5

Video operator, did you start the recording?

3

2

I

This Session

- Learn what is Chocolatey?

- Understand the Chocolatey recommended Organizational Architecture;

- Learn how to automate your package internalization process with Chocolatey, PowerShell and Jenkins;

- Based on a blog post https://blog.pauby.com/post/getting-started-with-chocolatey-and-jenkins/

PSCONF.EU

@pauby

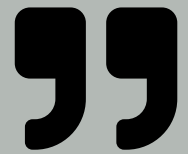@pauby

9

Before We Start

C4B

@pauby

What is Chocolatey?

A Definition Of Chocolatey

" Chocolatey is a package manager for Windows, like apt-get or yum but for Windows. It was designed to be a decentralized framework for quickly installing applications and tools that you need. It is built on the NuGet infrastructure currently using PowerShell as its focus for delivering packages from the distros to your door, err computer. "

@pauby

Chocolatey manages Packages

Packages manage Installers

Chocolatey Package Sources

Where do packages come from?

Chocolatey Sources

- Where packages come from;
- C4B comes with two Chocolatey sources by default:
  - **`chocolatey`** – Chocolatey Community Repository
  - **`chocolatey.licensed`** - Chocolatey Community Repository cached binaries;
- Add your own sources:
  - Repository manager: Artifactory, Nexus, ProGet
  - Local folder

@pauby

DEMO

Let's look at Chocolatey sources.

Internalizing Packages

Keeping it in the family.

Why Internalize Packages?

- What is 'package internalization'?

- Do I need Chocolatey 4 Business to use it?

- Organizations recommended to disable the default sources.
  - Reliability
  - Trust
  - Bandwidth
  - Copyright Restrictions

- Using the default `chocolatey` source is subject to:
  - rate limiting;
  - excessive download limiting;

Automated Internalization

- C4B automatically internalizes the vast majority of packages;
- Very fast;
- Don't reinvent the wheel;
- Automation!

DEMO

Package internalization.

Chocolatey Recommended Organizational Architecture
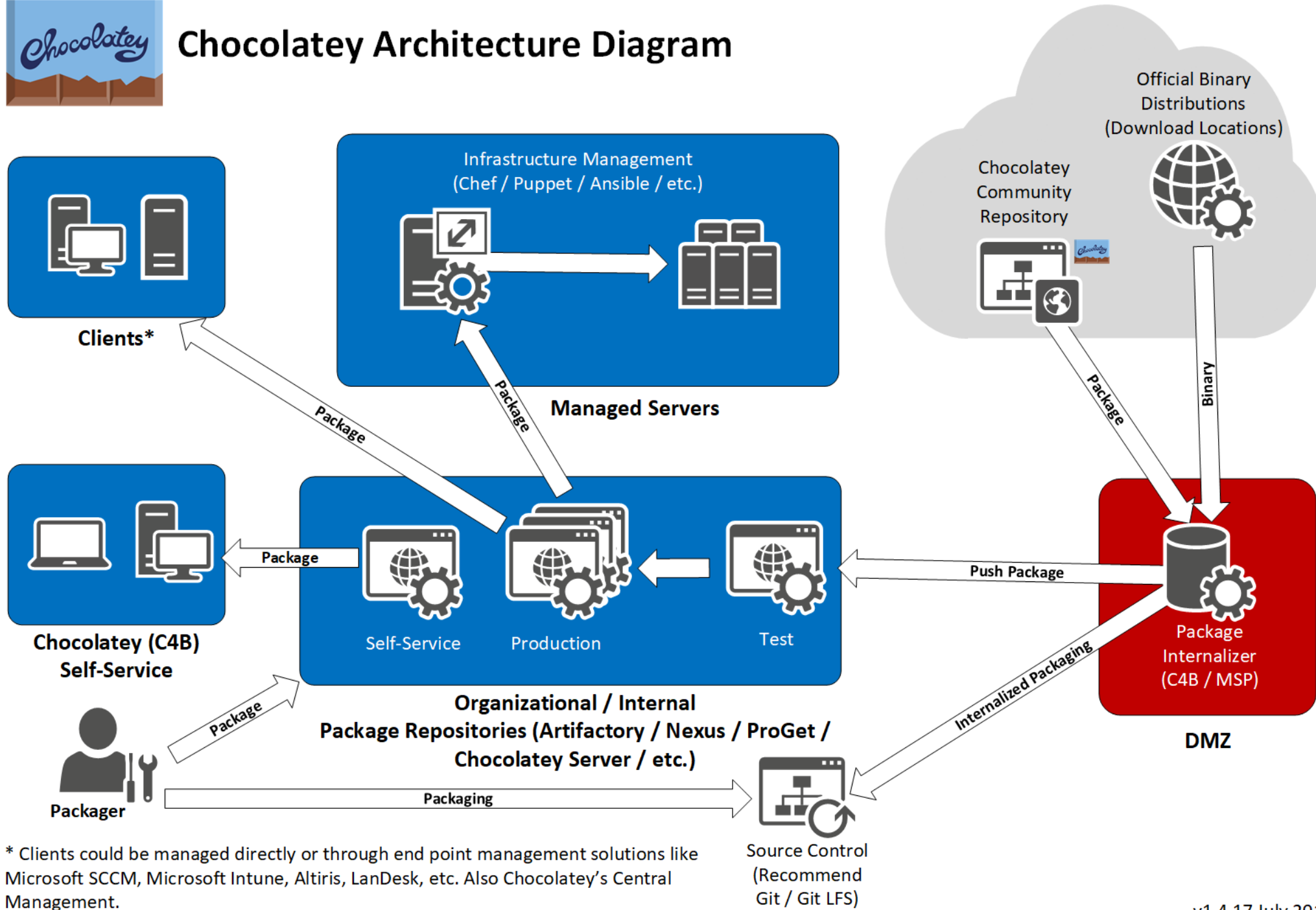
Every organisation is a snowflake.

(shield your eyes!)

@pauby

Chocolatey Architecture Diagram

Infrastructure Management
(Chef / Puppet / Ansible / etc.)

Managed Servers

Clients*

Official Binary
Distributions
(Download Locations)

Chocolatey
Community
Repository

Package

Binary

Package

Package

Self-Service

Production

Test

Push Package

Package Internalizer
(C4B / MSP)

DMZ

Chocolatey (C4B)
Self-Service

Package

Organizational / Internal
Package Repositories (Artifactory / Nexus / ProGet /
Chocolatey Server / etc.)

Internalized Packaging

Packager

Packaging
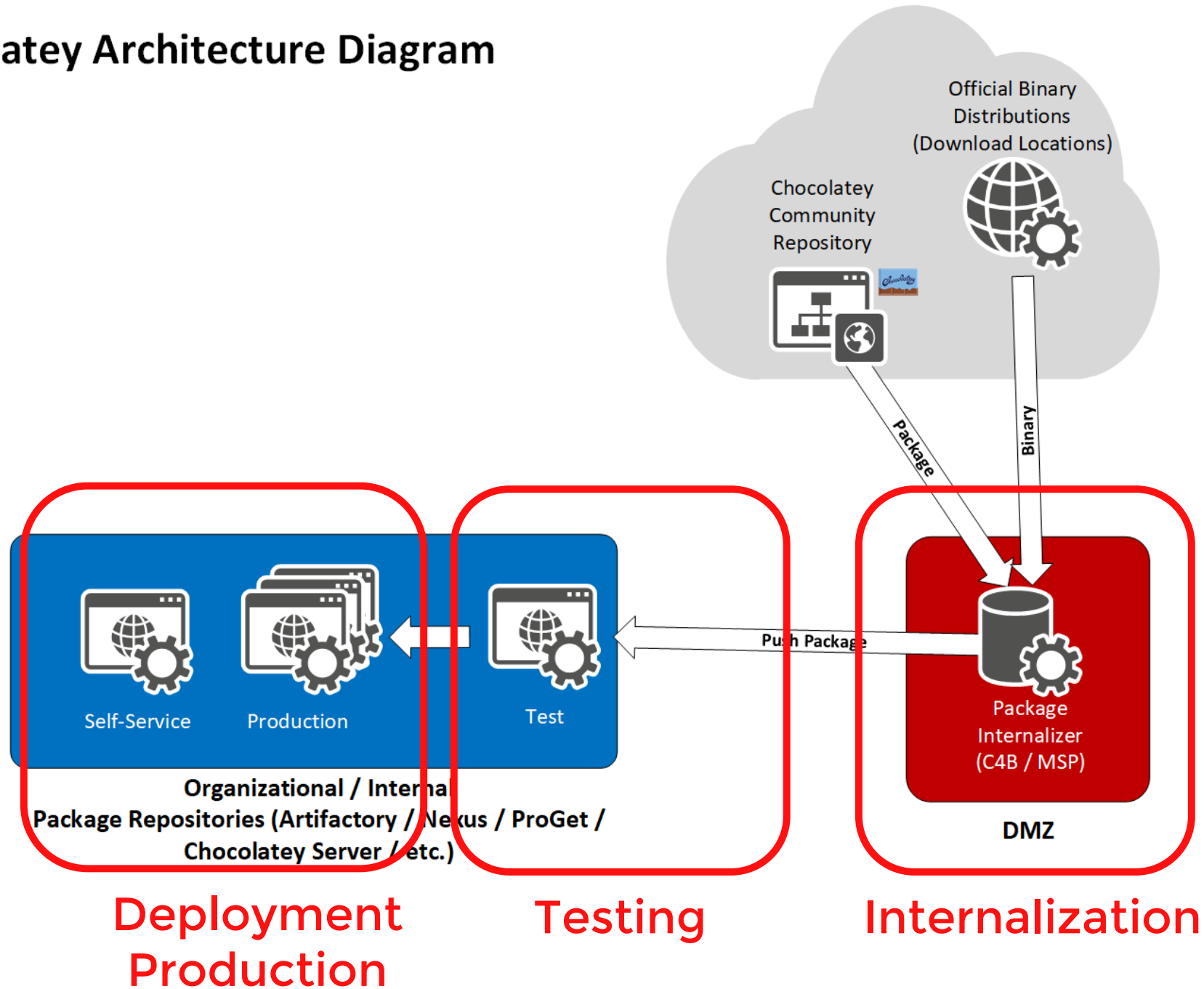
Source Control
(Recommend
Git / Git LFS)

* Clients could be managed directly or through end point management solutions like
Microsoft SCCM, Microsoft Intune, Altiris, LanDesk, etc. Also Chocolatey's Central
Management.

23

v1.4 17 July 2018

Chocolatey Architecture Diagram

Official Binary Distributions (Download Locations)

Chocolatey Community Repository

Package

Binary

Self-Service   Production   Test

Push Package

Package Internalizer (C4B / MSP)

DMZ

Organizational / Internal Package Repositories (Artifactory / Nexus / ProGet / Chocolatey Server / etc.)

Deployment Production

Testing

Internalization

v1.4 17 July 2018

Tools We Need

Package Repositories - Chocolatey.Server
Recommended: Artifactory, Nexus, ProGet

Code - PowerShell

Automation - Jenkins

```powershell
# Jenkins - Internalize Package and Push to Test Repo

$temp = Join-Path -Path $env:TEMP -ChildPath ([GUID]::NewGuid()).Guid
New-Item -Path $temp -ItemType Directory | Out-Null
Write-Output "Created temporary directory '$temp'."


Set-Location (Join-Path -Path $env:SystemDrive -ChildPath 'scripts')
.\Internalize-ChocoPackage.ps1 -Name $env:P_PKG_LIST `
    -OutputDirectory $temp -Verbose


if ($LASTEXITCODE -eq 0) {
    Set-Location $temp
    Get-ChildItem -Path *.nupkg | ForEach-Object {
        choco push $_.Name --source=$env:G_TEST_REPO `
            --apikey=$env:G_API_KEY --limitoutput
    }
}
```

```powershell
# Jenkins - Test Package and Push to Prod Repo

Set-Location (Join-Path -Path $env:SystemDrive -ChildPath
'scripts')
.\Update-ProdRepoFromTest.ps1 `
    -ProdRepo $env:G_PROD_REPO `
    -ProdRepoApiKey $env:G_API_KEY `
    -TestRepo $env:G_TEST_REPO `
    -Verbose
```
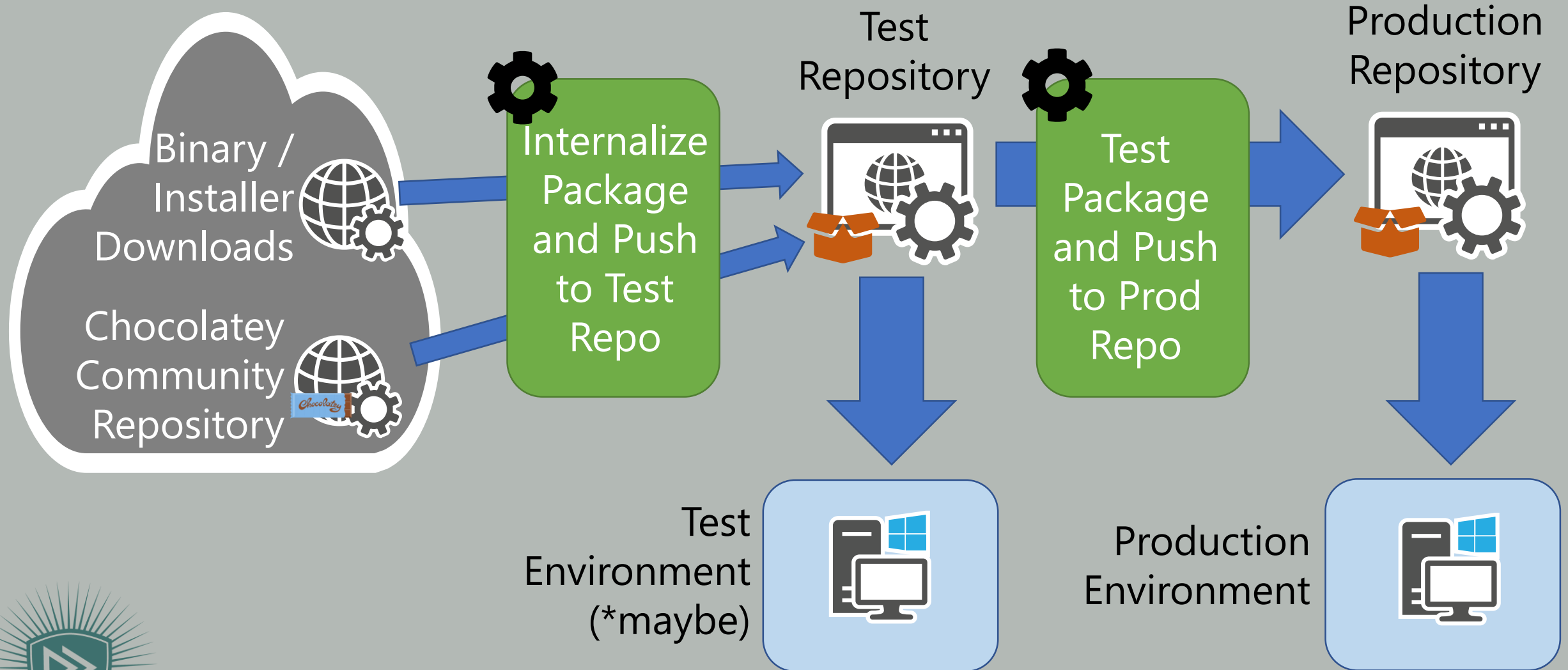
```powershell
# Jenkins - Upgrade Outdated Packages in Test Repository

Set-Location (Join-Path -Path $env:SystemDrive -ChildPath 'scripts')
.\Get-UpdatedPackage.ps1 -LocalRepo $env:P_LOCAL_REPO_URL `
    -LocalRepoApiKey $env:P_LOCAL_REPO_API_KEY `
    -RemoteRepo $env:P_REMOTE_REPO_URL `
    -Verbose
```

Jenkins Configuration

Binary / Installer Downloads

Chocolatey Community Repository

Internalize Package and Push to Test Repo

Test Repository

Test Package and Push to Prod Repo

Production Repository

Test Environment (*maybe)

Production Environment

@pauby

Ideas to Extend and Customise

- Schedule Updating Packages from Chocolatey Community Repository;

- Write your own Pester Tests that your organization needs;

- When packages are published to production send a message via email / Slack / Teams or whatever you use;

Summary

- What Chocolatey is and how sources work;

- Understand package internalization and why it's important for organizations;

- How to apply automation to Chocolatey package testing and deployment;

Slides and demo code

```
Start-Process -FilePath https://github.com/psconfeu/2019
```

@pauby

Questions?

Use the conference app to vote for this session:
https://my.eventraft.com/psconfeu

@pauby

about_Speaker

🏠 pauby.com

🐦 @pauby

(github) github.com/pauby

(in) linkedin.pauby.com

Paul Broadwith

pau.by/talks

🐦 @pauby