All changes made via the shell are preserved
  - Maintainable: Structure and abstraction is preserved
  - Ranked: Multiple possible repairs are ranked



```
awe@columba $ apt install apache2
awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ iptables -dport ssh -j DROP

awe@columba $ mount backup.local:/backup /mnt/backup
awe@columba $ crontab -e
```

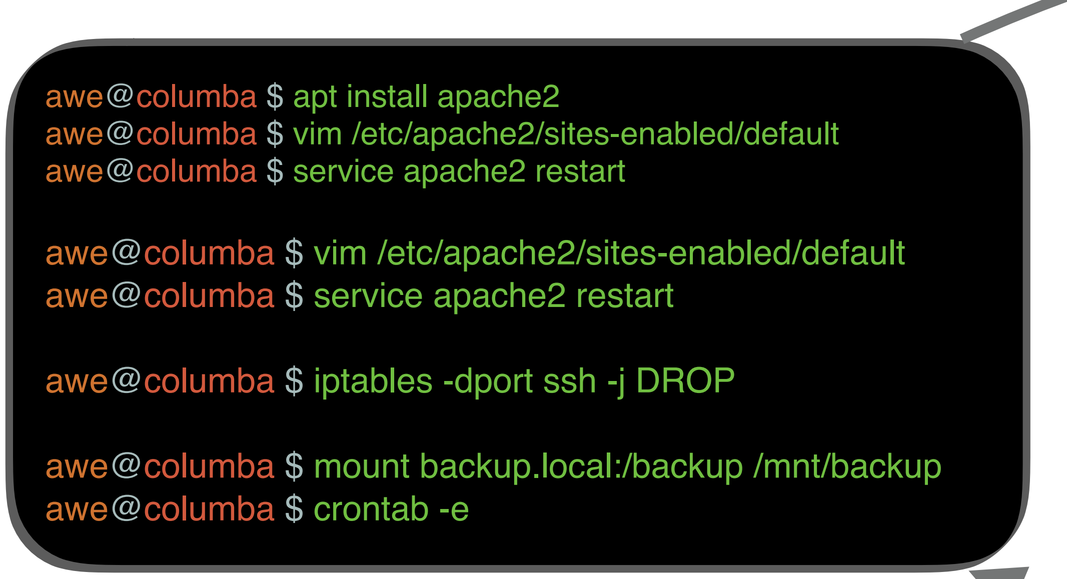
IMPERATIVE CONFIGURATION REPAIR

- Propagates changes made via the shell back to the original manifest!
  - Consistent: All changes made via the shell are preserved
  - Maintainable: Structure and abstraction is preserved
  - Ranked: Multiple possible repairs are ranked
  - Unrestricted: Works with all existing shells

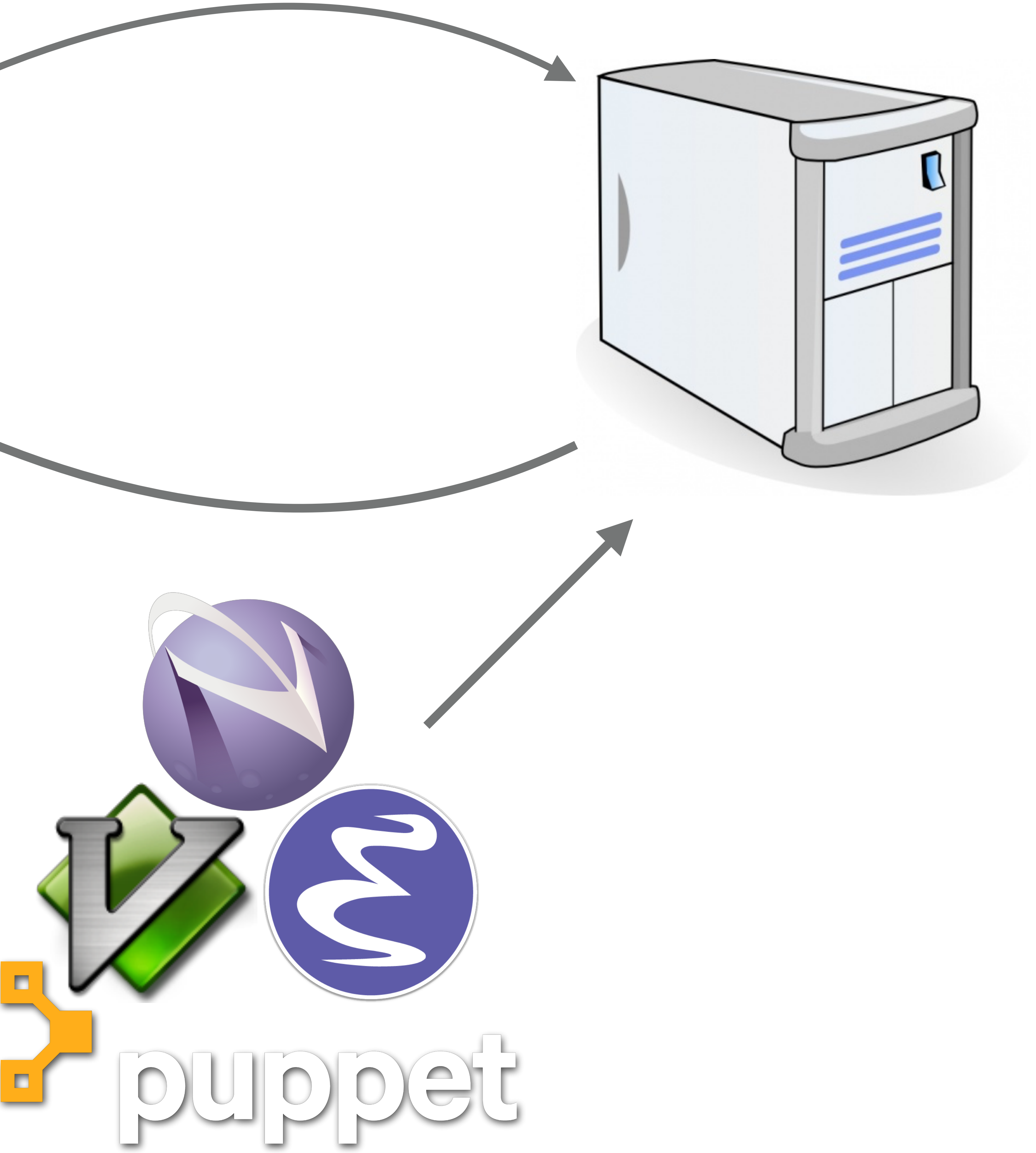


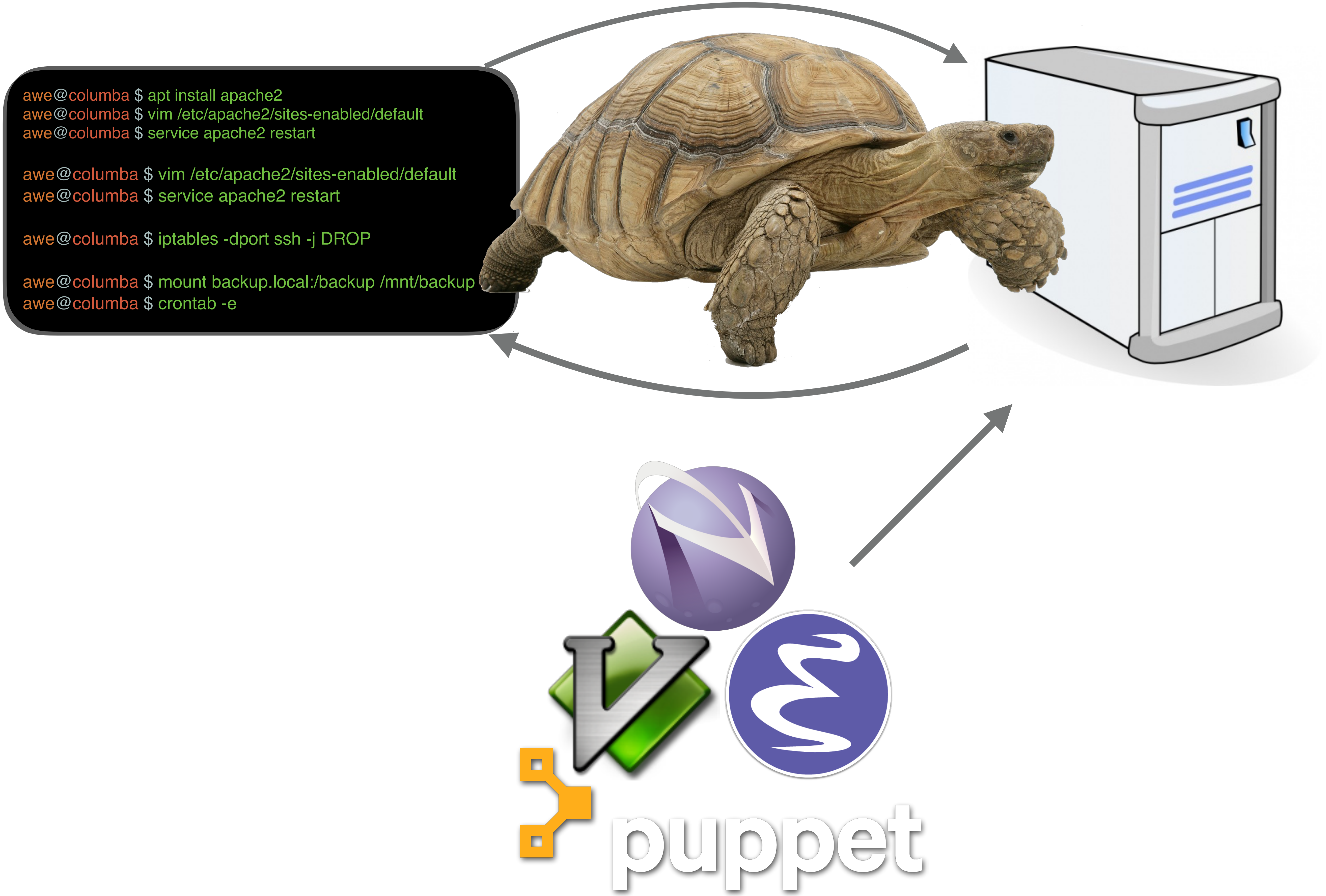
```
awe@columba $ apt install apache2
awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ iptables -dport ssh -j DROP

awe@columba $ mount backup.local:/backup /mnt/backup
awe@columba $ crontab -e
```

TORTOISE: AN ICR PROTOTYPE

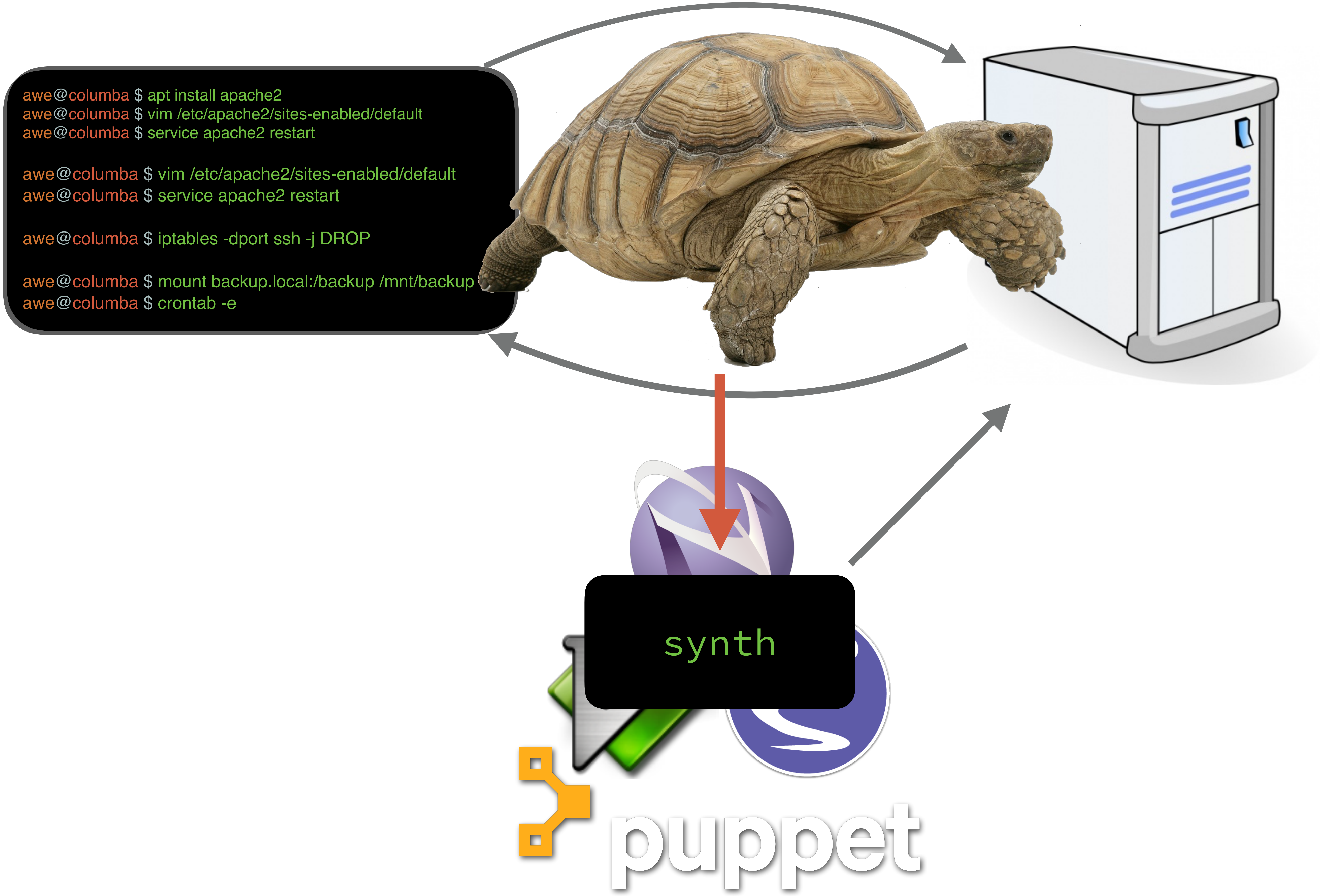


```
awe@columba $ apt install apache2
awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ iptables -dport ssh -j DROP

awe@columba $ mount backup.local:/backup /mnt/backup
awe@columba $ crontab -e
```



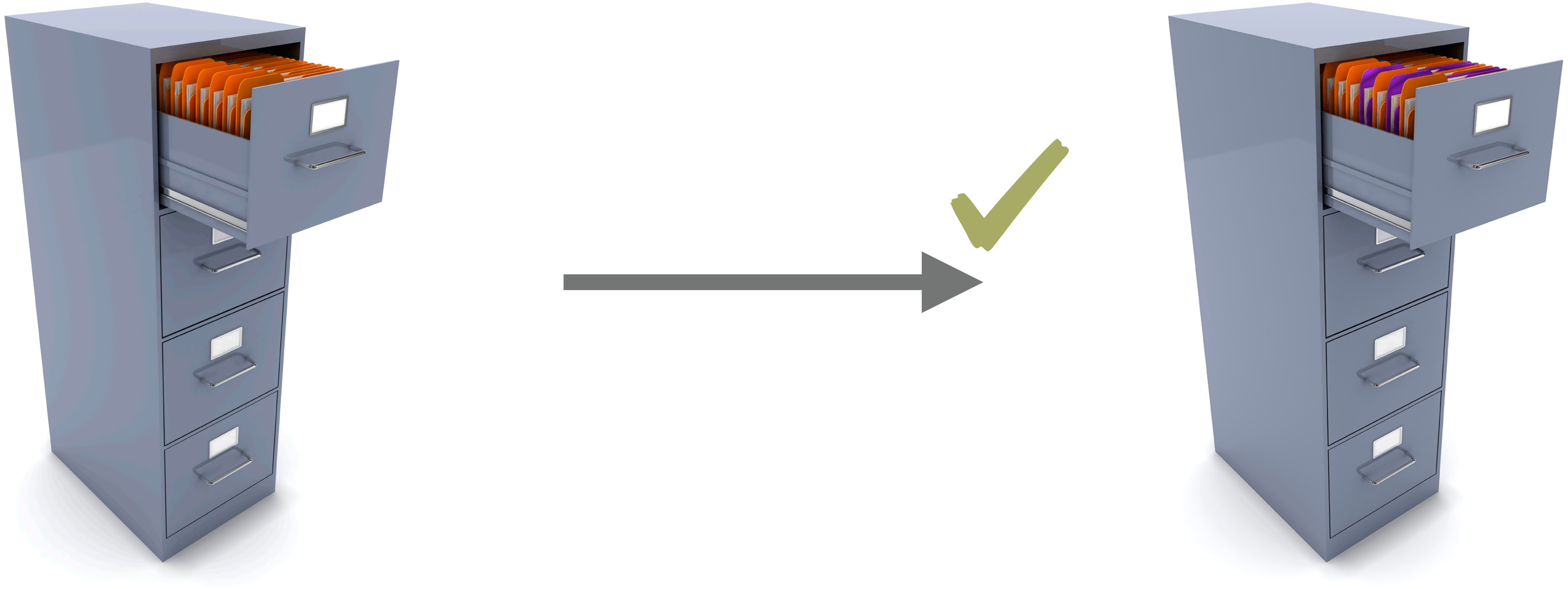


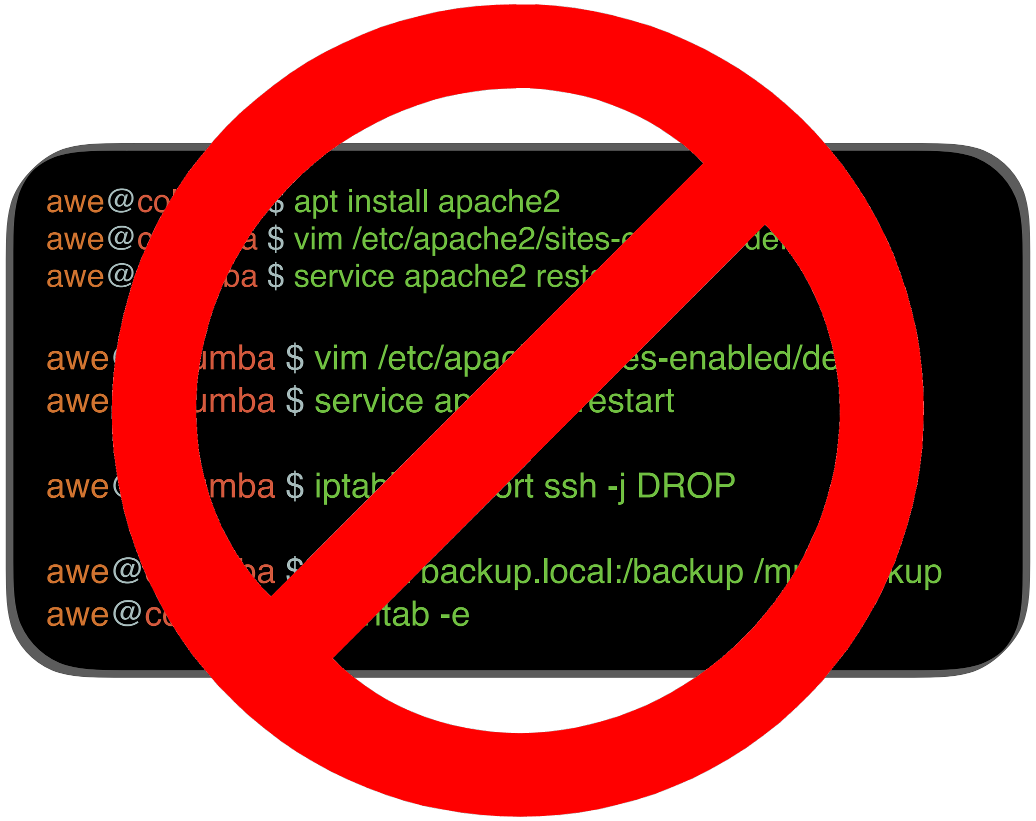
```
awe@columba $ apt install apache2
awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ iptables -dport ssh -j DROP

awe@columba $ mount backup.local:/backup /mnt/backup
awe@columba $ crontab -e
```



```
awe@columba $ apt install apache2
awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ iptables -dport ssh -j DROP

awe@columba $ mount backup.local:/backup /mnt/backup
awe@columba $ crontab -e
```

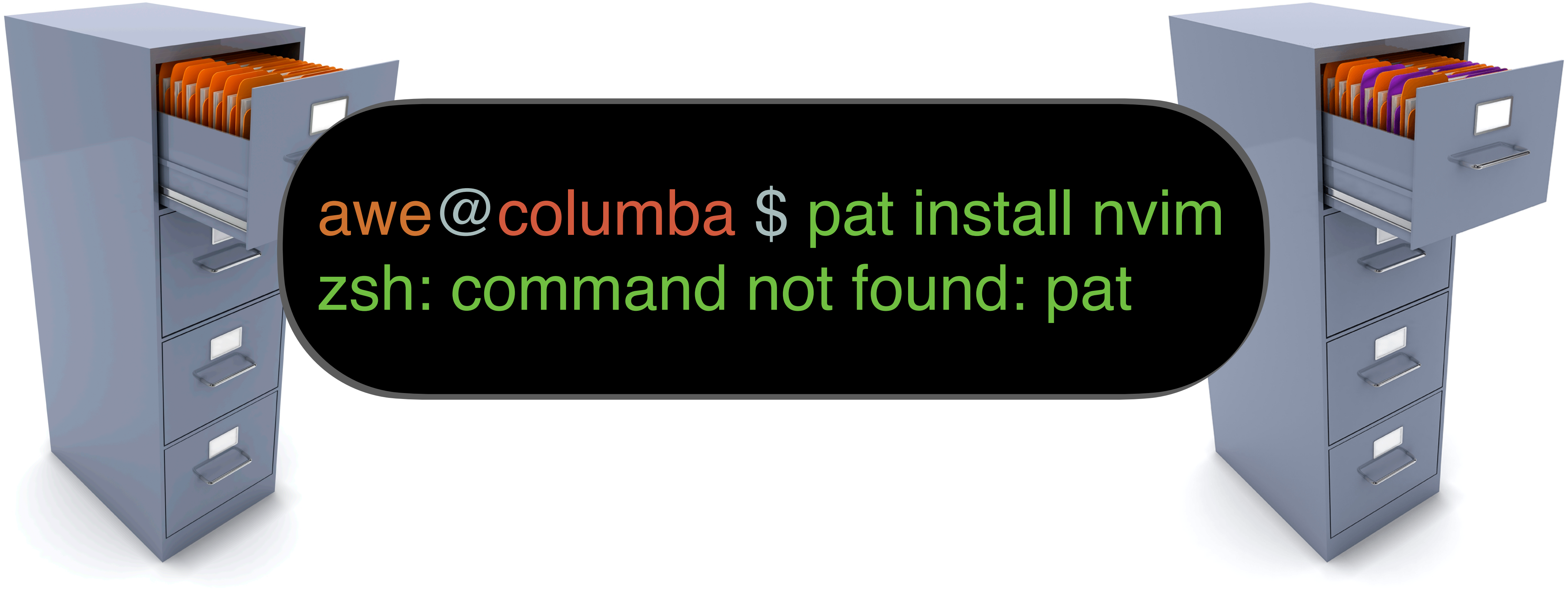
FILE SYSTEMS, NOT COMMANDS



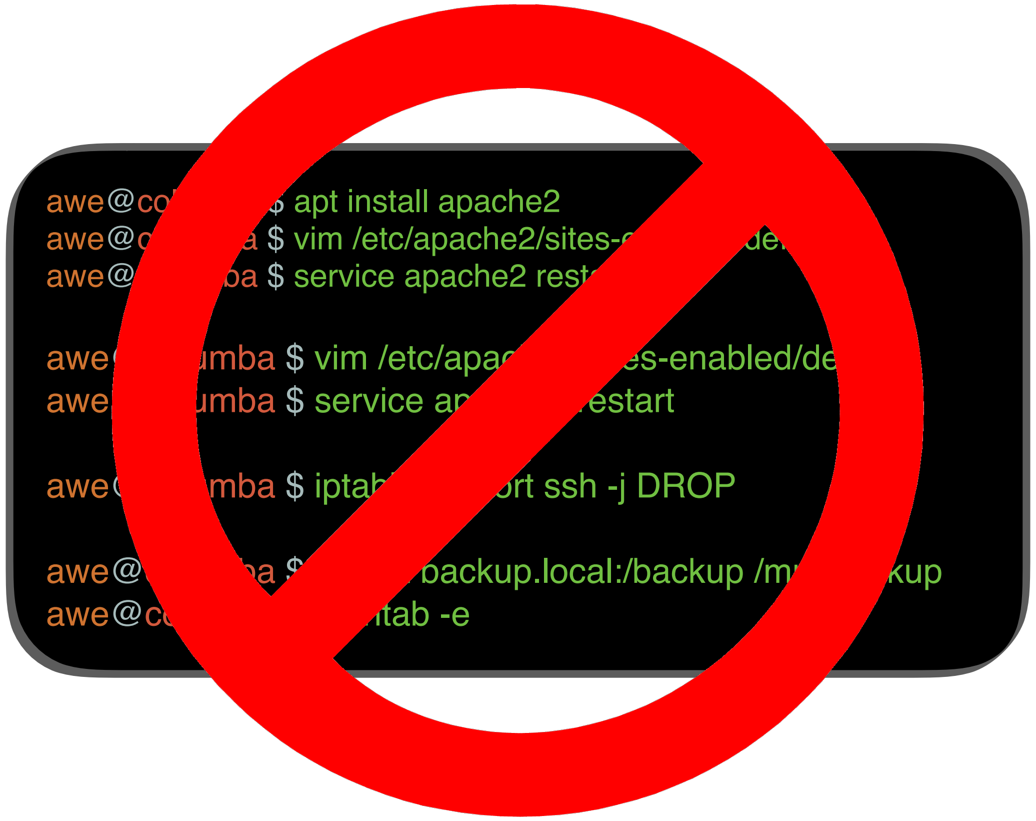


```
awe@co       $ apt install apache2
awe@c      a $ vim /etc/apache2/sites-        de
awe@      ba $ service apache2 rest

awe      umba $ vim /etc/apac        es-enabled/de
awe      umba $ service ap        estart

awe      mba $ iptab        ort ssh -j DROP

awe@      ba $        backup.local:/backup /m        kup
awe@c              tab -e
```

FILE SYSTEMS, NOT COMMANDS

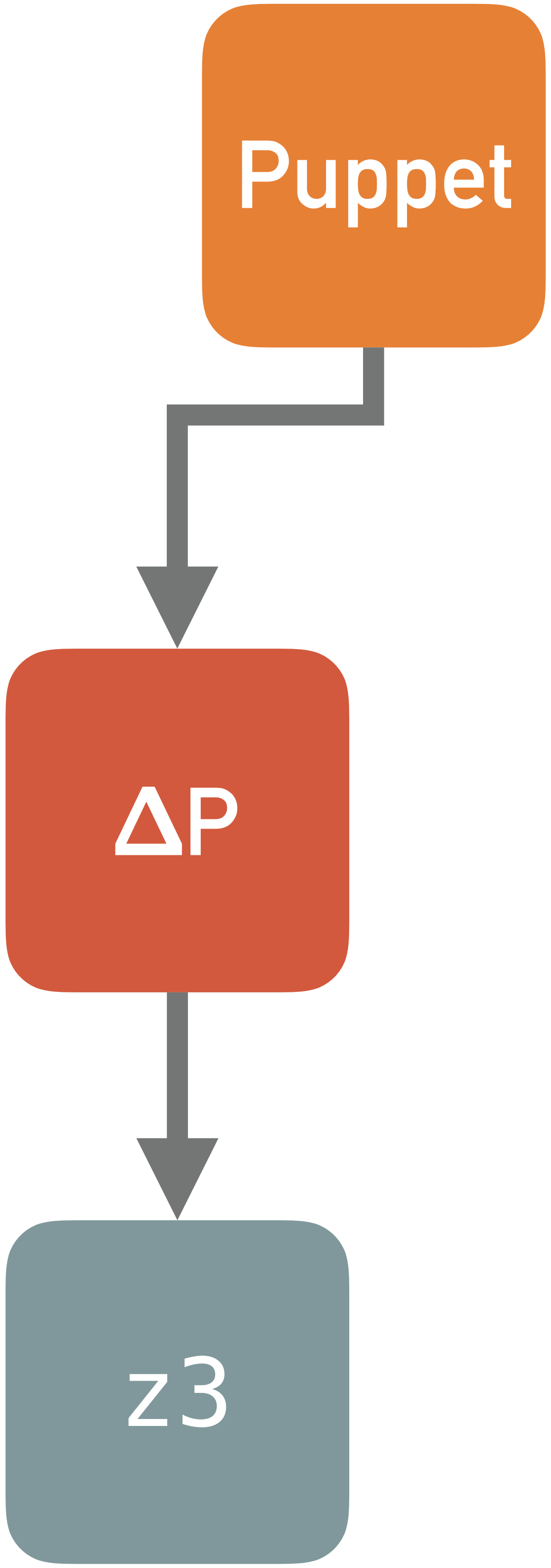


```
awe@columba $ pat install nvim
zsh: command not found: pat
```



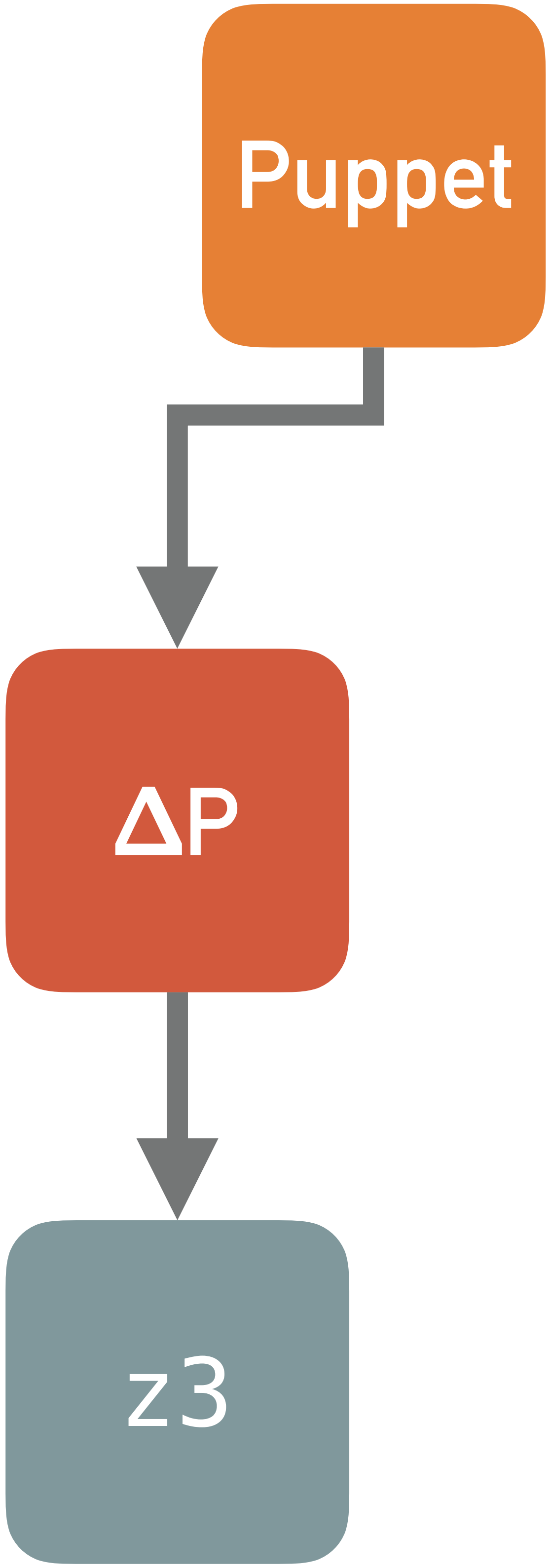
```
awe@co       $ apt install apache2
awe@c      a $ vim /etc/apache2/sites-       de
awe@      ba $ service apache2 rest

awe      umba $ vim /etc/apac      es-enabled/de
awe      umba $ service ap        estart

awe      mba $ iptab       ort ssh -j DROP

awe@      ba $      backup.local:/backup /m      kup
awe@c              tab -e
```

BEHIND THE SCENES



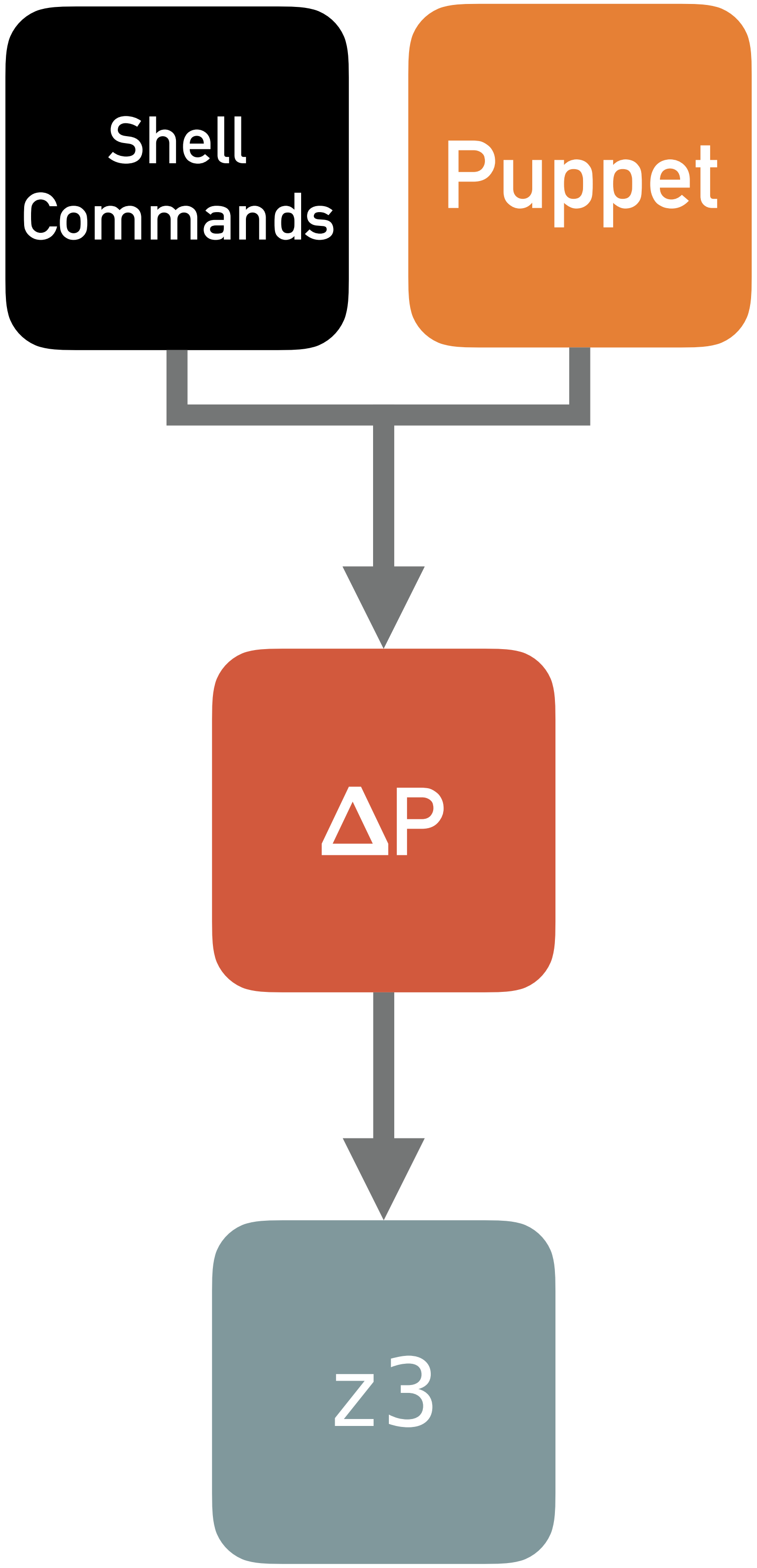
BEHIND THE SCENES

- Compile manifest into ΔP specification
  - Imperative language + filesystem ops
  - Let bindings with z3-updatable values

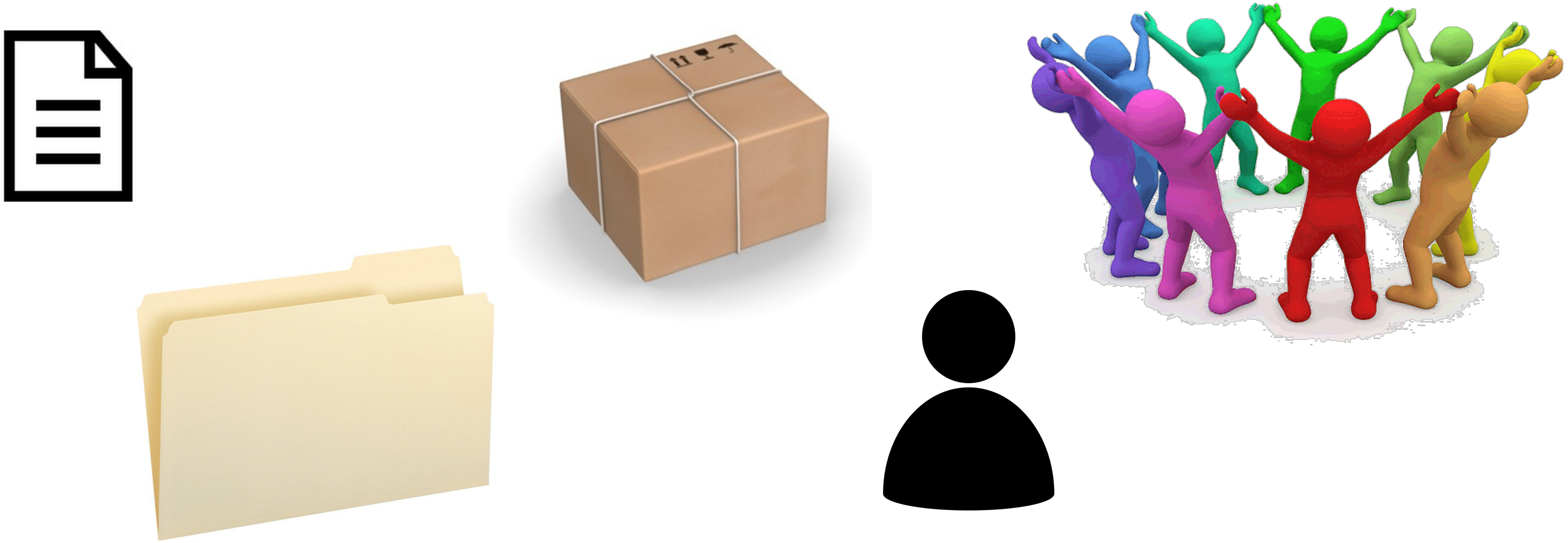


BEHIND THE SCENES

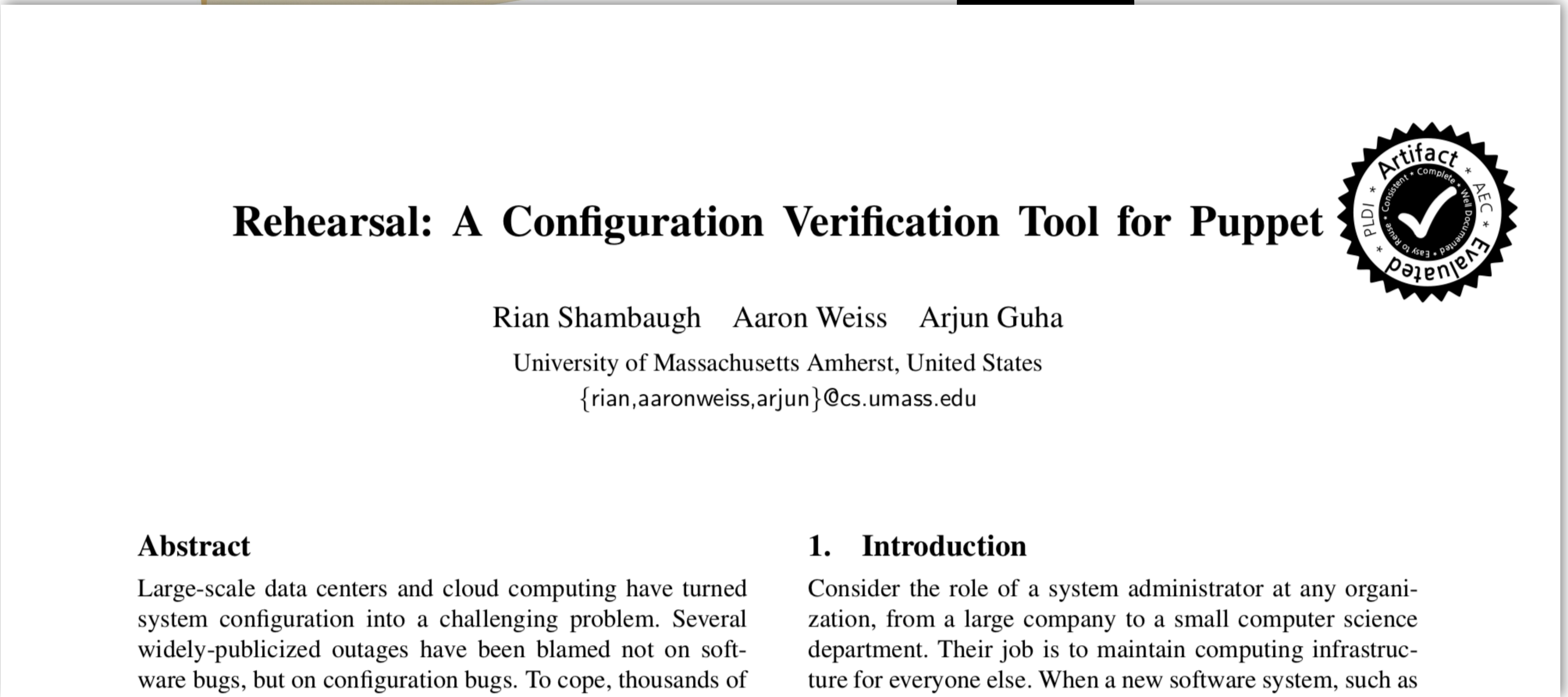
- Compile manifest into ΔP specification
  - Imperative language + filesystem ops
  - Let bindings with z3-updatable values
- Convert shell commands into ΔP assertions
  - Use ptrace to collect affected paths, and check their final state at synthesis time
  - Changes via the shell mean these assertions are false
  - z3 will repair the manifest to make them true



MODELING PUPPET



MODELING PUPPET



Rehearsal: A Configuration Verification Tool for Puppet

Artifact * AEC * Evaluated * PLDI * Consistent * Complete * Well Documented * Easy to Reuse

Rian Shambaugh Aaron Weiss Arjun Guha
University of Massachusetts Amherst, United States
{rian,aaronweiss,arjun}@cs.umass.edu

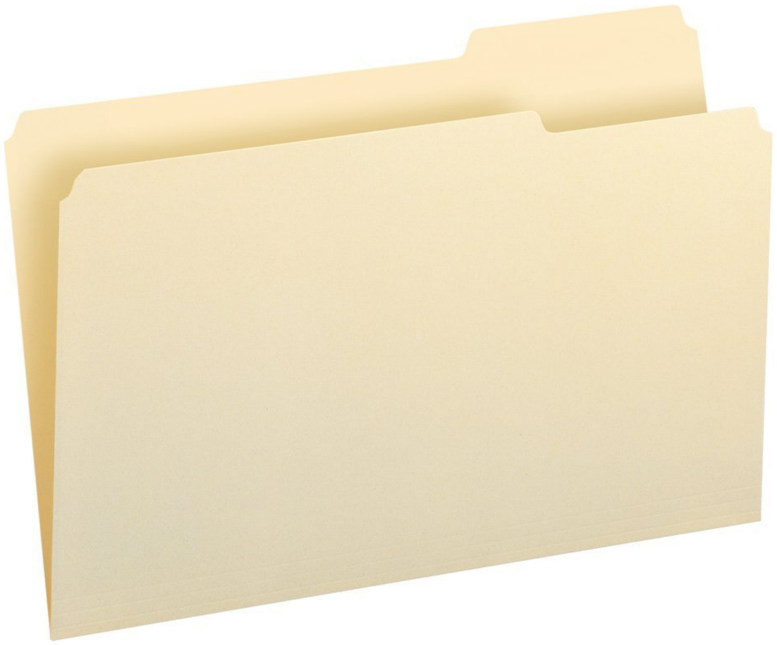
Abstract

Large-scale data centers and cloud computing have turned system configuration into a challenging problem. Several widely-publicized outages have been blamed not on software bugs, but on configuration bugs. To cope, thousands of

1. Introduction

Consider the role of a system administrator at any organization, from a large company to a small computer science department. Their job is to maintain computing infrastructure for everyone else. When a new software system, such as

ΔP: A MODEL OF FILE SYSTEM EFFECTS



COMPILING A SIMPLE EXAMPLE TO ΔP

```
define rootdir($name) {
  file {"/$name":
    ensure => directory
  }
}
rootdir {
  name => “foo”
}
```

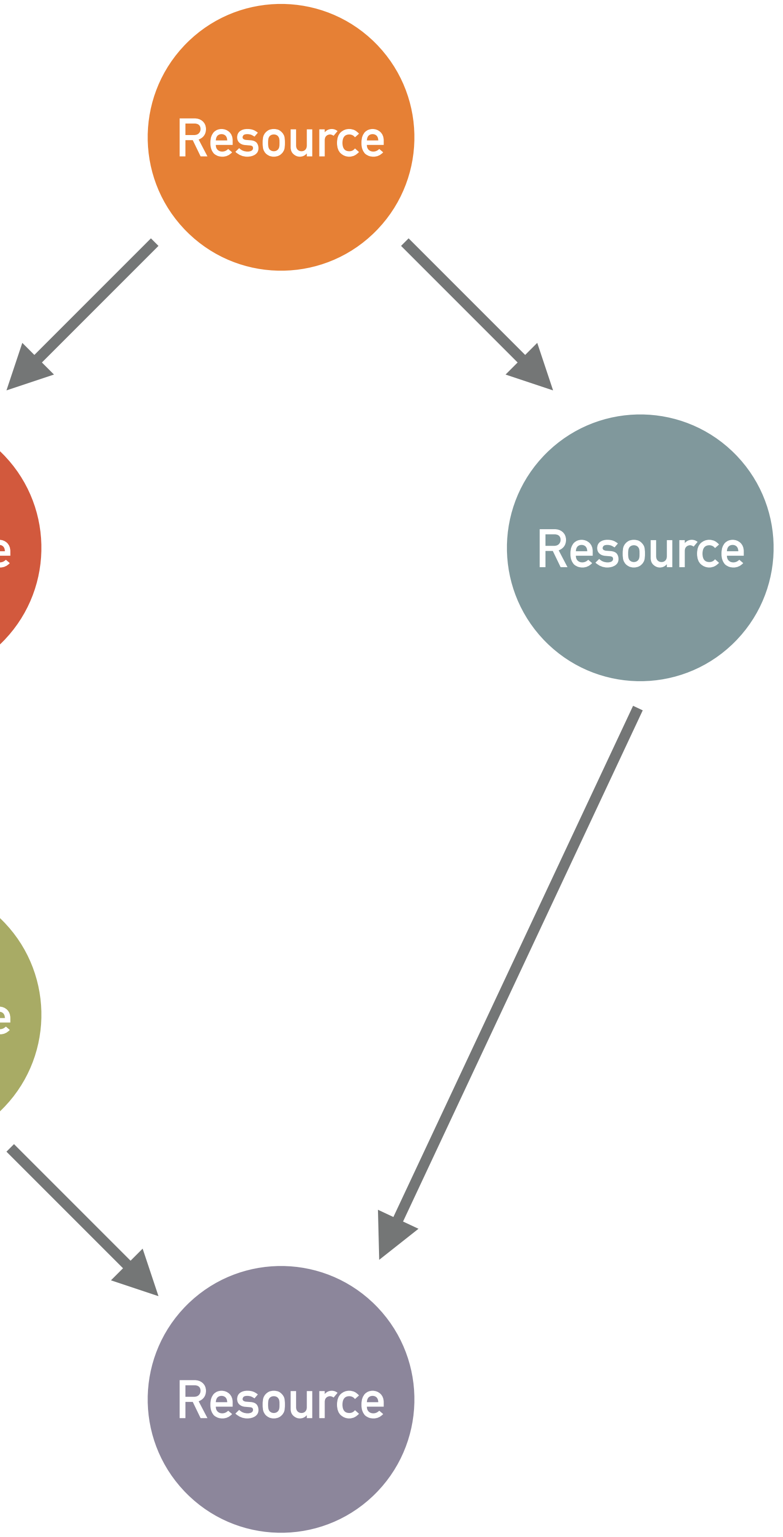
COMPILING A SIMPLE EXAMPLE TO ΔP

```
    define rootdir($name) {
rlet name = "foo" from str;
rlet ensure = "directory" from str;
if (ensure == "directory") {
  mkdir("/" + name)
} else if ... {
  ...
} ...
    }
```

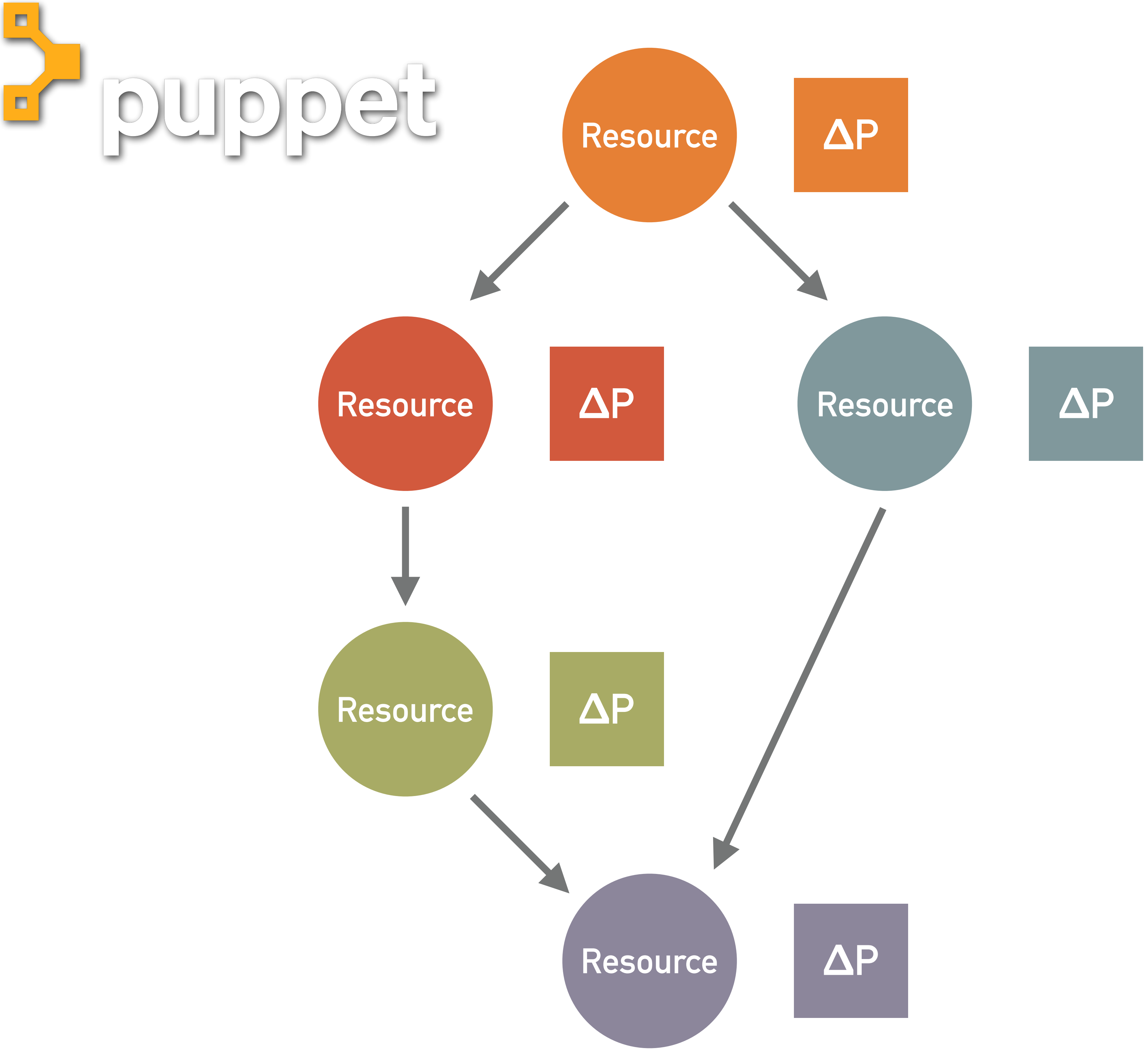
COMPILING A SIMPLE EXAMPLE TO ΔP

```
    define rootdir($name) {
rlet name ℓ1= "foo" ℓ2from str;
rlet ensure = "directory" from str;
if (ensure == "directory") {
  mkdir("/" + name)
} else if ... {
  ...
} ...
    }
```

RESOURCE GRAPHS AND ΔP

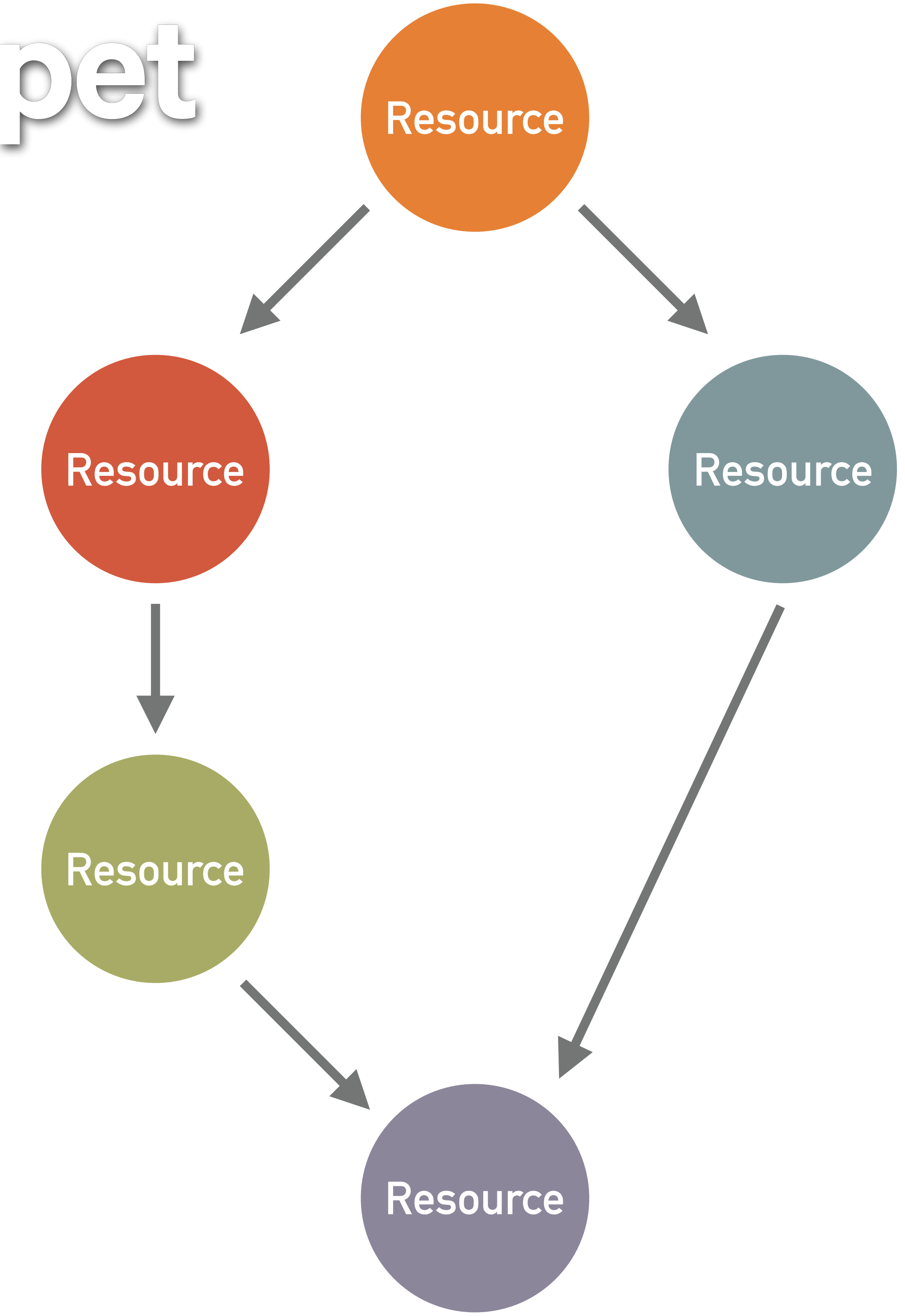


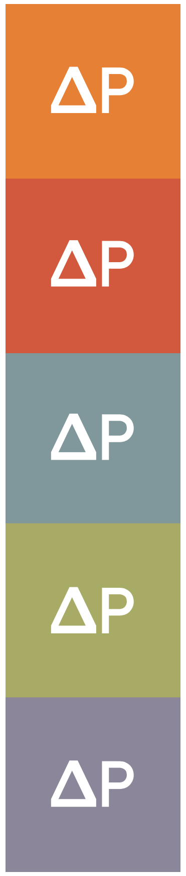
RESOURCE GRAPHS AND ΔP



RESOURCE GRAPHS AND ΔP

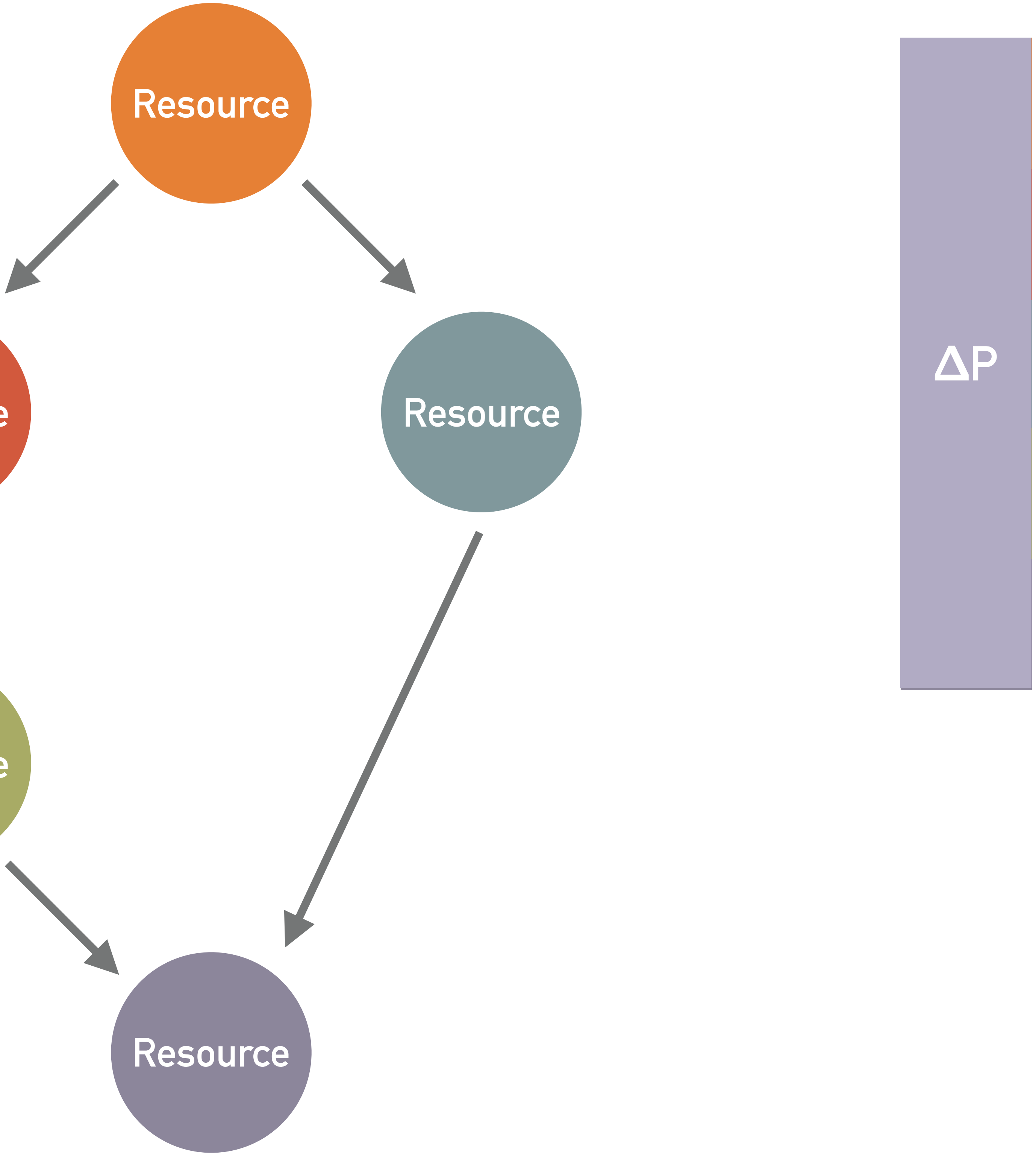
puppet



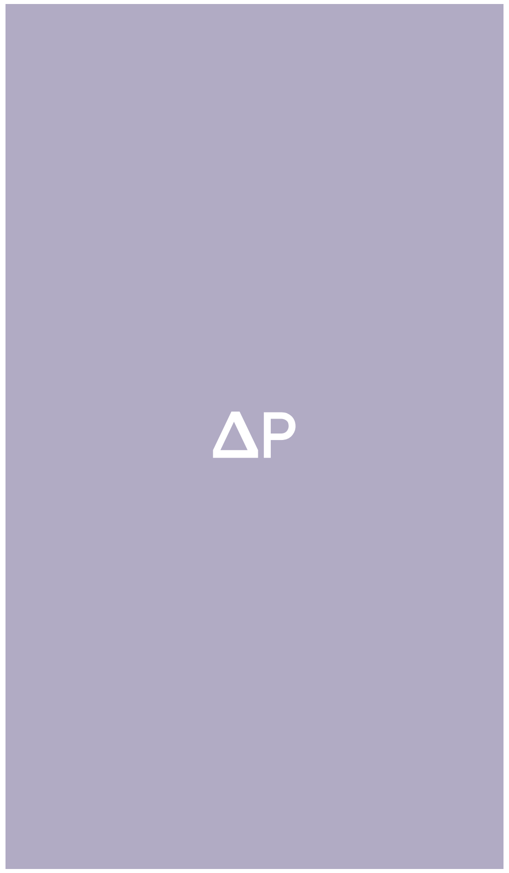


RESOURCE GRAPHS AND ΔP

puppet

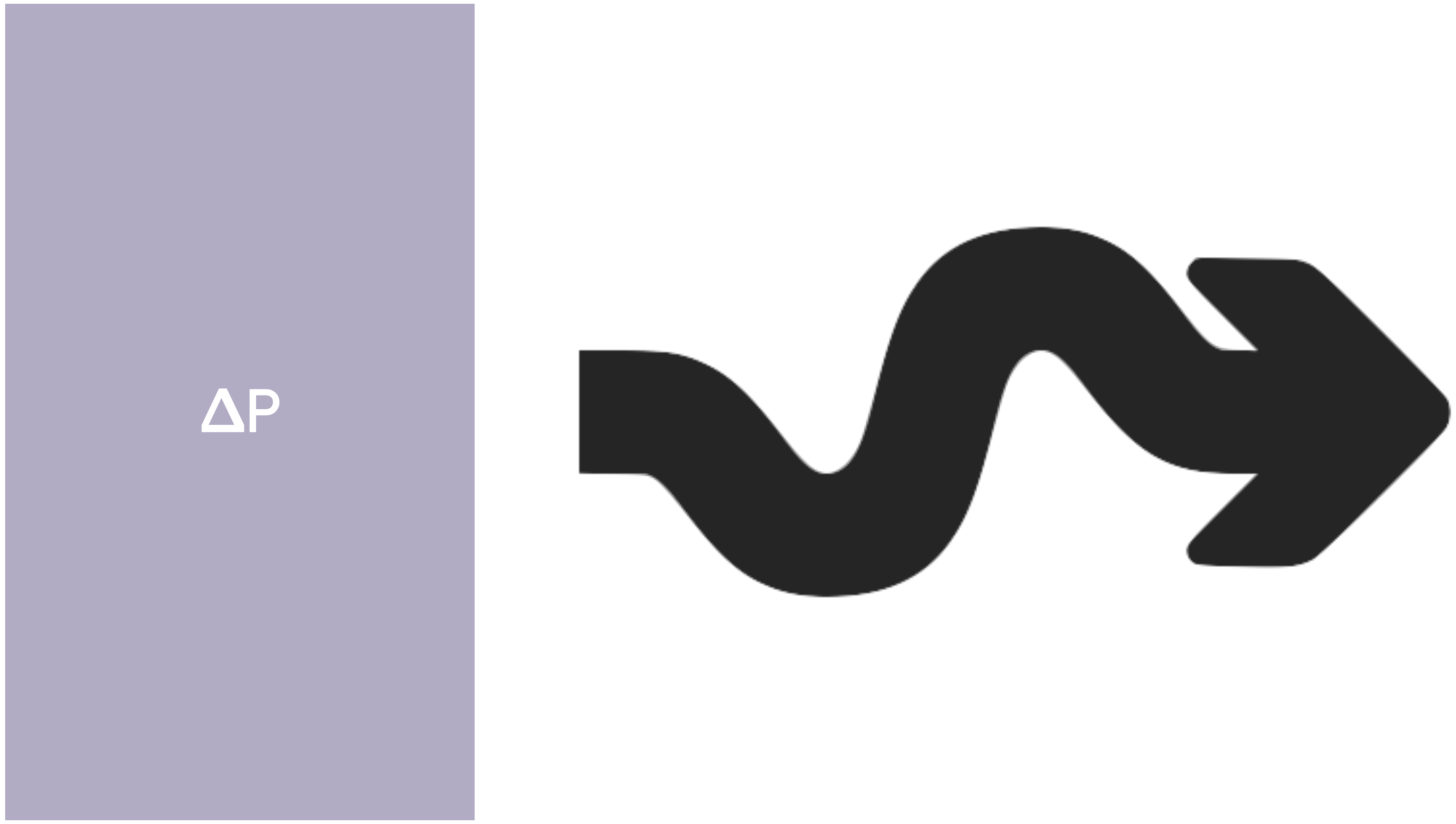


FROM ΔP TO SMT



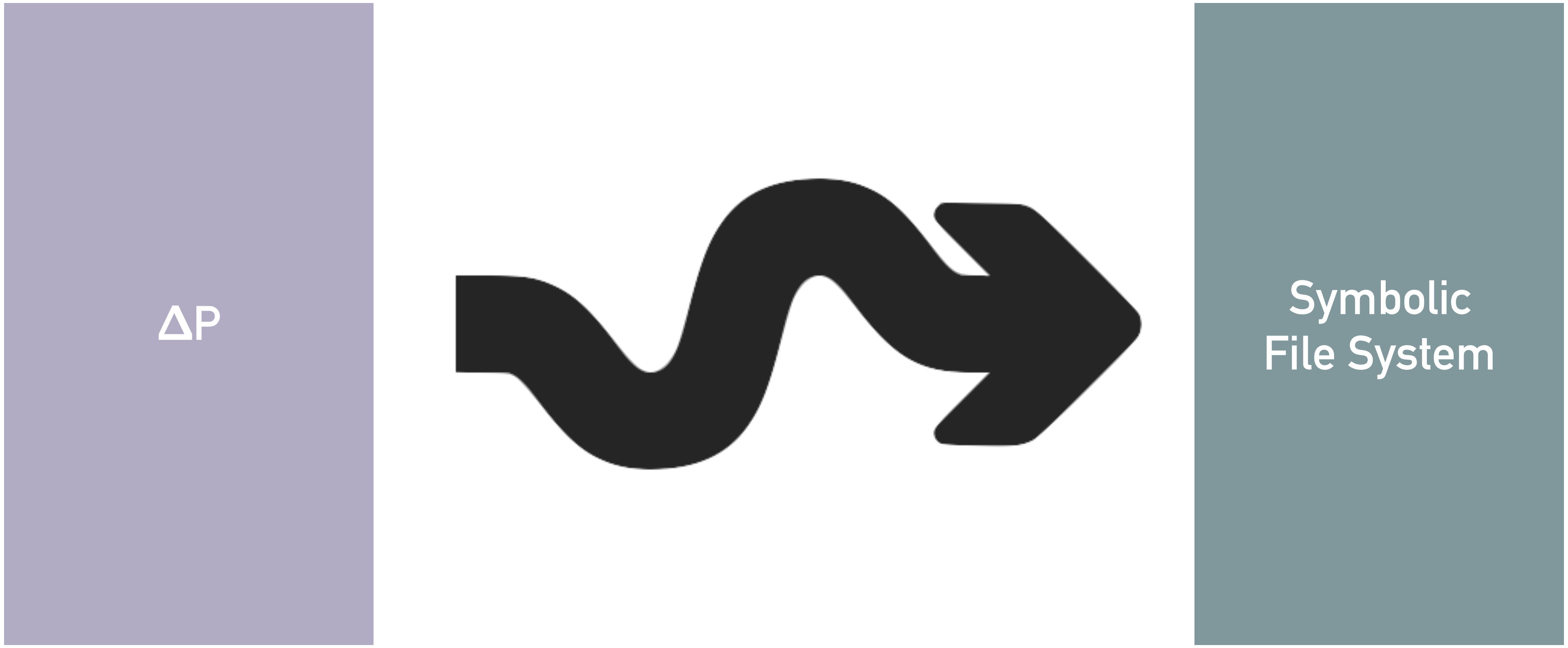
ΔP

FROM ΔP TO SMT



ΔP

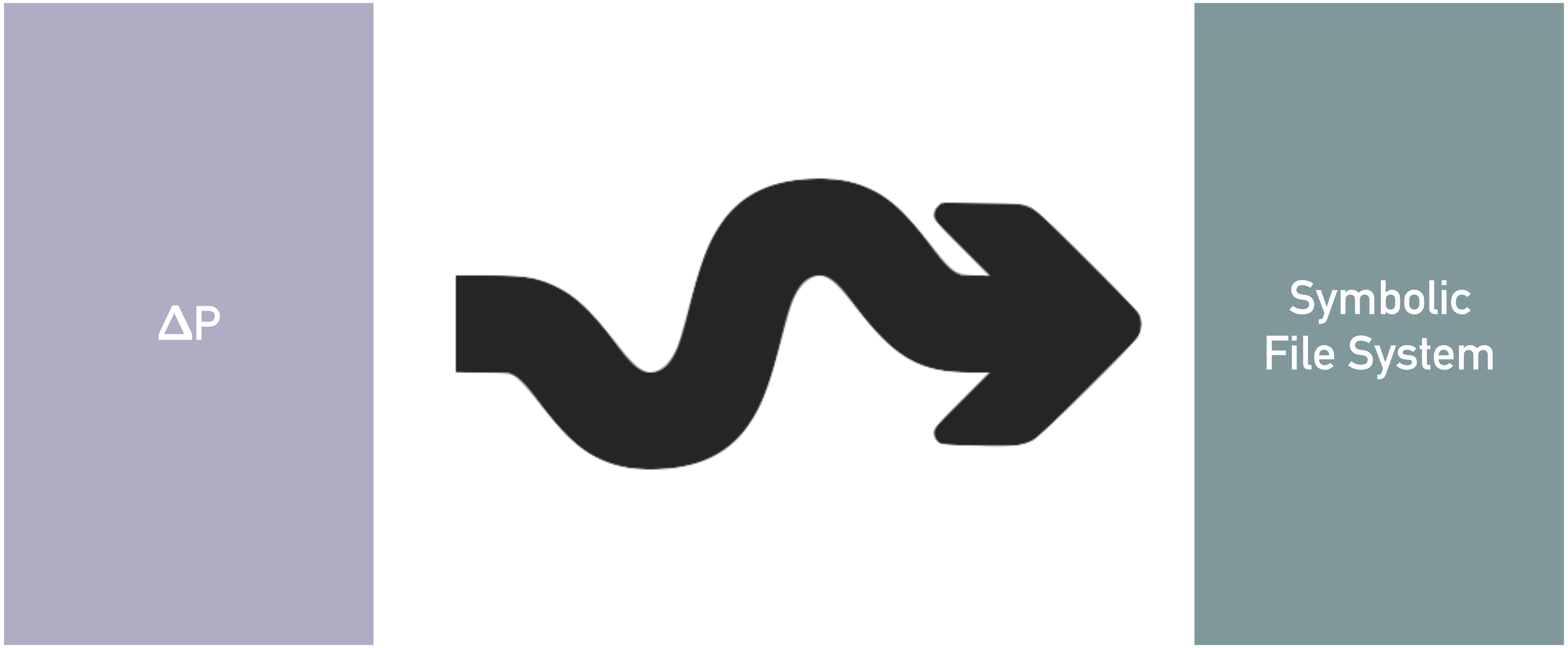
FROM ΔP TO SMT



ΔP

Symbolic File System

FROM ΔP TO SMT

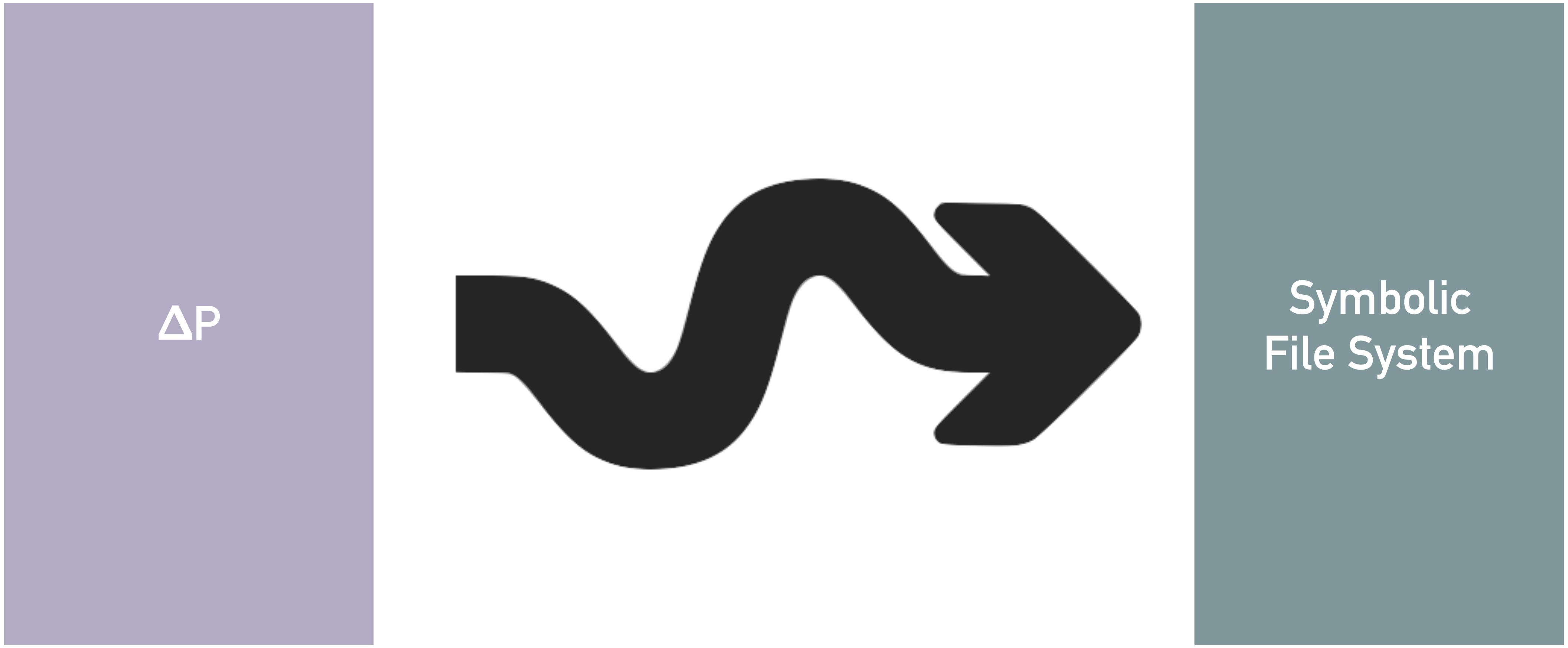


ΔP

Symbolic File System

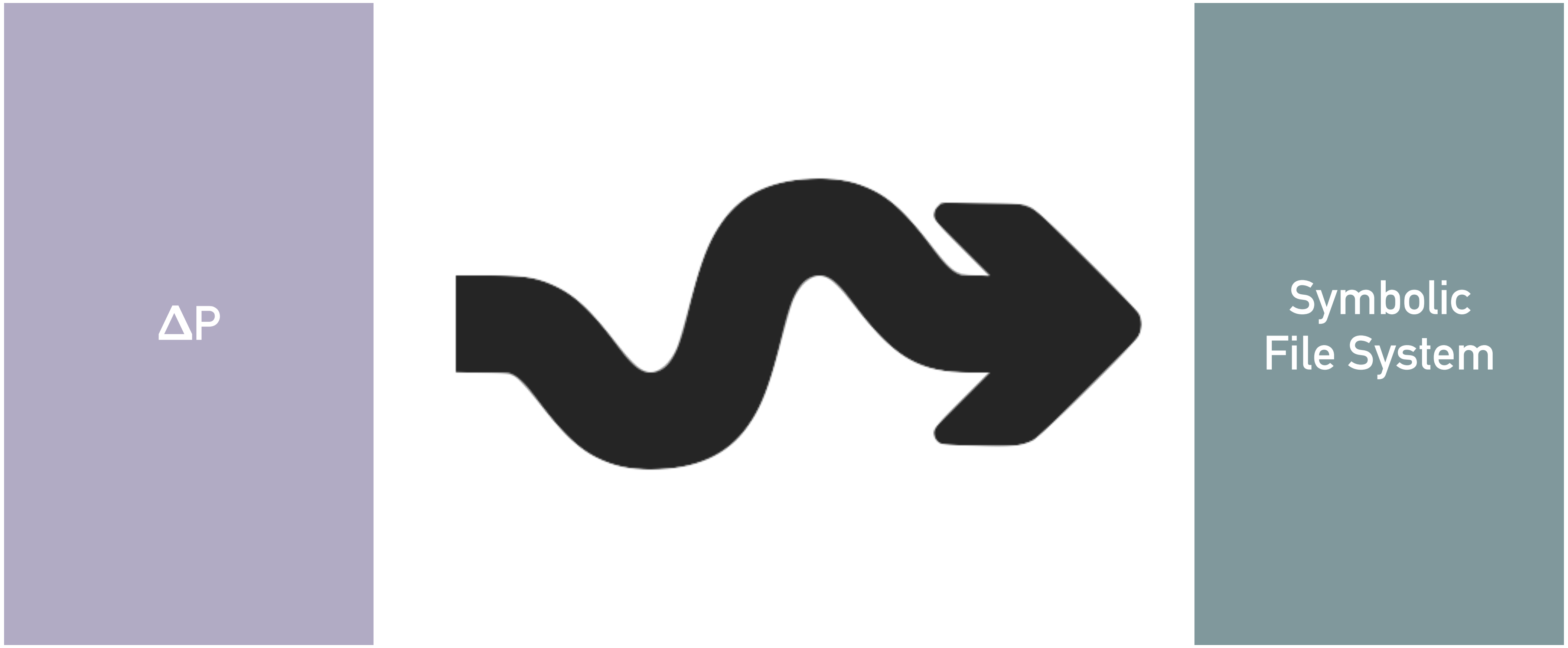
ϕ

FROM ΔP TO SMT



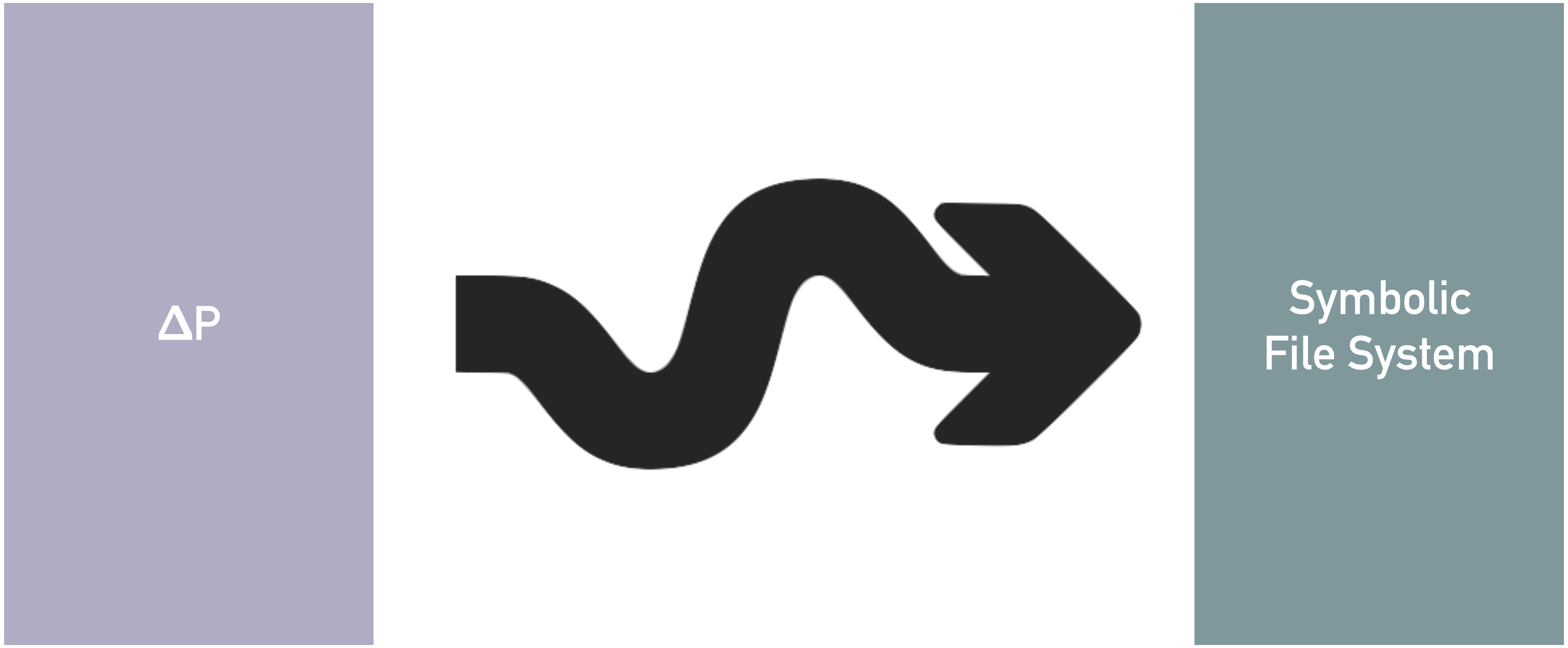
$$\phi(\qquad\qquad\qquad)$$

FROM ΔP TO SMT



$$\phi(\vec{fs}_{in}, \qquad\qquad)$$

FROM ΔP TO SMT

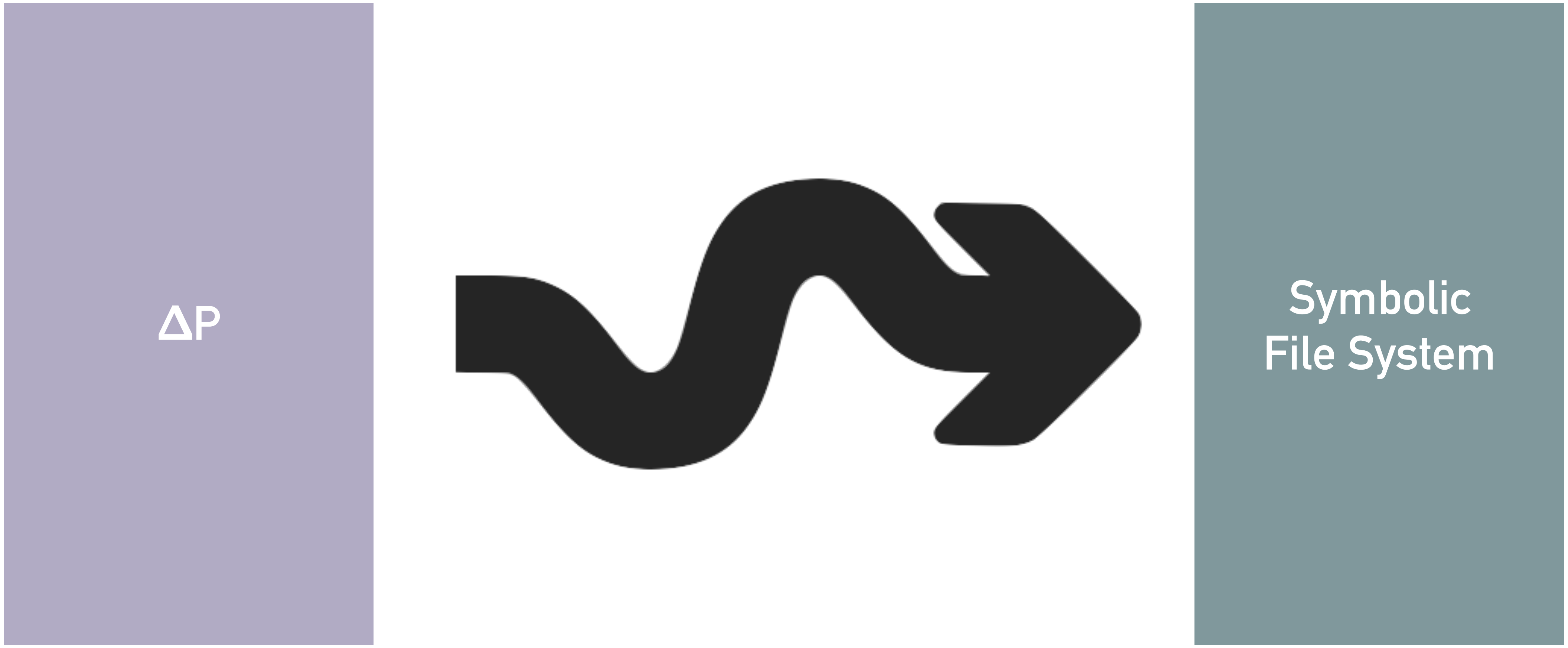


ΔP

Symbolic
File System

$$\phi(\overrightarrow{fs}_{in}, \overrightarrow{fs}_{out}, \qquad)$$

FROM ΔP TO SMT

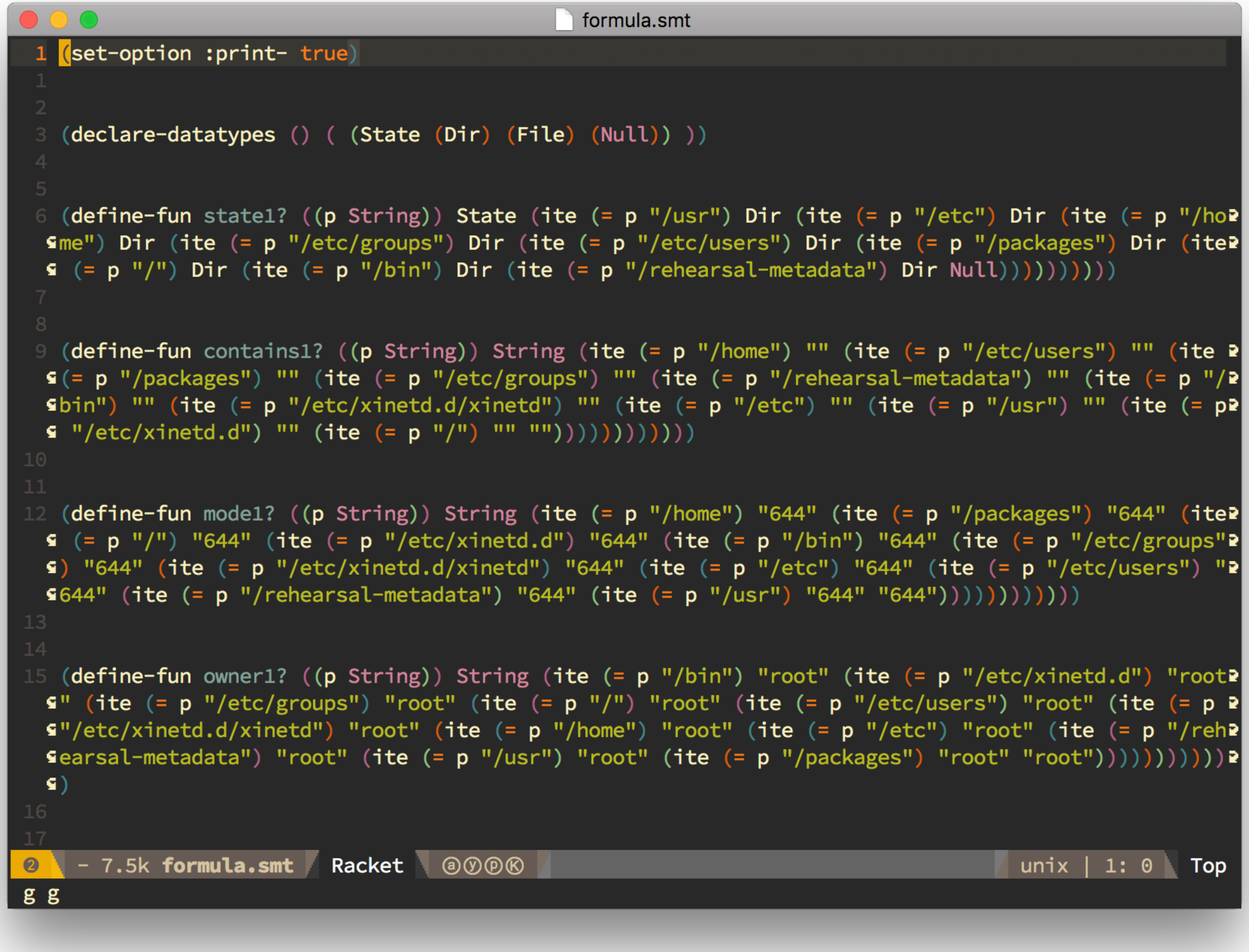


ΔP

Symbolic File System

$$\phi(\overrightarrow{fs}_{in}, \overrightarrow{fs}_{out}, \overrightarrow{x})$$

FORMULAE GET VERY LARGE



```
(set-option :print- true)


(declare-datatypes () ( (State (Dir) (File) (Null)) ))


(define-fun state1? ((p String)) State (ite (= p "/usr") Dir (ite (= p "/etc") Dir (ite (= p "/home") Dir (ite (= p "/etc/groups") Dir (ite (= p "/etc/users") Dir (ite (= p "/packages") Dir (ite (= p "/") Dir (ite (= p "/bin") Dir (ite (= p "/rehearsal-metadata") Dir Null))))))))))


(define-fun contains1? ((p String)) String (ite (= p "/home") "" (ite (= p "/etc/users") "" (ite (= p "/packages") "" (ite (= p "/etc/groups") "" (ite (= p "/rehearsal-metadata") "" (ite (= p "/bin") "" (ite (= p "/etc/xinetd.d/xinetd") "" (ite (= p "/etc") "" (ite (= p "/usr") "" (ite (= p "/etc/xinetd.d") "" (ite (= p "/") "" ""))))))))))))


(define-fun mode1? ((p String)) String (ite (= p "/home") "644" (ite (= p "/packages") "644" (ite (= p "/") "644" (ite (= p "/etc/xinetd.d") "644" (ite (= p "/bin") "644" (ite (= p "/etc/groups") "644" (ite (= p "/etc/xinetd.d/xinetd") "644" (ite (= p "/etc") "644" (ite (= p "/etc/users") "644" (ite (= p "/rehearsal-metadata") "644" (ite (= p "/usr") "644" "644"))))))))))))


(define-fun owner1? ((p String)) String (ite (= p "/bin") "root" (ite (= p "/etc/xinetd.d") "root" (ite (= p "/etc/groups") "root" (ite (= p "/") "root" (ite (= p "/etc/users") "root" (ite (= p "/etc/xinetd.d/xinetd") "root" (ite (= p "/home") "root" (ite (= p "/etc") "root" (ite (= p "/rehearsal-metadata") "root" (ite (= p "/usr") "root" (ite (= p "/packages") "root" "root"))))))))))))


```

MULTIPLE REPAIRS AND REPAIR RANKING

```
define rootdir($name) {
  file {"/$name":
    ensure => directory
  }
}
rootdir {
  name => “foo”
}
```

MULTIPLE REPAIRS AND REPAIR RANKING

```
define rootdir($name) {
  file {"/$name":
    ensure => directory
  }
}
rootdir {
  name => “foo”
}
```

```
awe@columba $ mv /foo /bar
```

MULTIPLE REPAIRS AND REPAIR RANKING

```
define rootdir($name) {
  file {"/$name":
    ensure => directory
  }
}
rootdir {
  name => "bar"
}
```

```
awe@columba $ mv /foo /bar
```

MULTIPLE REPAIRS AND REPAIR RANKING

```
define rootdir($name) {
  file {"/bar":
    ensure => directory
  }
}
rootdir {
  name => “foo”
}
```

```
awe@columba $ mv /foo /bar
```

MULTIPLE REPAIRS AND REPAIR RANKING

```
define rootdir($name) {
  file {"/bar":
```

```
1. rootdir { name => "foo" } BECOMES rootdir { name => "bar" }
2. file {"/$name": BECOMES file {"/bar":
```

```
}
rootdir {
  name => “foo”
}
```

```
awe@columba $ mv /foo /bar
```

THE REPAIR SCENARIO EXPERIMENT

THE REPAIR SCENARIO EXPERIMENT

```
define amavis::config($spamchecker='true', ...) {
  amavis { spamassassin => $spamchecker }

  file {'/etc/amavis/conf.d/50-user':
    ensure  => present,
    content => ... }

  file {'/etc/amavis/conf.d/15-content_filter_mode':
    ensure  => present,
    content => ... } }

define amavis($spamassassin='true') {
  package {'amavisd-new':
    ensure => present,
    alias  => 'amavis' }

  if $spamassassin == 'true' {
    package {'spamassassin': ensure  => present }
  } }

amavis::config { spamchecker => 'false' }
```

THE REPAIR SCENARIO EXPERIMENT

```
awe@columba $ sudo apt install spamassassin
```

```
define amavis::config($spamchecker='true'      ) {
  am
  
  f
    e
    content => ... }

  file {'/etc/amavis/conf.d/15-content_filter_mode':
    ensure  => present,
    content => ... } }

define amavis($spamassassin='true') {
  package {'amavisd-new':
    ensure => present,
    alias  => 'amavis' }

  if $spamassassin == 'true' {
    package {'spamassassin': ensure  => present }
  } }

amavis::config { spamchecker => 'false' }
```

THE REPAIR SCENARIO EXPERIMENT

```
define amavis::config($spamchecker='true'        ) {
  am
  
  f
    e
    content => ... }

  file {'/etc/amavis/conf.d/15-content_filter_mode':
    ensure  => present,
    content => ... } }

define amavis($spamassassin='true') {
  package {'amavisd-new':
    ensure => present,
    alias  => 'amavis' }

  if $spamassassin == 'true' {
    package {'spamassassin': ensure  => present }
  } }

amavis::config { spamchecker => 'false' }
```

```
awe@columba $ sudo apt install spamassassin
awe@columba $ synth
```

THE REPAIR SCENARIO EXPERIMENT

```
define amavis::config($spamchecker='true'        ) {
  am
  
  f
    e
    content => ... }

  file {'/etc/amavis/conf.d/15-content_filter_mode':

    ensure => present,
    alias  => 'amavis' }

  if $spamassassin == 'true' {
    package {'spamassassin': ensure  => present }
  } }

amavis::config { spamchecker => 'false' }
```

```
awe@columba $ sudo apt install spamassassin
awe@columba $ synth
```

```
1. spamchecker => 'false' REMOVED
2. spamchecker => 'false' BECOMES spamchecker => 'true'
3. $spamassassin == 'true' BECOMES 'false' == 'true'
```

THE REPAIR SCENARIO EXPERIMENT

```
define amavis::config($spamchecker='true'      ) {
  am
```

```
awe@columba $ sudo apt install spamassassin
awe@columba $ synth
```

```
  f
    e
    content => ... }

  file {'/etc/amavis/conf.d/15-content_filter_mode':
```

```
1. spamchecker => 'false' BECOMES spamchecker => 'true'
2. spamchecker => 'false' REMOVED
3. $spamassassin == 'true' BECOMES 'false' == 'true'
```

```
    ensure => present,
    alias  => 'amavis' }

  if $spamassassin == 'true' {
    package {'spamassassin': ensure  => present }
  } }

amavis::config { spamchecker => 'false' }
```

THE REPAIR SCENARIO EXPERIMENT

```
define amavis::config($spamchecker='true'      ) {
  am
```

```
awe@columba $ sudo apt install spamassassin
awe@columba $ synth
```

```
  f
    e
    content => ... }

  file {'/etc/amavis/conf.d/15-content_filter_mode':
```

```
1. spamchecker => 'false' BECOMES spamchecker => 'true'
2. spamchecker => 'false' REMOVED
3. $spamassassin == 'true' BECOMES 'false' == 'true'
```

```
    ensure => present,
    alias  => 'amavis' }

  if $spamassassin
    package {'spa                          esent }
  } }

amavis::config { spamchecker => 'false' }
```

Pick best repair: 2

EVALUATING TORTOISE

Benchmarks from Shambaugh, Weiss, and Guha (PLDI 2016)

Benchmark	# of resources	# of repair scenarios	Tortoise runtime (s)	Average repair rank
amavis	6	1	0.025	1.00
bind	6	3	0.021	1.60
clamav	6	2	0.023	3.50
hosting	19	1	0.026	1.00
irc	18	1	0.292	1.00
jpa	10	1	0.021	1.00
logstash	14	6	0.048	1.00
monit	7	4	0.025	1.00
nginx	9	4	0.027	1.00
ntp	4	3	0.018	1.33
powerdns	5	7	0.039	1.43
rsyslog	7	4	0.129	1.25
xinetd	4	5	1.970	1.20
Total	**115**	**42**		

EVALUATING TORTOISE

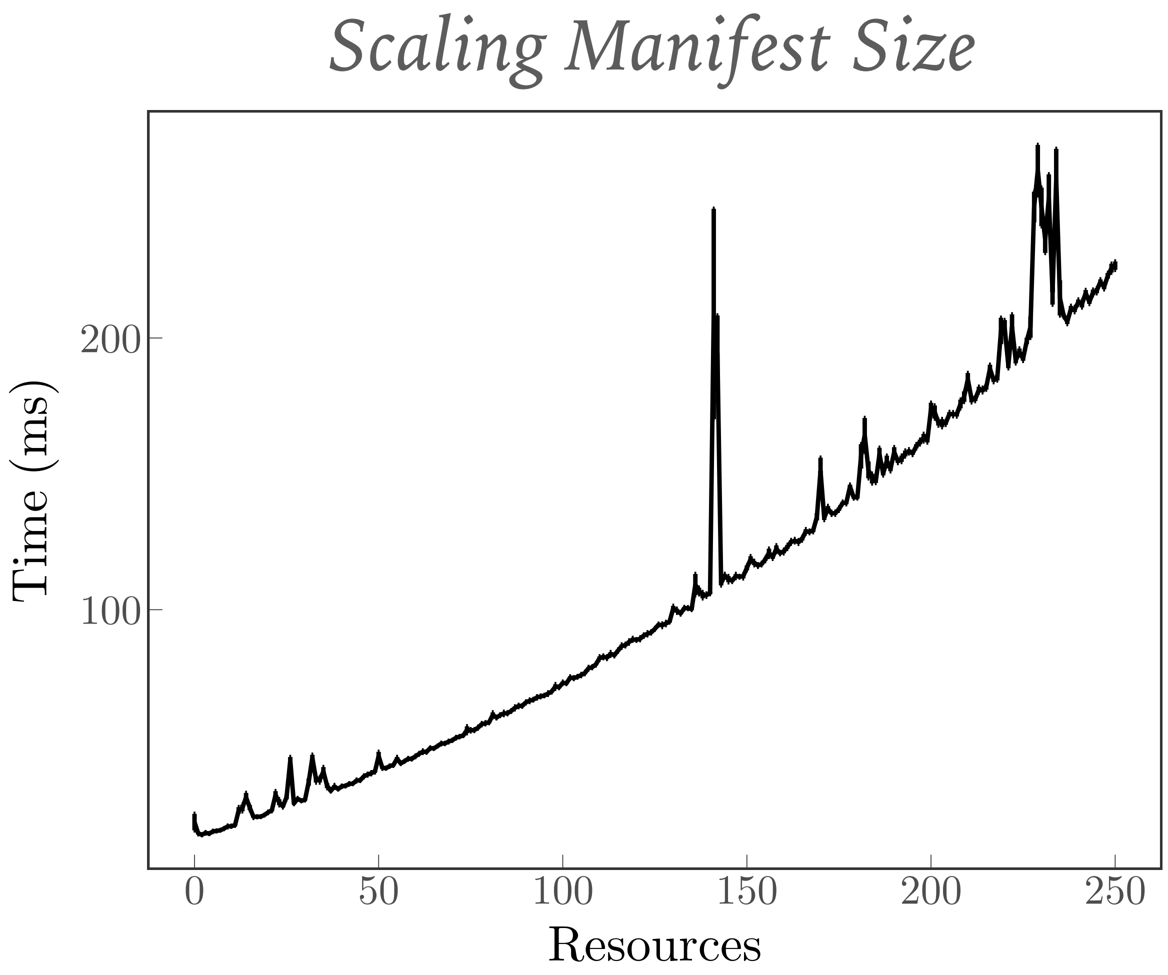
Benchmarks from Shambaugh, Weiss, and Guha (PLDI 2016)

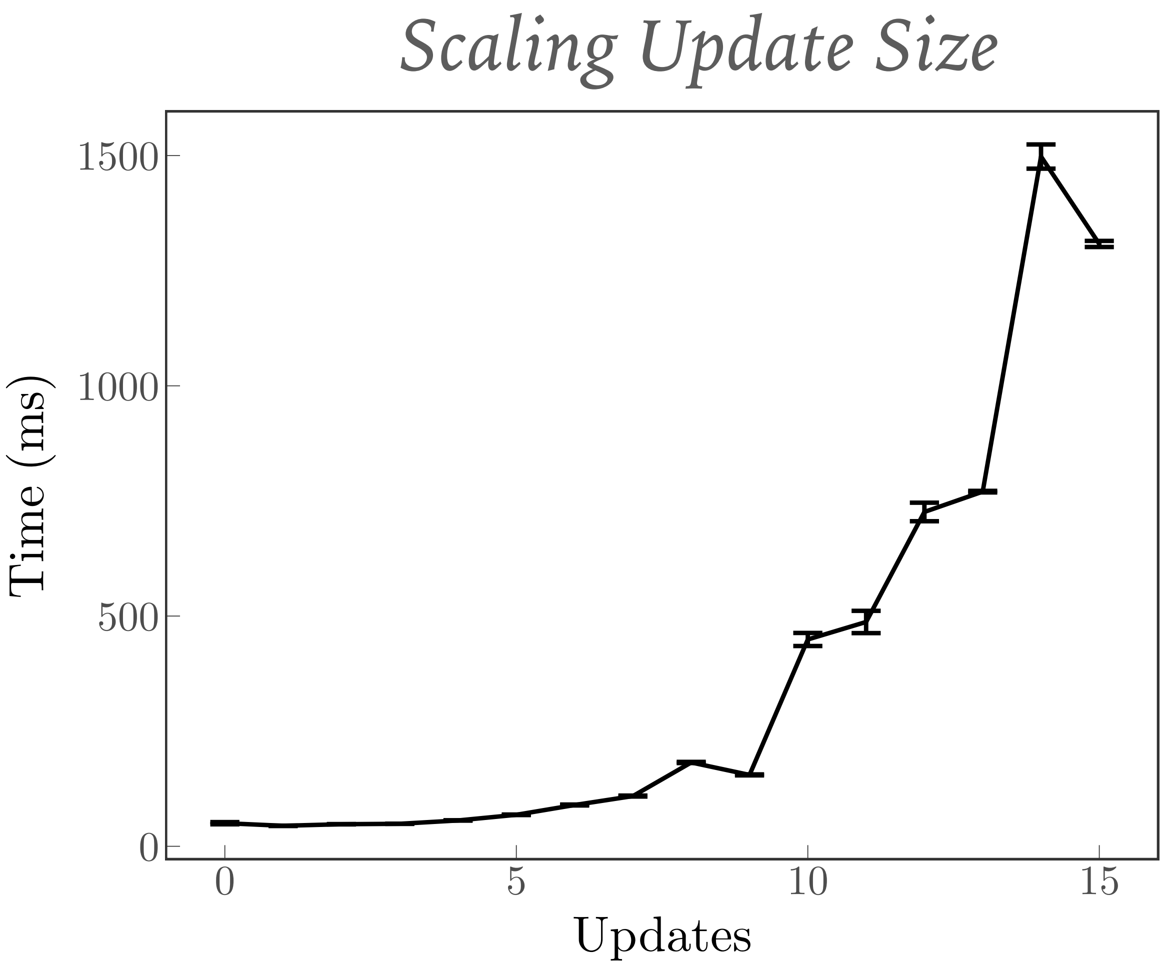
Benchmark	# of resources	# of repair scenarios	Tortoise runtime (s)	Average repair rank
amavis	6	1	0.025	1.00
bind				1.60
clama				3.50
hosti				.00
irc				.00
jpa				.00
logsta				.00
mon				.00
ngin				.00
ntp				1.33
powerdns				1.43
rsyslog	7	4	0.129	1.25
xinetd	4	5	1.970	1.20
Total	**115**	**42**		

Average Runtime: 0.2 seconds

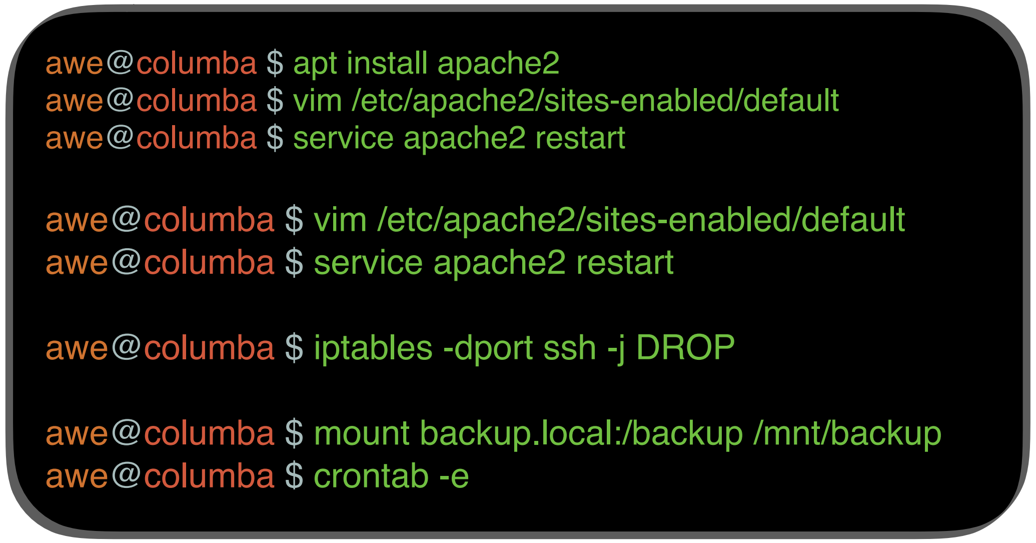
Average Rank: 1.31

TORTOISE SCALABILITY





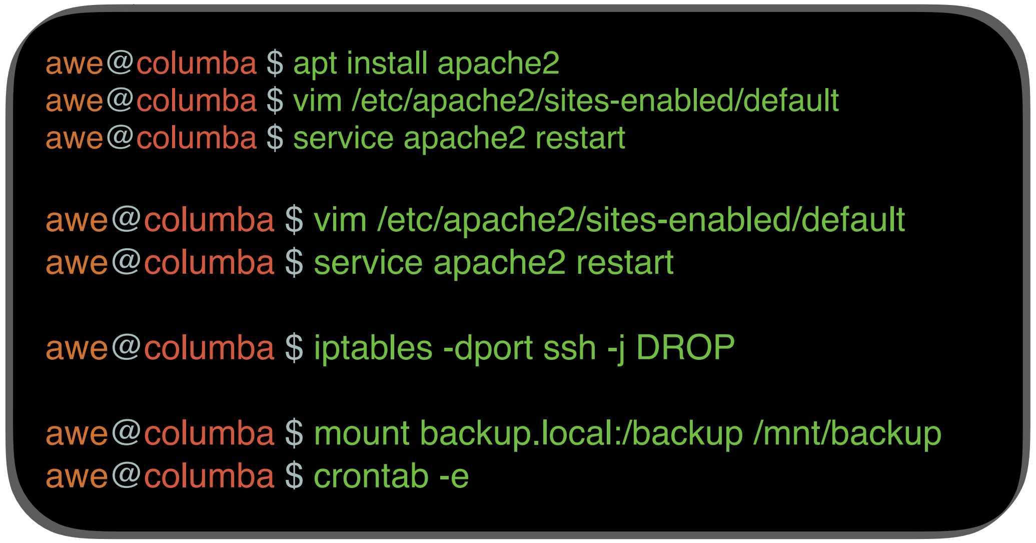
SUMMARY: IMPERATIVE CONFIGURATION REPAIR



```
awe@columba $ apt install apache2
awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ iptables -dport ssh -j DROP

awe@columba $ mount backup.local:/backup /mnt/backup
awe@columba $ crontab -e
```

SUMMARY: IMPERATIVE CONFIGURATION REPAIR

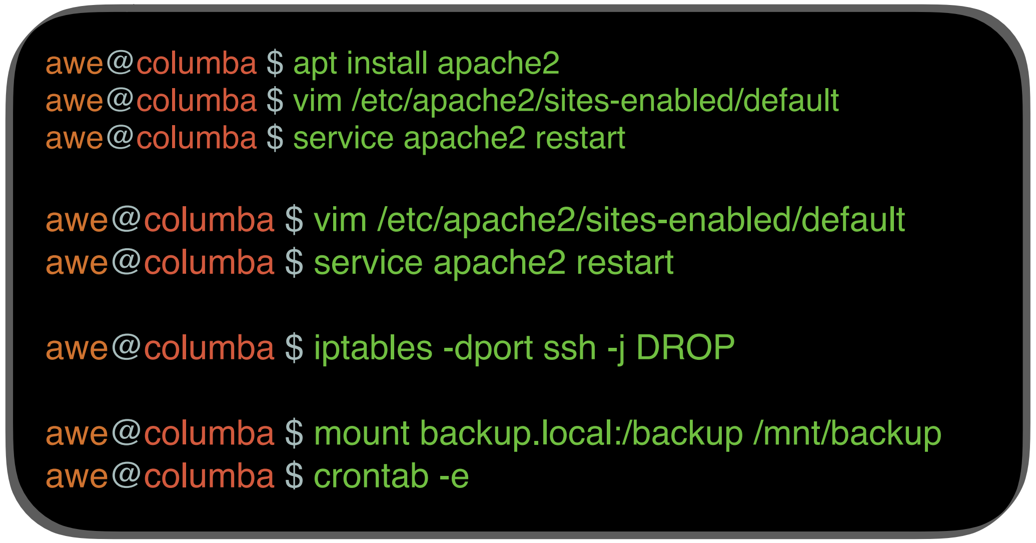
- Propagates changes made via the shell back to the original manifest!



```
awe@columba $ apt install apache2
awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ iptables -dport ssh -j DROP

awe@columba $ mount backup.local:/backup /mnt/backup
awe@columba $ crontab -e
```

SUMMARY: IMPERATIVE CONFIGURATION REPAIR

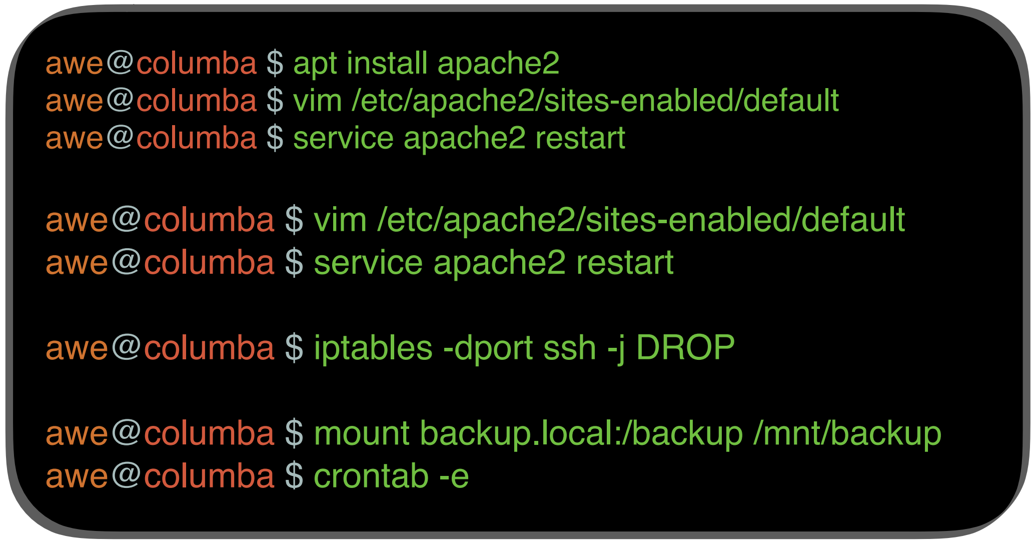
- Propagates changes made via the shell back to the original manifest!
  - Consistent: All changes made via the shell are preserved



```
awe@columba $ apt install apache2
awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ iptables -dport ssh -j DROP

awe@columba $ mount backup.local:/backup /mnt/backup
awe@columba $ crontab -e
```

SUMMARY: IMPERATIVE CONFIGURATION REPAIR

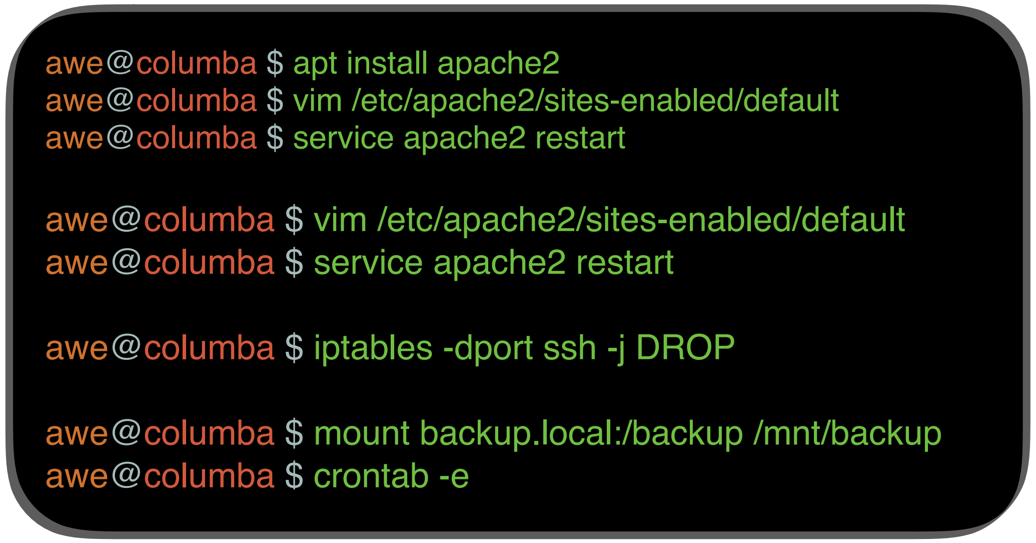
- Propagates changes made via the shell back to the original manifest!
  - Consistent: All changes made via the shell are preserved
  - Maintainable: Structure and abstraction is preserved



```
awe@columba $ apt install apache2
awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ iptables -dport ssh -j DROP

awe@columba $ mount backup.local:/backup /mnt/backup
awe@columba $ crontab -e
```

SUMMARY: IMPERATIVE CONFIGURATION REPAIR

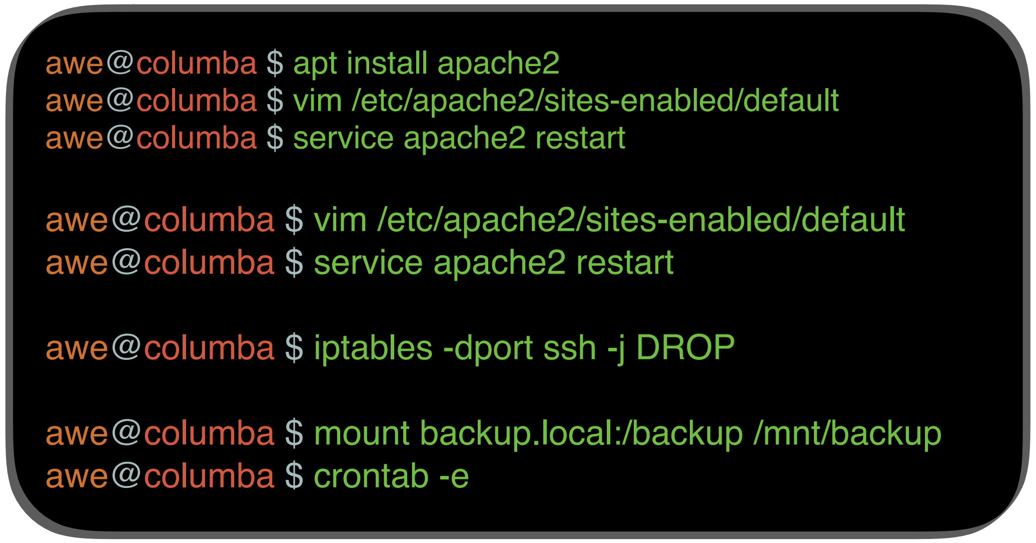
- Propagates changes made via the shell back to the original manifest!
  - Consistent: All changes made via the shell are preserved
  - Maintainable: Structure and abstraction is preserved
  - Ranked: Multiple possible repairs are ranked



```
awe@columba $ apt install apache2
awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ iptables -dport ssh -j DROP

awe@columba $ mount backup.local:/backup /mnt/backup
awe@columba $ crontab -e
```

SUMMARY: IMPERATIVE CONFIGURATION REPAIR

- Propagates changes made via the shell back to the original manifest!
  - Consistent: All changes made via the shell are preserved
  - Maintainable: Structure and abstraction is preserved
  - Ranked: Multiple possible repairs are ranked
  - Unrestricted: Works with all existing shells

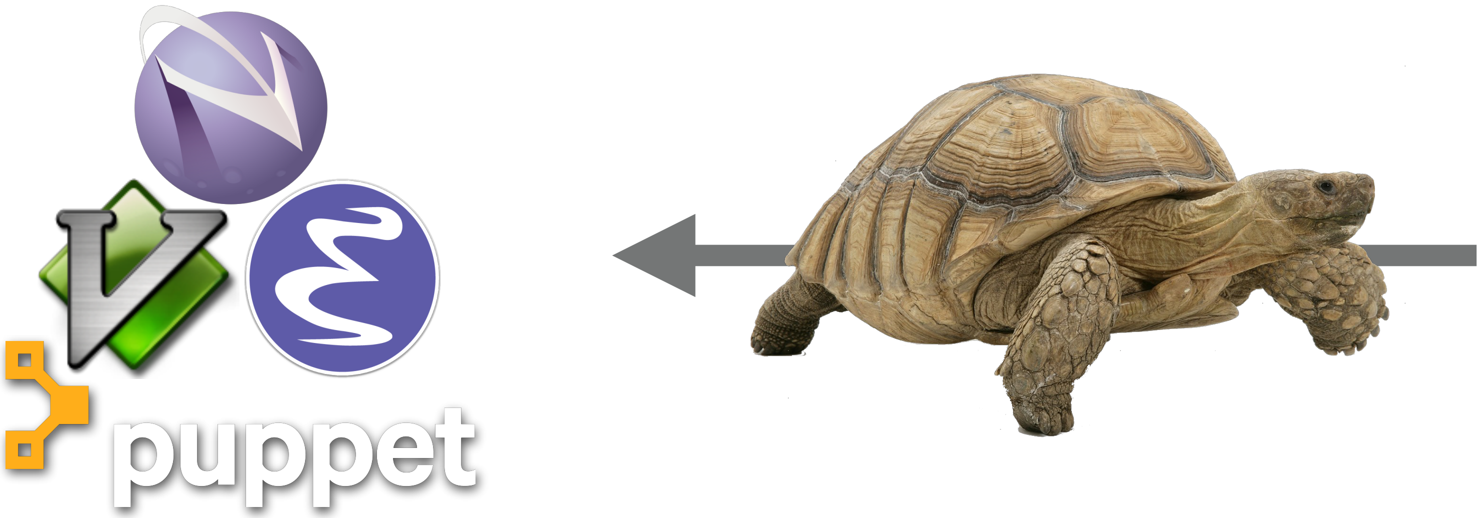


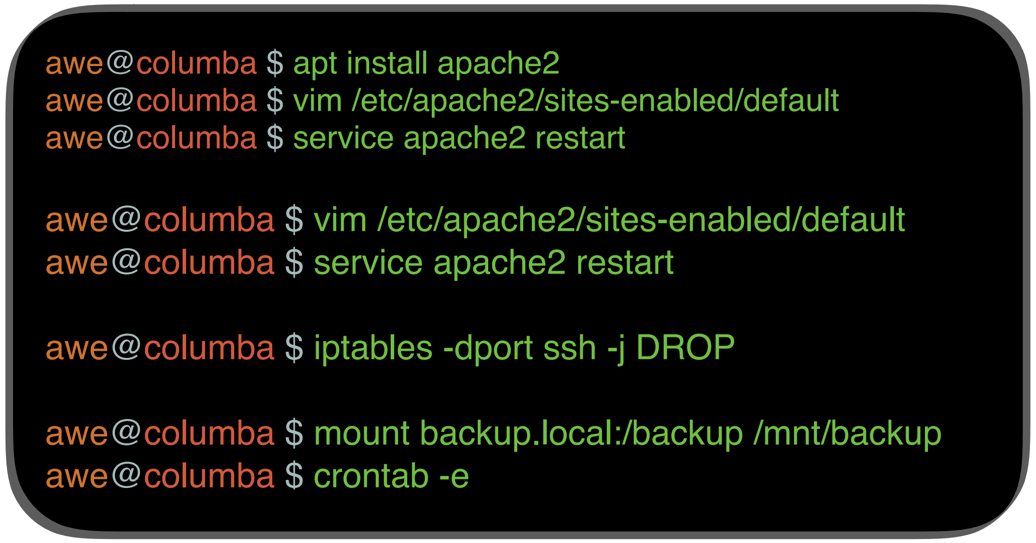
```
awe@columba $ apt install apache2
awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ iptables -dport ssh -j DROP

awe@columba $ mount backup.local:/backup /mnt/backup
awe@columba $ crontab -e
```

SUMMARY: IMPERATIVE CONFIGURATION REPAIR

- Propagates changes made via the shell back to the original manifest!
    - Consistent: All changes made via the shell are preserved
    - Maintainable: Structure and abstraction is preserved
    - Ranked: Multiple possible repairs are ranked
    - Unrestricted: Works with all existing shells

Live Programming for Configuration Management





```
awe@columba $ apt install apache2
awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ vim /etc/apache2/sites-enabled/default
awe@columba $ service apache2 restart

awe@columba $ iptables -dport ssh -j DROP

awe@columba $ mount backup.local:/backup /mnt/backup
awe@columba $ crontab -e
```

TESTING AND VERIFYING PROPERTIES OF SYSTEM CONFIGURATIONS

- Configuration Management Tools (Infrastructure as Code)
  - Rehearsal (Shambaugh, Weiss, and Guha. PLDI 2016)
  - Citac (Hanappi et al. OOPSLA 2016)
  - Idempotency for Chef (Hummer et al. Middleware 2013)
- Application-Specific Configurations
  - ConfValley (Huang et al. EuroSys 2015)
  - PCheck (Xu et al. OSDI 2016)
- Specific Configurations via Analysis
  - ConfSuggester (Zhang and Ernst. ICSE 2014)
  - Autobash (Su et al. SOSP 2007)