

Puppet on Windows

Yes, it works.

Gabriel M Schuyler
gabe@puppetlabs.com
@gabe_sky



Hi.

Me:

- Professional Services engineer,
- Linux and Windows Admin,
- Keeping people honest.

You:

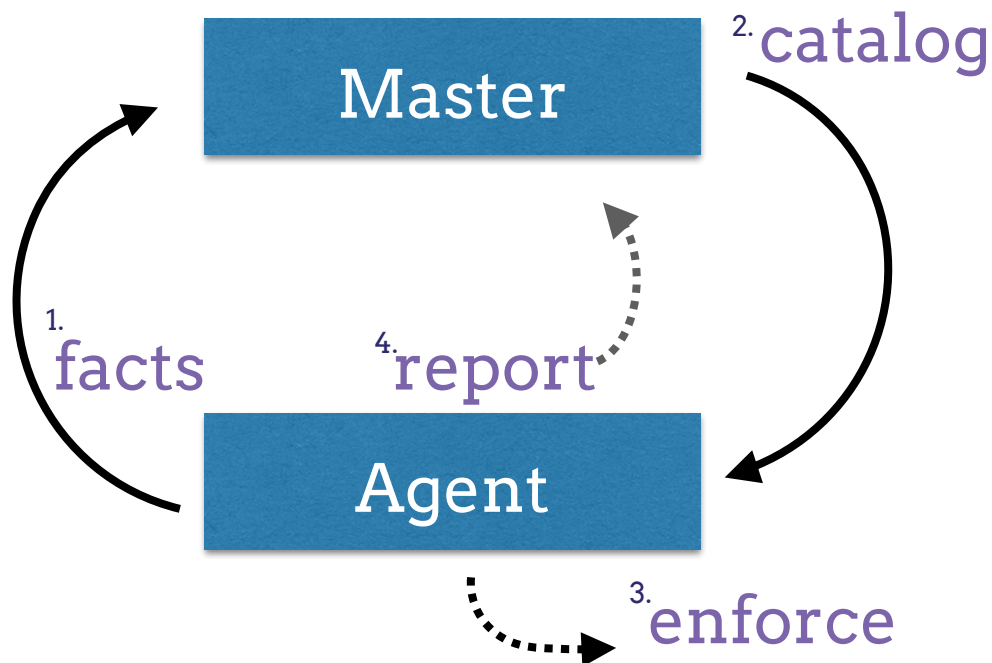
- Windows Admins?
- Puppet Users?

Overview

- Agents & Masters
- Built-In Types
- Package Management
- Service Management
- Managing the Registry
- Leveraging PowerShell
- Managing Reboots
- Using the Forge

Briefly

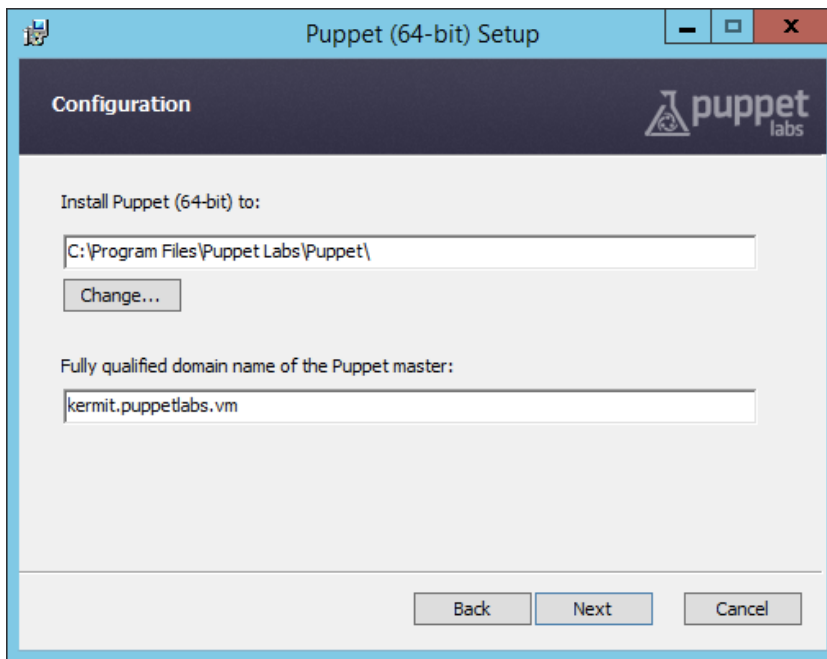
- Model individual "resources,"
- Grouped into reusable "classes."
- Compile a customized "catalog" so the
- Agent can enforce a "desired state"
- using the Resource Abstraction Layer.



```
class render {  
  
  user { 'wrangler':  
    ensure => present,  
    groups => ['Administrators']  
  }  
  package { 'maya3d':  
    ensure => installed,  
    provider => chocolatey,  
  }  
  file { 'C:/ProgramData/Maya':  
    ensure => directory,  
  }  
  file { 'C:/ProgramData/Maya/licenses':  
    ensure => file,  
    source => 'puppet:///modules/r  
  }  
  service { 'maya-control':  
    ensure => running,  
    enable => true,  
  }  
}
```

Installation

Agent

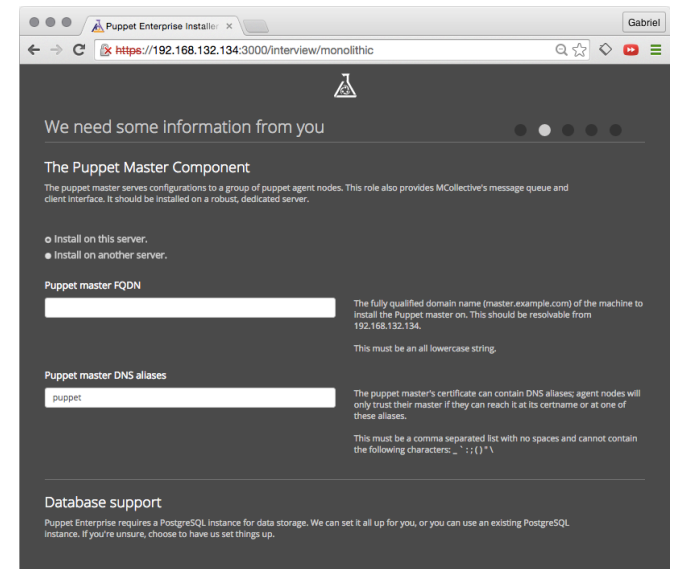


Master

free:

```
rpm -ivh http://yum.puppetlabs.com/...  
yum install puppet-server  
vi /etc/puppet/puppet.conf  
# add firewall rules if needed  
# start the services
```

enterprise:



(10 nodes free)



Command Prompt

```
puppet resource <type> [title]
```

```
puppet describe <type>
```

```
puppet apply <filename>
```

```
puppet agent -t
```

Built-In Types

"Mostly what you'd expect."

Users & Groups

```
class gms::users {  
  
  user { 'gabe':  
    ensure => present,  
    comment => 'Gabriel M Schuyler',  
    gid => '123',  
    groups => ['Administrators', 'Users'],  
    password => 'MyPassword6',  
  }  
  
  group { 'gonzo':  
    ensure => present,  
    members => ['gabe'],  
  }  
  
}
```

no 'gid'

'groups'
instead

'password'
required,
to create

'groups'
alternative

Files

```
class gms::file {  
  File {  
    owner => 'Administrator',  
    group => 'Administrators',  
    source_permissions => ignore,  
  }  
  
  file { 'C:/Temp':  
    ensure => directory,  
  }  
  file { 'C:/Temp/gabe.txt':  
    ensure => file,  
    source => 'puppet:///modules/gms/gabe.txt',  
  }  
}
```

don't try to use
master's
permissions

forward slashes &
consistent
capitalization

Packages

```
class gms::package {  
  
  package { 'Minecraft':  
    ensure => installed,  
    source => 'C:/MinecraftInstaller.msi',  
  }  
  
  exec { 'install sublime text 3':  
    command => '"C:\Sublime Text Build 3065 x64 Setup.  
    exe" /SILENT /NORESTART',  
    creates => 'C:/Program Files/Sublime Text 3/  
    sublime_text.exe',  
  }  
}
```

must match
DisplayName (in
the registry)

file on local disk or
network mount

hardcoded
command

best guess at
whether it's
installed

Don't panic! Help is on the way.

Services

```
class gms::service {  
  
  service { 'SharedAccess':  
    ensure => stopped,  
    enable => false,  
  }  
  
  file { 'C:/license_wrapper.bat':  
    ensure => file,  
    source => 'puppet:///modules/gms/license_wrapper.bat',  
  }  
  
  exec { 'start the license server':  
    command => 'C:\Windows\System32\cmd.exe C:\license_wrap  
    unless => 'C:\Windows\System32\cmd.exe C:\license_wrap  
    require => File['C:/license_wrapper.bat'],  
  }  
  
}
```

must match the
Service Name

Do-it-yourself
init script kludge

Start it, unless
it's already
running

Which Slash Where?

```
file { 'C:\thisway.txt':  
  ensure => file,  
  source => 'C:\some\local\path\file.txt',  
}  
  
file { 'C:/thatway.txt':  
  ensure => file,  
  source => 'C:/another/local/file.txt',  
}  
  
exec { 'C:\thisway.exe': }  
exec { 'C:/thatway.exe': }  
  
exec { 'C:\Windows\System\cmd.exe C:\my_script.bat': }  
  
package { 'sprinkles':  
  ensure => installed,  
  source => 'C:\backslashonly\sprinkles.msi'  
  install_options => '/s c:\lastrun.answers.txt'  
}
```

Windows paths
can be either way

Commands..
either way

but cmd.exe (e.g.)
expects backslash

source and options
must use
backslash

Is there a drive letter? Use backslash.

Windows Specific

"Let's do some *real* administration, now."

The Forge

The screenshot shows a web browser window with the URL `https://forge.puppetlabs.com/modules?utf-8=✓&sort=rank&q=windows+feature`. The page header includes navigation links: Open Source Projects, Docs, Support, Ask, Bug Tracker, Security, and Puppet Labs. The main header features the Puppet Forge logo, a search bar with the text "Search from 3,250 modules", and buttons for "Find", "Publish a Module", "Sign Up", and "Log In".

The search results section displays "Found 2 modules matching 'windows feature'". Below this, there are two modules listed:

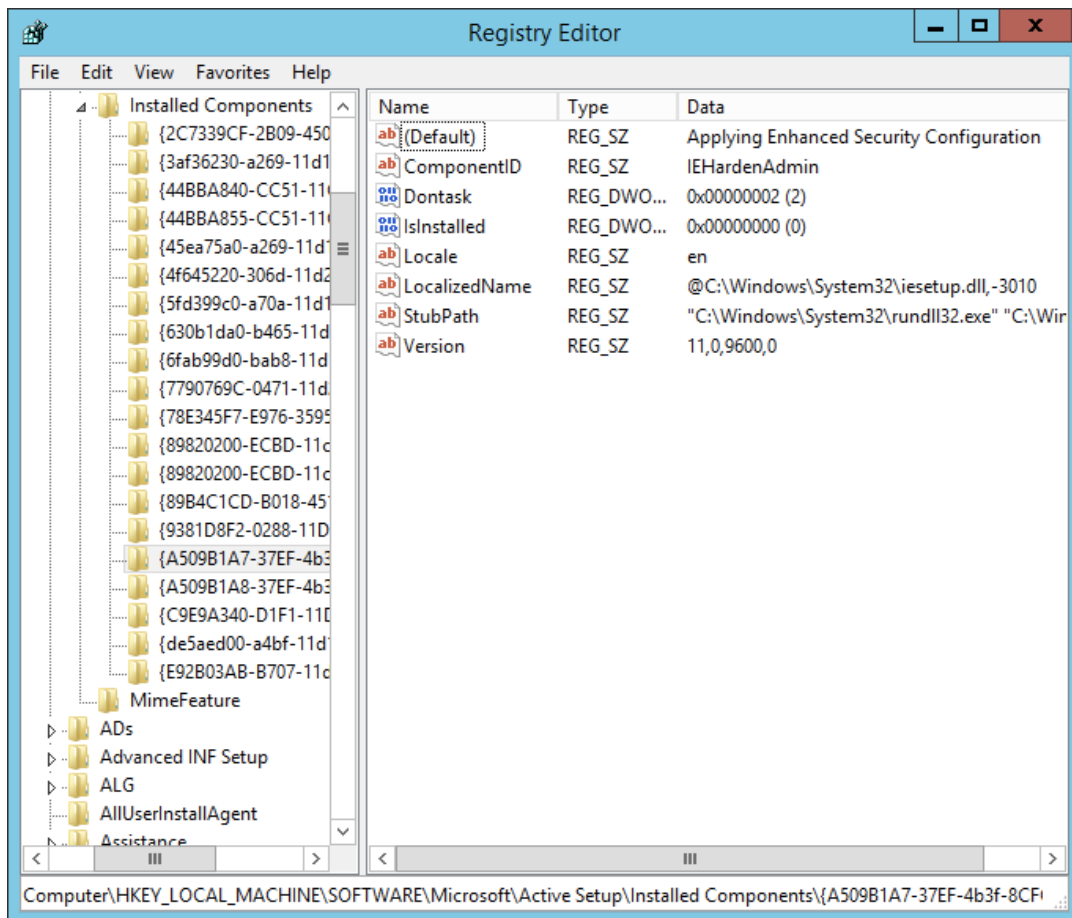
- opentable/windows_feature**: Module that will turn windows features on or off for Windows Server 2008 and above. Version 0.0.2 • Jul 7, 2014 • 833 downloads. 1,630 downloads, 1.0.
- admin36/winfeature**: Installs and Removes Windows Features. Version 0.0.3 • Dec 9, 2013 • 973 downloads. 1,337 downloads, 2.0.

Navigation controls include "« Prev", "1", and "Next »". Below the results, it says "Showing 1 - 2 of 2 results".

On the left side, there are filters for "Operating System", "Puppet Version", "Puppet Enterprise Version", and "Supported or Approved", each with a dropdown menu set to "- Any -". There is also a checkbox for "Include deleted modules" and an "Apply Filters" button.

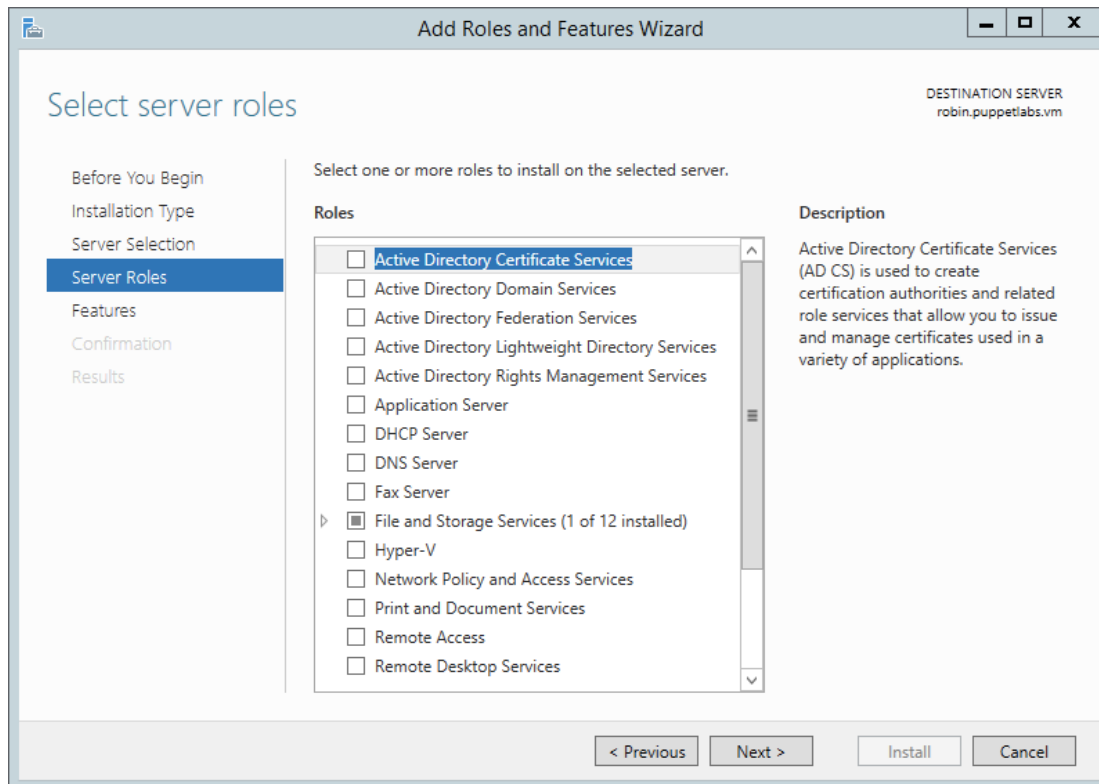
On the right side, there are two sections: "Puppet Supported" with links to `puppetlabs/postgresql`, `puppetlabs/haproxy`, `puppetlabs/mysql`, `puppetlabs/f5`, and `puppetlabs/concat`; and "Puppet Approved" with links to `camptocamp/openssl` and `elasticsearch/elasticsearch`. Both sections have "Learn more" and "View all" links.

puppetlabs-registry



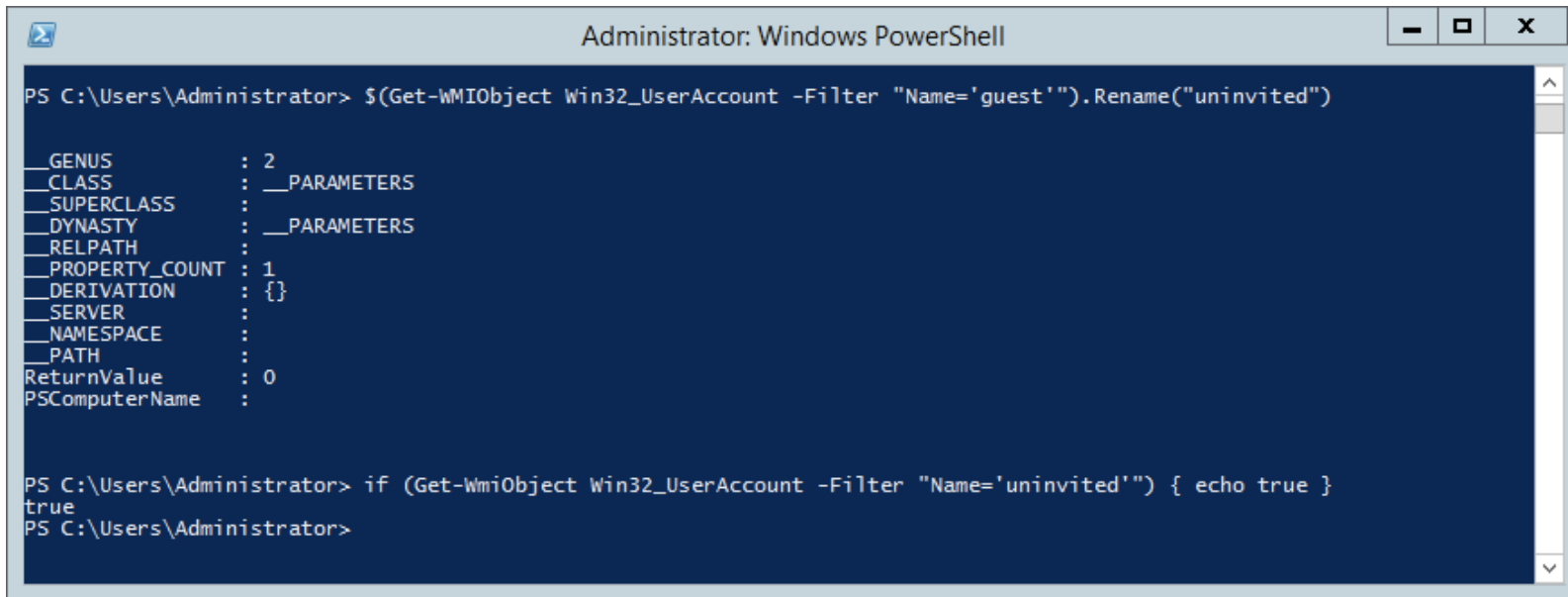
```
registry::value { 'My App Setting':  
  key   => 'HKLM\Software\Vendor\Puppet',  
  value => 'ThisOne',  
  data  => 'Share and Enjoy',  
}
```

puppetlabs-dism



```
dism { 'IIS-Webserver':  
  ensure => present,  
  all    => true,  
}
```


puppetlabs-powershell



```
Administrator: Windows PowerShell

PS C:\Users\Administrator> $(Get-WMIObject Win32_UserAccount -Filter "Name='guest'").Rename("uninvited")

__GENUS           : 2
__CLASS           : __PARAMETERS
__SUPERCLASS      : 
__DYNASTY         : __PARAMETERS
__RELPATH         : 
__PROPERTY_COUNT  : 1
__DERIVATION      : {}
__SERVER          : 
__NAMESPACE       : 
__PATH            : 
ReturnValue       : 0
PSComputerName    : 

PS C:\Users\Administrator> if (Get-WmiObject Win32_UserAccount -Filter "Name='uninvited'") { echo true }
true
PS C:\Users\Administrator>
```

single
quoted
'command'

```
class gms::powershell {

  exec { 'rename the guest account':
    command => '$(Get-WMIObject Win32_UserAccount -Filter "Name=\'guest\'").',
              Rename("uninvited")',
    unless => 'if (Get-WmiObject Win32_UserAccount -Filter "Name=\'guest\'") { exit 1 }',
    provider => 'powershell',
  }

}
```

puppetlabs-powershell

```
command      => # single quotes around this
cwd          => # from where
environment  => # extra variables to set

creates      => # check for a file before running
onlyif       => # a command returns zero
unless       => # a command returns non-zero
refreshonly  => # don't run unless notified

returns      => # if success returns a non-zero code
timeout      => # only wait this long, in seconds
logoutput    => # true, false or only 'onfailure'
```

puppetlabs-reboot

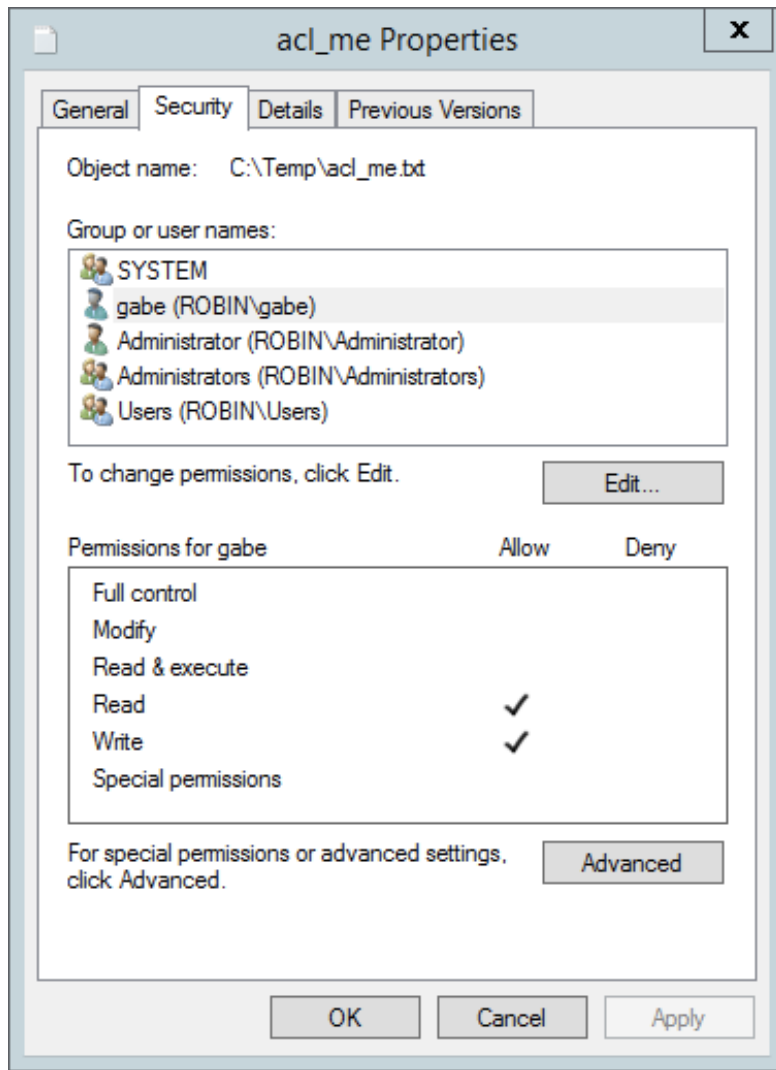
```
reboot { 'right away':  
  apply => immediately,  
  message => 'Puppet triggered a mid-run reboot.',  
  timeout => '60', # one minute  
}  
  
reboot { 'after the run':  
  apply => finished,  
  message => 'Puppet triggered a reboot after the run.',  
  catalog_apply_timeout => '7200', # two hours  
}  
  
reboot { 'before if pending':  
  when => pending,  
  message => 'Puppet triggered a mid-run (pending) reboot.',  
}
```

skip the rest,
do it *now*.

it can wait ..

if someone said "it can wait,"
skip the rest and do it now *anyway*.

puppetlabs-acl



```
class gms::acl {  
  
  file { 'C:\Temp\acl_me.txt':  
    ensure => file,  
    content => 'Set acls on me!',  
  }  
  
  acl { 'C:\Temp\acl_me.txt':  
    permissions => [  
      { identity => 'Administrator', rights => ['full'] },  
      { identity => 'Users', rights => ['read'] },  
      { identity => 'gabe', rights => ['read','write'] },  
    ],  
    purge => true,  
    inherit_parent_permissions => false,  
  }  
}
```

.. And More

liamjbennet-windows_autoupdate

liamjbennet-windows_firewall

trlinkin-domain_membership

opentable-iis

puppetlabs-sqlserver

nanliu-staging

Send Help

- Add windows compatibility -- or editions.
- Replace the "install and run away screaming" modules.
- Add more features that can be managed.

"Pull requests welcome."

Tools

"When all you have is a hammer."

Package Management



www.chocolatey.org

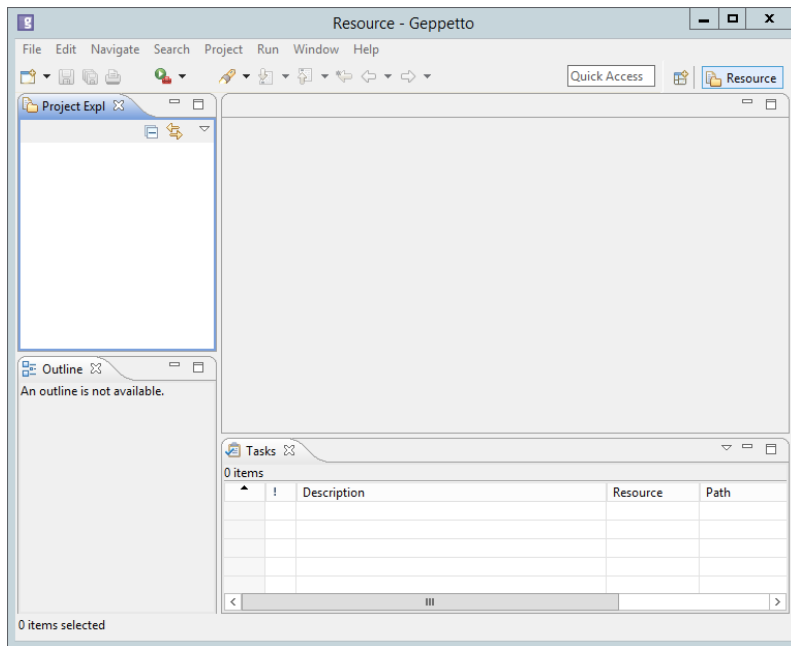
```
class gms::choco {  
  
    include chocolatey_sw  
    Package { require => Class['chocolatey_sw'], }  
  
    package { 'notepadplusplus':  
        ensure => installed,  
        provider => 'chocolatey',  
    }  
  
}
```

forge: ceritsc-chocolatey_sw

forge: rismoney-chocolatey

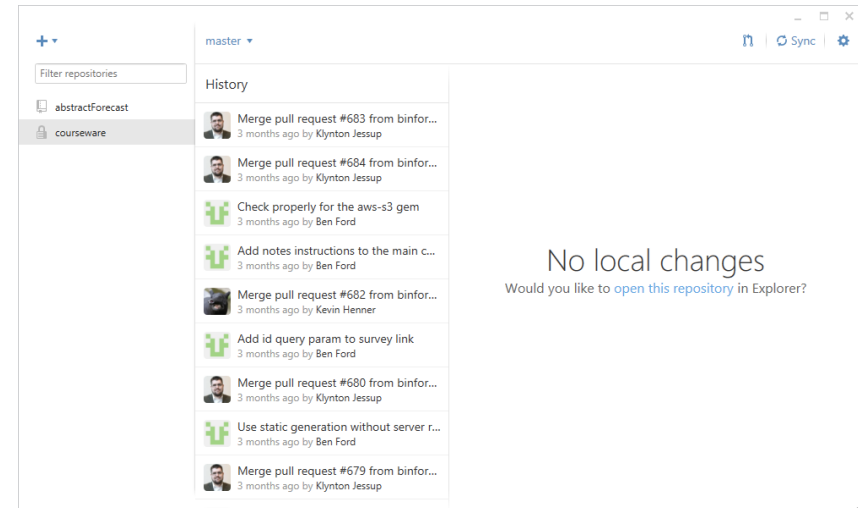
Coding

Geppetto



puppetlabs.github.io/geppetto/

GitHub



windows.github.com

Community



[www.puppetlabs.com/
community/PUG](http://www.puppetlabs.com/community/PUG)



www.puppetlabs.com/ptp

github.com/fnaard/cascadia-puppetcode

Gabriel M Schuyler
gabe@puppetlabs.com
@gabe_sky

