Test-Driven Infrastructure with Test Kitchen, Chef and DSC

Stuart Preston, Principal Engineer – Chef Software
Microsoft MVP (Azure)
stuart@chef.io @stuartpreston

Gael Colas
Microsoft MVP (CDM)
gaelcolas@synedgy.com @gaelcolas

Agenda

- Introduction
- What is Test-Driven Infrastructure?
- What tools did Chef create?
- Demos
- Recap and tools involved
- Questions/Feedback session

Deming Cycles

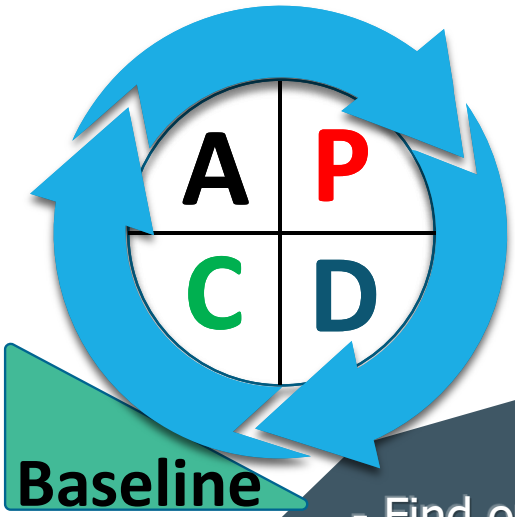
https://en.wikipedia.org/wiki/PDCA

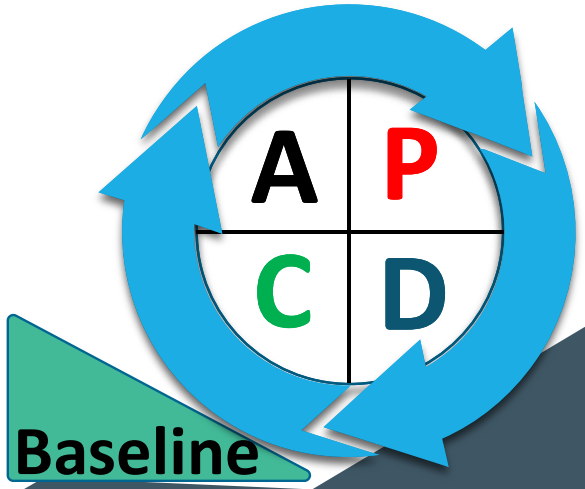
PLAN

DO

CHECK

ACT





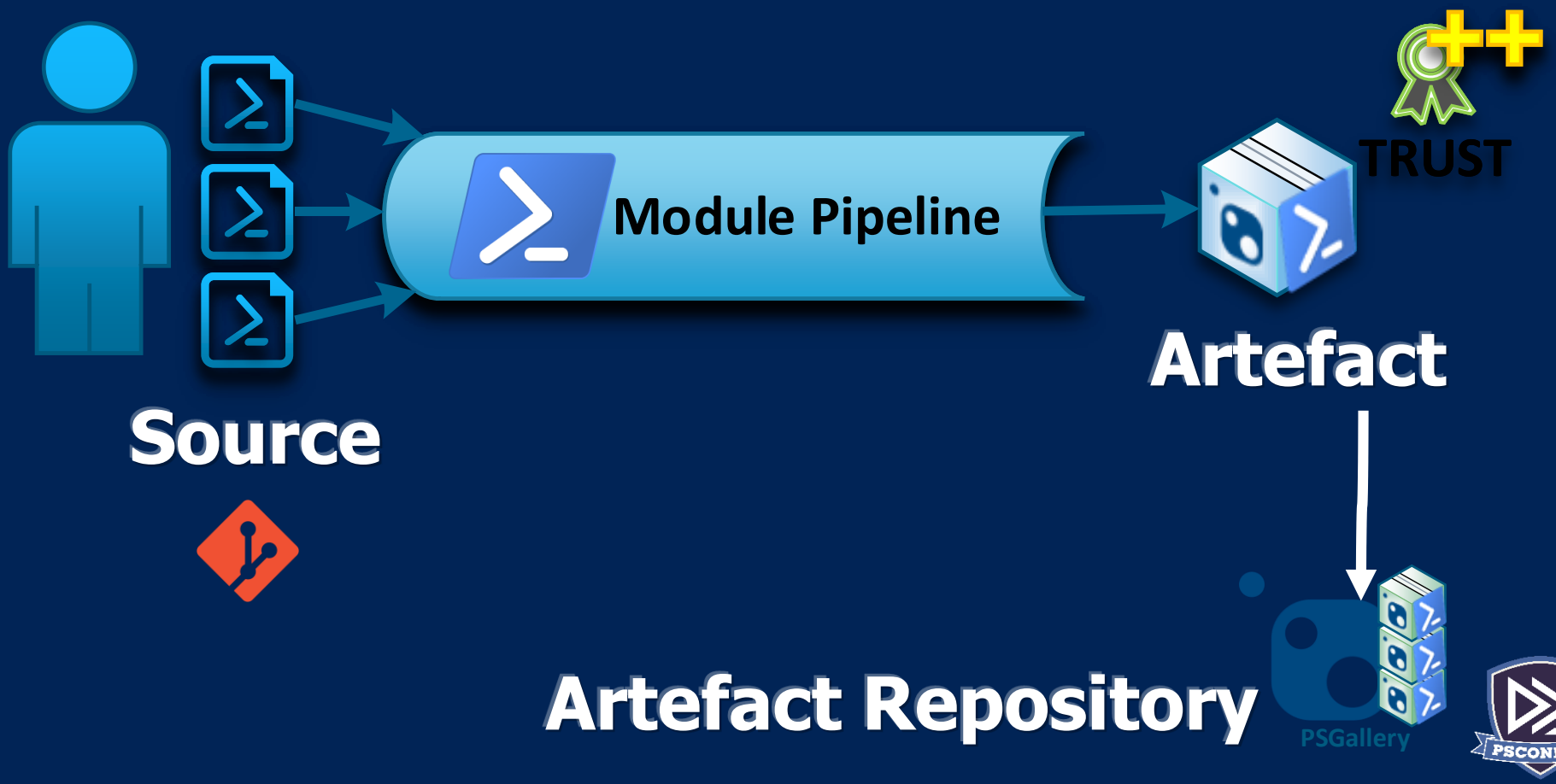
- Find out where you are
- Take a small step towards your goal
- Adjust your understanding based on what you learned
- Repeat

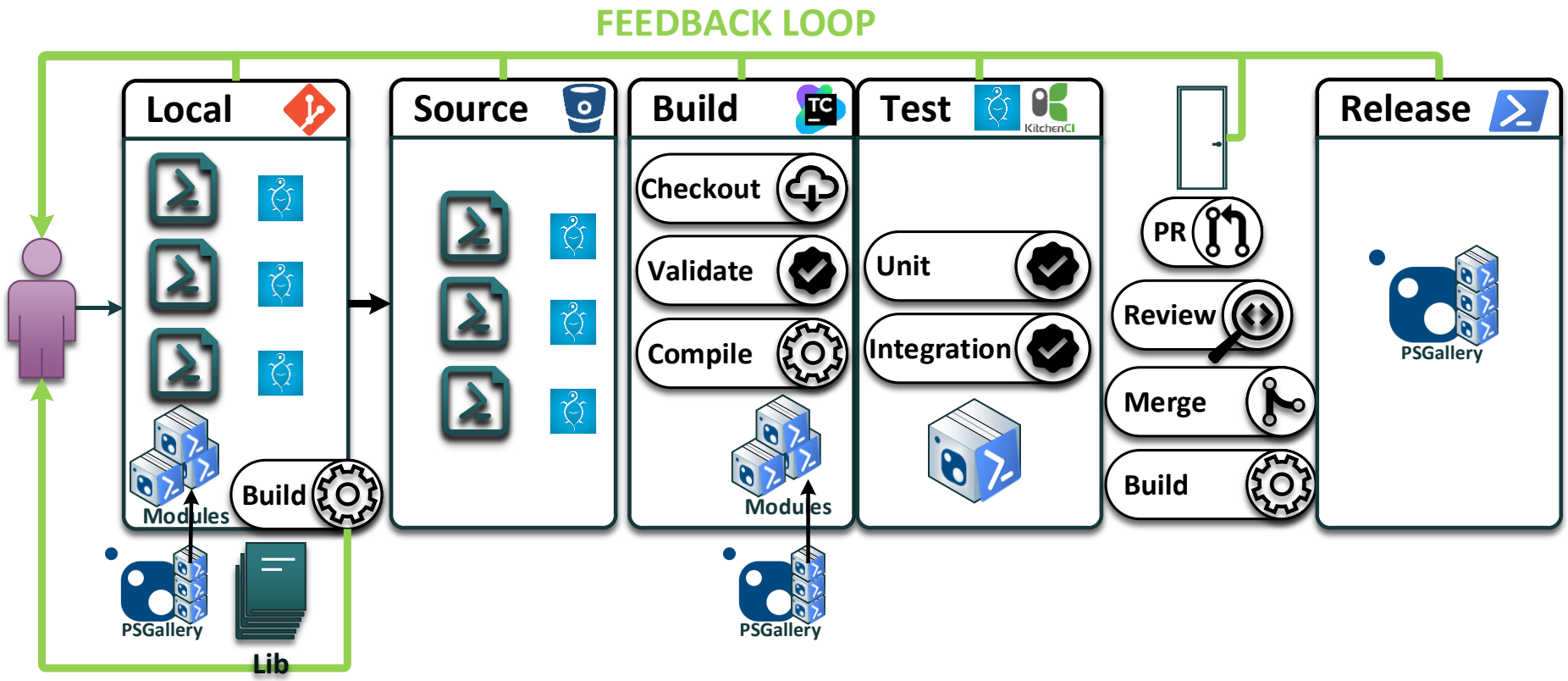
Dave Thomas

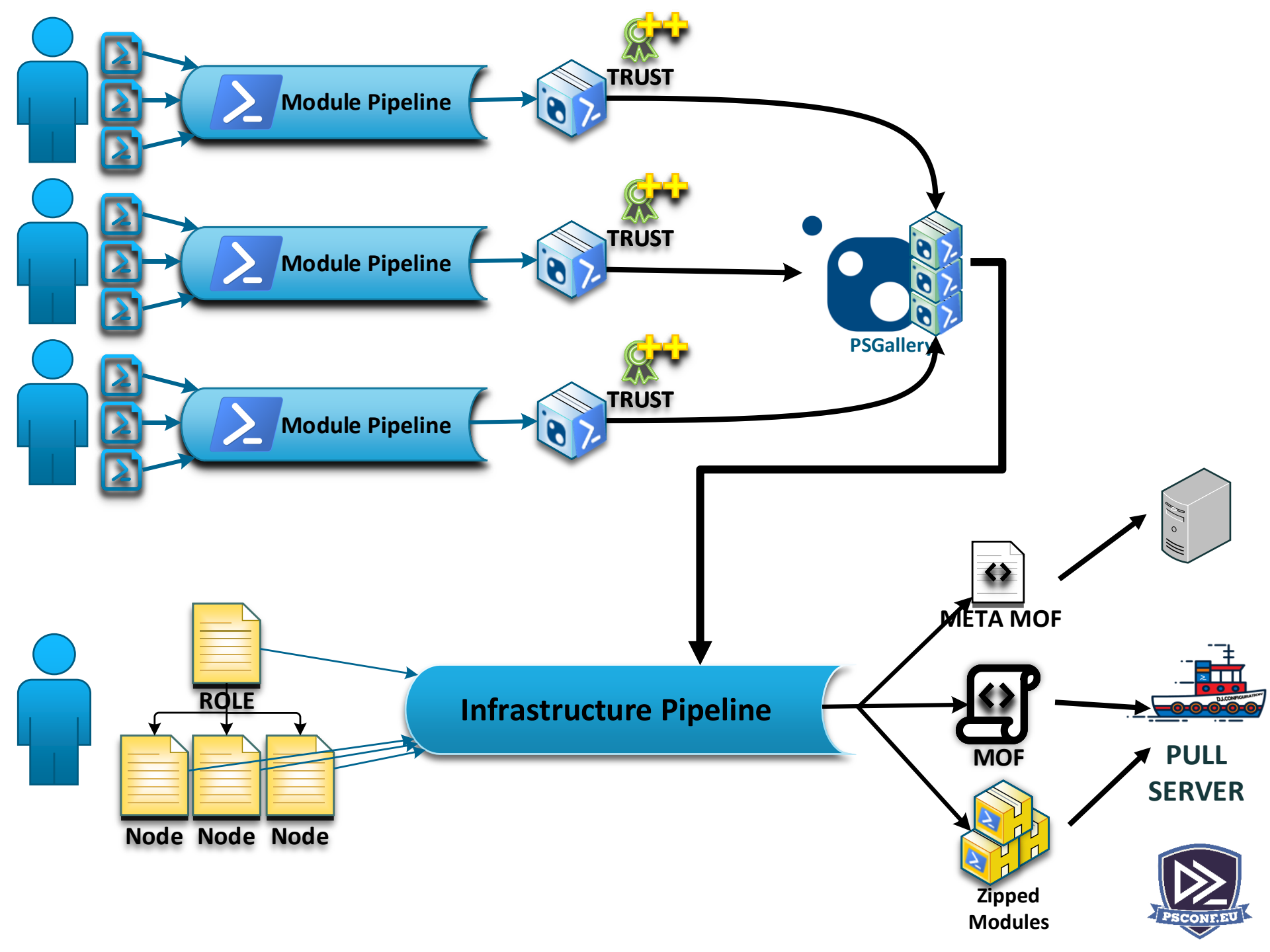
Module Pipeline - Overview

"The pipeline [...] applies its rules in a rigorous, unemotional way"

- Mark Schwartz, *The Art of Business Value*




FEEDBACK LOOP
Local
Source
Build
Test
KitchenCI
Release
Checkout
Validate
Compile
Unit
Integration
PR
Review
Merge
Build
Modules
Build
PSGallery
Lib
Modules
PSGallery
PSGallery


Module Pipeline
TRUST
Module Pipeline
TRUST
Module Pipeline
TRUST
PSGallery
ROLE
Node
Node
Node
Infrastructure Pipeline
META MOF
MOF
Zipped
Modules
PULL
SERVER
PSCONF.EU

What is Test-Driven Infrastructure?

- Software behaves differently on different platforms
  - Windows 7 vs Windows 10
  - Windows Server vs Linux
  - Debian vs CentOS/RedHat
- Test your application or payload across multiple platforms
- Need to test the desired state of the application/payload to check we have the correct result
- We express those tests in code and use tools to execute the tests

What did Chef do about this?

- Chef built **Test Kitchen** in 2014
  - Windows support added in 2015
- Focus on testing "cookbook" across platforms
- Open Source – https://github.com/test-kitchen
  - Contributions from Puppet and Ansible
- "Configuration as code" - .kitchen.yml
- "Clean room" environment every execution
- Tests are copied from the workstation to the target

Demo[0]

Code is available at:
https://github.com/stuartpreston/psconfeu2018

Recap

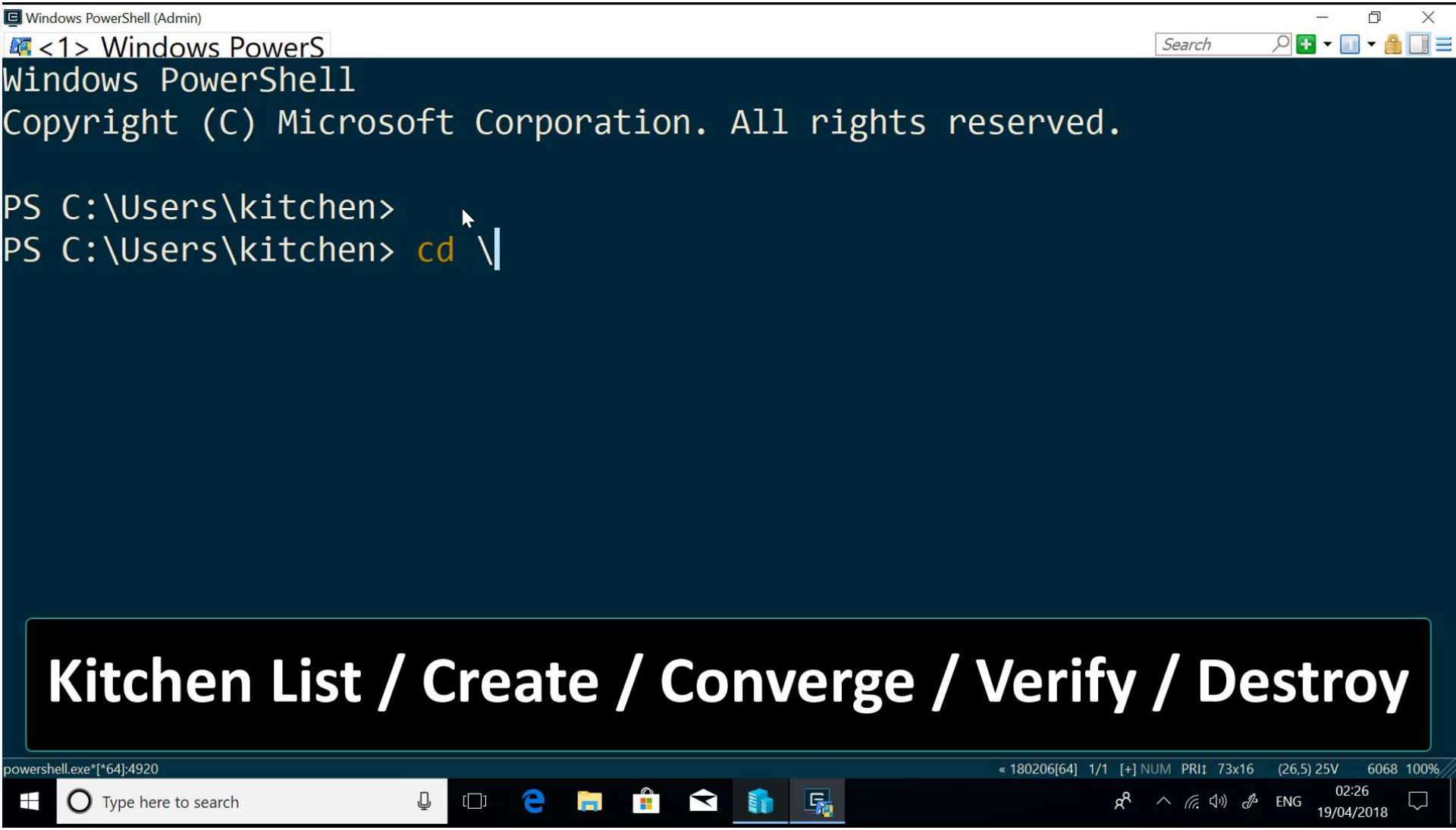
- Install Vagrant and Test Kitchen
  - https://vagrantup.com https://kitchen.ci
- Ensure Hyper-V is enabled
  - Use “Default Switch” on a workstation
- .kitchen.yml file drives configuration
  - Avoid temptation to test in parallel if possible
  - (makes debugging easier)
- Find Vagrant/Hyper-V “boxes”
  - https://app.vagrantup.com
- Check out other drivers (e.g. Azure)

Test Kitchen configuration

```
---
driver:
  name: vagrant

provisioner:
  name: chef_zero

platforms:
  - name: ubuntu-14.04
  - name: windows-2012r2

suites:
  - name: client
    run_list:
      - recipe[postgresql::client]
  - name: server
    run_list:
      - recipe[postgresql::server]
```

Demo [1]

Code is available at:
https://github.com/gaelcolas/kitchen_hyperV_dsc


Windows PowerShell (Admin)
<1> Windows PowerS
Search
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
PS C:\Users\kitchen>
PS C:\Users\kitchen> cd \
Kitchen List / Create / Converge / Verify / Destroy
powershell.exe*[*64]:4920
« 180206[64] 1/1 [+] NUM PRI↕ 73x16 (26,5) 25V 6068 100%
Type here to search
ENG
02:26
19/04/2018

Questions?

Further Reading

- https://gaelcolas.com/2016/07/11/introduction-to-kitchen-dsc/
- https://gaelcolas.com/2016/07/09/preparing-an-image-for-dsc-development/
- https://kitchen.ci
- https://app.vagrantup.com
- https://inspec.io
- https://docs.chef.io