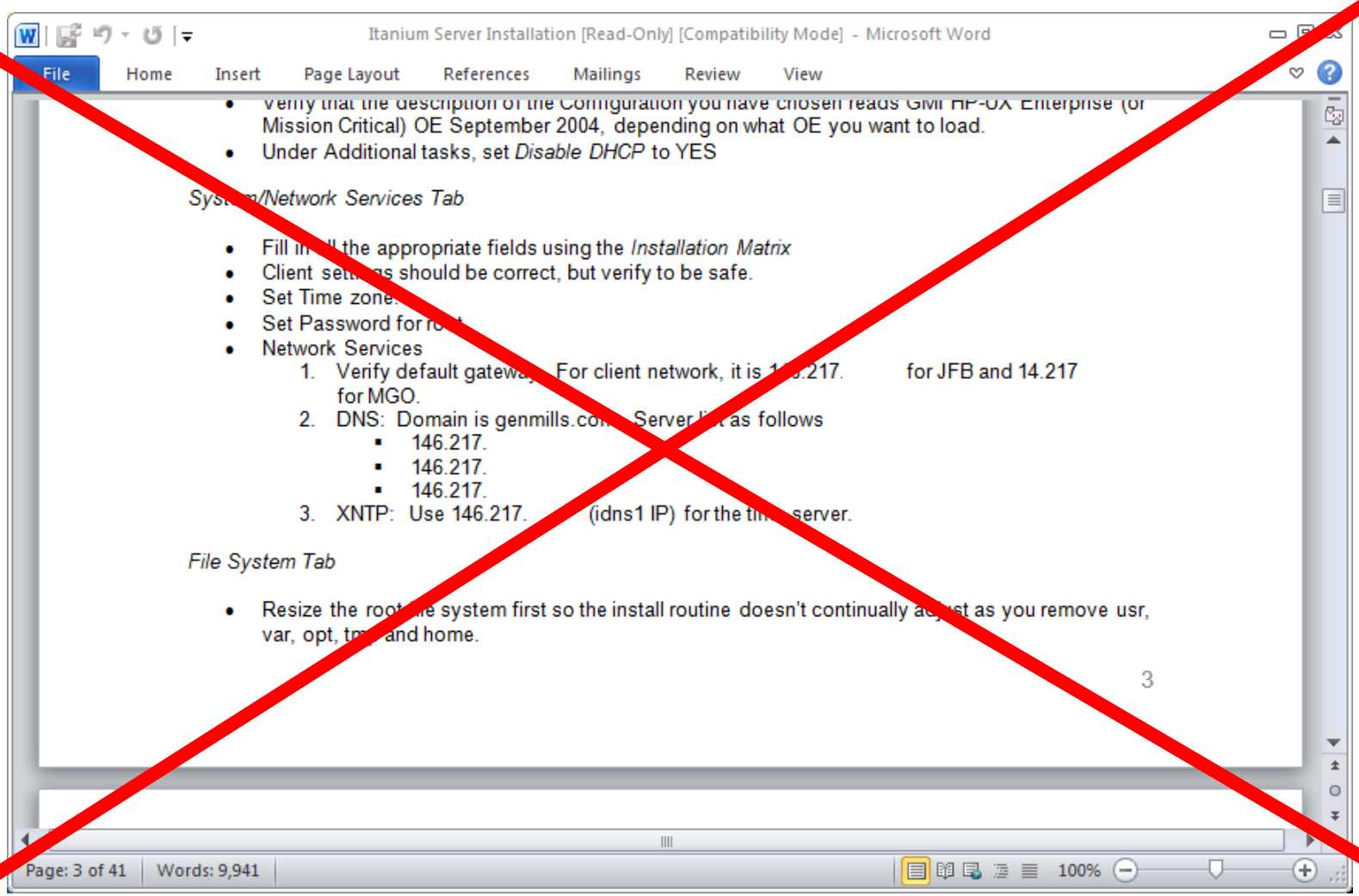
PUPPET

Use at General Mills

Preface

- HP UX platform at GMI is 15+ years old
- Consolidated Superdome architecture today
- Moving enterprise apps to RHEL6
  - Oracle
  - SAP
  - BW/BI
  - Warehouse Management
- Short migration timeframe

Preface


Itanium Server Installation [Read-Only] [Compatibility Mode] - Microsoft Word
File
Home
Insert
Page Layout
References
Mailings
Review
View
Mission Critical) OE September 2004, depending on what OE you want to load.
Under Additional tasks, set Disable DHCP to YES
Network Services Tab
Fill in all the appropriate fields using the Installation Matrix
Client settings should be correct, but verify to be safe.
Set Time zone.
Set Password for root
Network Services
Verify default gateway. For client network, it is 146.217. for JFB and 14.217 for MGO.
DNS: Domain is genmills.com. Server list as follows
146.217.
146.217.
146.217.
XNTP: Use 146.217. (idns1 IP) for the time server.
File System Tab
Resize the root file system first so the install routine doesn't continually adjust as you remove usr, var, opt, tmp and home.
3
Page: 3 of 41
Words: 9,941
100%

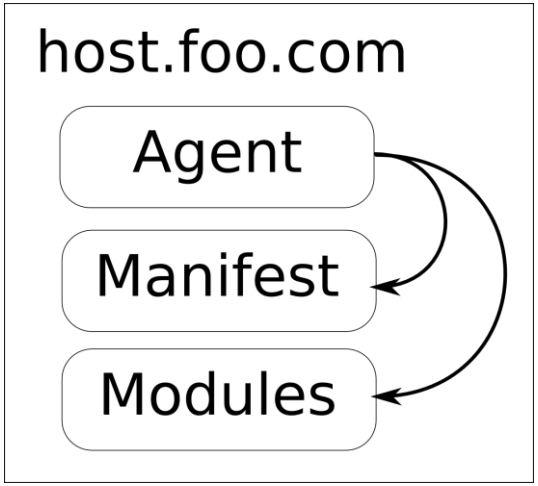
Topics

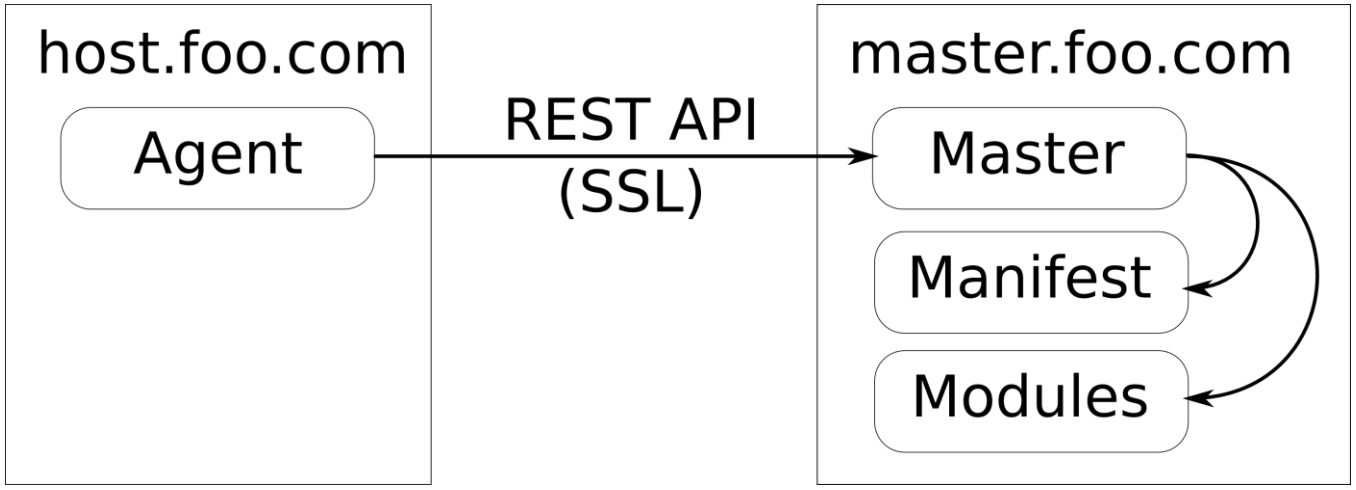
- Puppet basics
- Usage at GMI
- Rough spots
- Questions

What is Puppet?

- Configuration management
  - Files
  - Software packages
  - Users/groups
  - Consistent interface for wide selection of OSes
  - Action by declaration
- Multiple uses
  - Run-once provisioning
  - Continuous compliance
  - Audit

Components


host.foo.com
Agent
Manifest
Modules


host.foo.com
Agent
REST API
(SSL)
master.foo.com
Master
Manifest
Modules

Common Resource Types

- file
- user
- group
- mount
- package
- service
- exec
- nagios_*
- ssh_authorized_key
- tidy
- yumrepo
- augeas
- cron

Language Example

```
user { 'httpd':
      ensure => present,
      uid => 80, gid => 80,
      groups => ['users', 'engr'],
      comment => 'Apache User'
}
package { 'emacs': ensure => absent }
```

Language Example

```
service { 'ntpd':
      ensure => running,
      enable => true
}
file { 'ntp.conf':
      path => '/etc/ntp.conf',
      content => template('ntp/ntp.erb'),
      notify => Service['ntpd']
}
```

Language Example

```
class ntp {
      package { 'ntp': … }
      file { 'ntp.conf': …
            require => Package['ntp']
      }
      service { 'ntpd': …
            require => File['ntp.conf']
      }
}
```

Language Example (cont.)

```
node 'appserver1.genmills.com' {
      include 'ntp'
      include 'kerberos'
      class { 'net':
            search => 'genmills.com'
      }
      net::iface { 'eth0':
            address => '3.3.3.3/24',
            mtu => 1500
      }
}
```

RHEL6 Install

Main RPMs from PuppetLabs:

http://yum.puppetlabs.com/el/6/products/x86_64/

- puppet.noarch
  - Agent/client
- puppet-server.noarch
  - Master/server
- facter.x86_64
  - Agent data collection
  - Pure Ruby despite arch tag

RHEL6 Install

Augeas from RHN server-optional channel:

- augeas.$ARCH
  - Structure config file manipulations

EPEL for ruby-augeas:

- http://dl.fedoraproject.org/pub/epel/6/x86_64/repoview/ruby-augeas.html
- ruby-augeas.noarch
  - Ruby bindings

Resources

PuppetLabs

http://docs.puppetlabs.com/puppet/

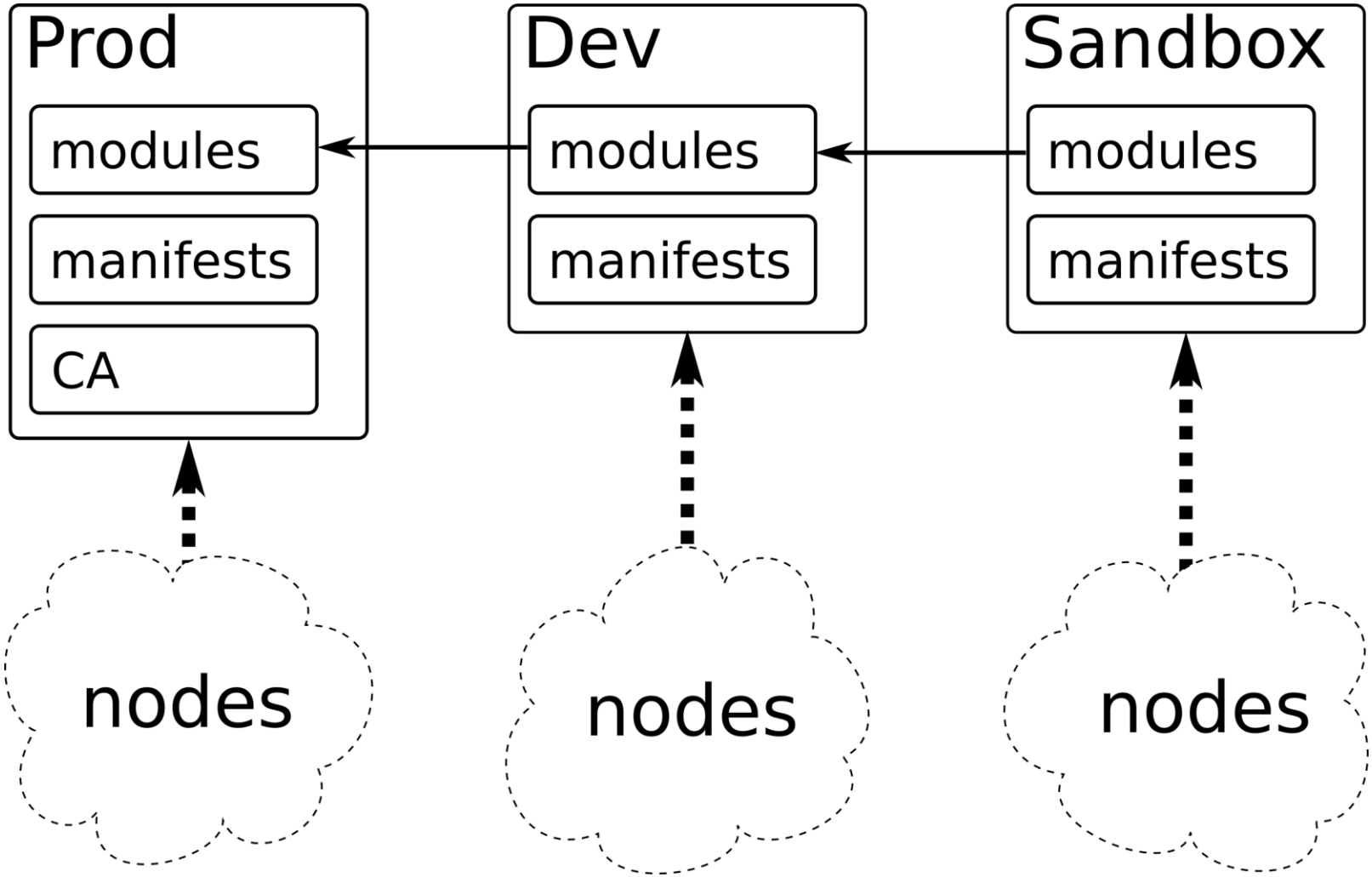
Pro Puppet

ISBN - 978-1430230571

Puppet at GMI

- Initial provisioning via RHN Satellite
- No machine-specific configuration in Kickstart
- All RHEL hosts provisioned/controlled this way
- Running 2.7.x agents/masters
  - Headed to 3.x series
- sysadmin by declaration, not action

Puppet at GMI



Rough spots

- Resource sharing
- Source control workflow
- Node inheritance (with classes)

Resource Sharing

```
class sap_server {
     package { 'compat-libstdcpp-33':
          ensure => present
     }
}
class oracle_server {
     package { 'compat-libstdcpp-33':
          ensure => present
     }
}
```

Resource Sharing

- Puppet Labs stdlib for Puppet fixes this

```
class sap_server {
      ensure_resource('package',
            'compat-libstdcpp-33',
            { ensure => present }
      )
}
```

- Includes many utility functions
- https://github.com/puppetlabs/puppetlabs-stdlib

Node Inheritance

```
node base { … }
node 'host.com' inherits base { … }
```

- Good - Can be more simple than ENC or Hiera
- Bad - Discouraged by Puppet Labs documentation
- Ugly - Parameterized classes are problematic

Node Inheritance: Example

```
class appservice($secure) {
      if ($secure) {
            file { '/usr/app/secure': … }
      }
}

node base_node {
      class { 'appservice':
            secure => false
      }
}
```

Node Inheritance: Example

```
node 'box.genmills.com' inherits base_node {
      Class['appservice'] {
            secure => true
      }
}
```

- /usr/app/secure will not be created
- Class parameters aren't overridden between nodes

Node Inheritance: Hack

```
class appservice($secure) {
      if ($secure) {
            file { '/usr/app/secure': … }
      }
}
define appservice::instance($secure) {
      class { 'appservice':
            secure => $secure
      }
}
```

Node Inheritance: Hack Usage

```
node base_node {
      appservice::instance { 'appservice':
            secure => false
      }
}
node 'box.genmills.com' inherits base_node {
      Appservice::Instance['appservice'] {
            secure => true
      }
}
```

Node Inheritance: Worth it?

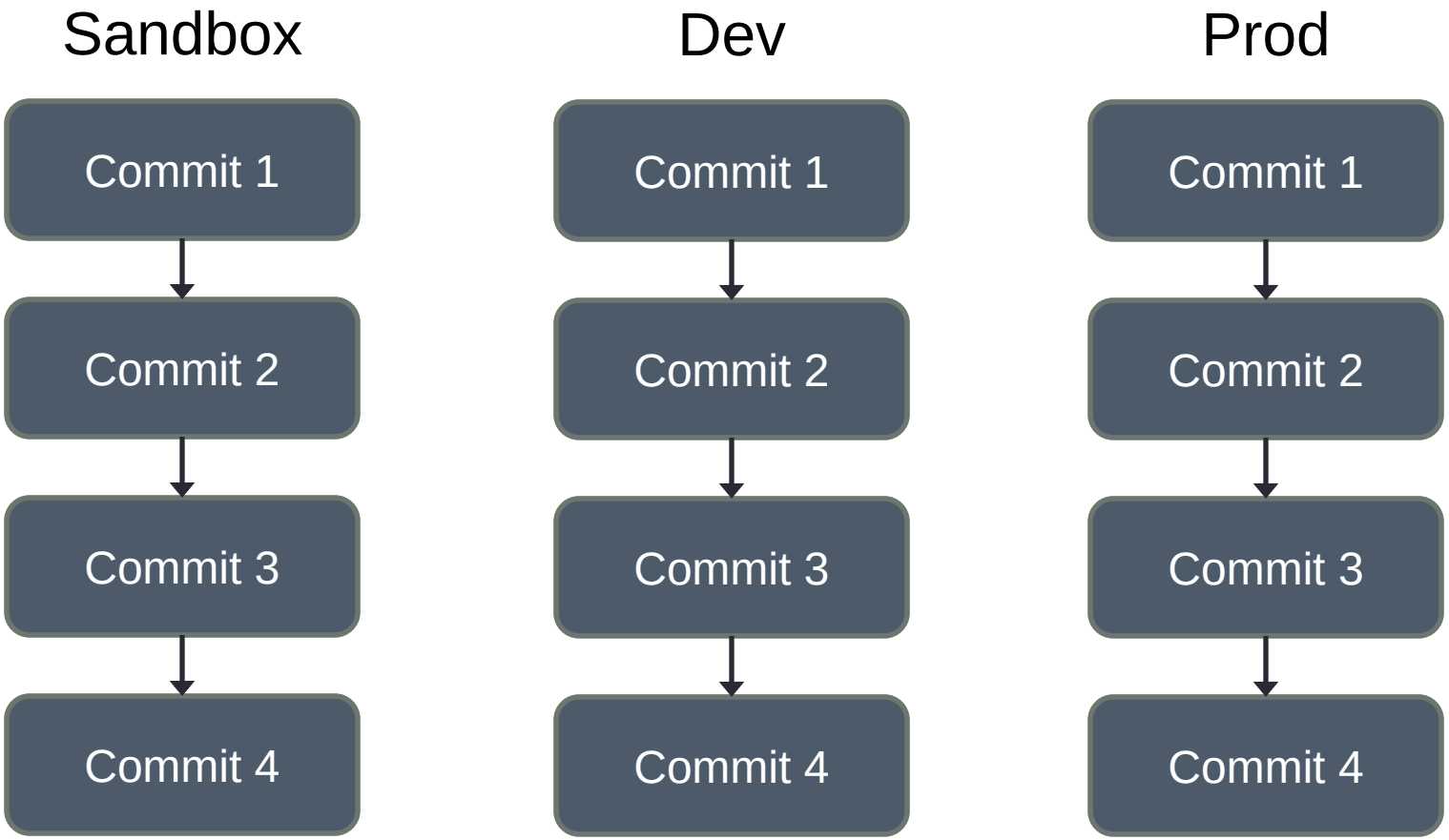
Caveats:

- Different syntax for invocation and alteration
- Class variables are inaccessible to outside
- Naming standards must be followed
- Language changes might have negative effects

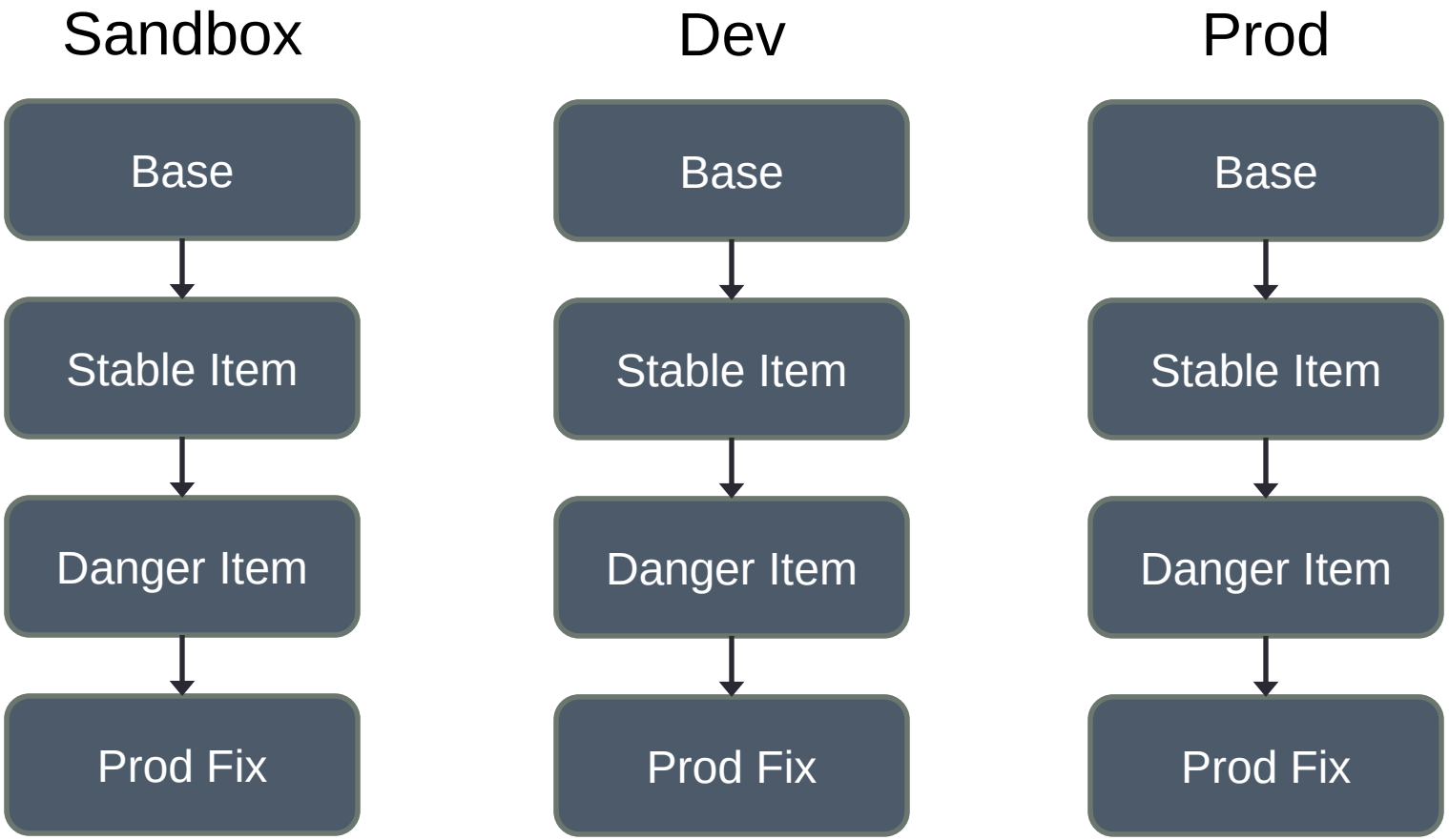
Workflow

- Source control is ***strongly*** recommended
  - Git is a popular choice
  - Steeper learning curve than "traditional" VCSs
  - Flexible structure lends itself well to the task
- Plan for change/feature promotion process
- Test isolation is a must

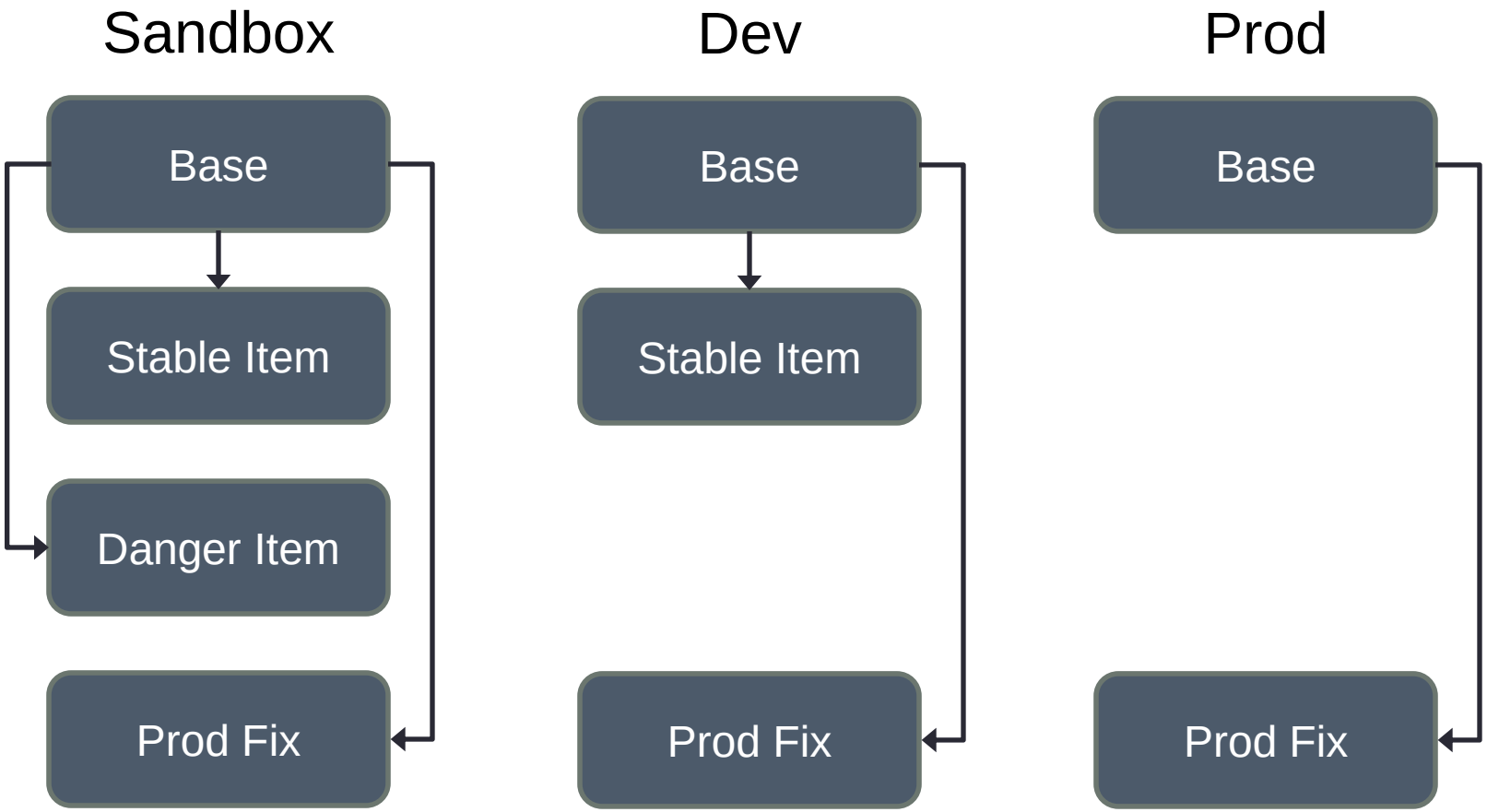
Workflow: Using git



Workflow: Failure



Workflow: Fixed



Workflow: How?

- Manipulating (meddling) with git history
  - git reset –hard <commit>
- Use clones, not branches, for safety
- Know how far back to turn the clock
- Automation in the works

Questions?