EBRU CUCEN

POWERSHELL MODULE LIFECYCLE

ABOUT ME

▸ EBRU CUCEN

▸ @ebrucucen

▸ https://github.com/ebrucucen

▸ Co-ordinator for Get-PSUGUK

▸ Associate Trainer for DevOps Tools

▸ .NET dev\trainer turned into software lifecycle inhabitant

AGENDA

▸ Script Modules

▸ Lifecycle of Modules

SCRIPT MODULE

▸ Control

  ▸ public/private functions/variables

▸ Group

  ▸ reuse/abstraction

▸ Describe

  ▸ versions, dependencies

SCRIPT MODULE – [CONTROL]

```powershell
#Get public and private function definition files.

    $Public  = @( Get-ChildItem -Path $PSScriptRoot\Public\*.ps1 -ErrorAction SilentlyContinue )

    $Private = @( Get-ChildItem -Path $PSScriptRoot\Private\*.ps1 -ErrorAction SilentlyContinue )

#Dot source the files

    Foreach($import in @($Public + $Private))
    {
        Try
        {
            . $import.fullname
        }
        Catch
        {
            Write-Error -Message "Failed to import function $($import.fullname): $_"
        }
    }

Export-ModuleMember -Function $Public.Basename
```

MODULE MANAGEMENT

- New-Module

- New-ModuleManifest

- Import-Module

- Get-Module

- Export-ModuleMember

- Remove-Module

- Test-ModuleManifest

SCRIPT MODULE [GROUP]

▸ appveyor.yml

▸ deploy.PSDeploy.ps1

▸ main.build.ps1

▸ README.md

▸ PSEventLogEntry\

    ▸ en-US\

    ▸ Public\

    ▸ Private\

    ▸ Test\

    ▸ PSEventLogEntry.psd1

    ▸ PSEventLogEntry.psm1

SCRIPT MODULE [DESCRIBE]

New-ModuleManifest -path .\sample.psd1
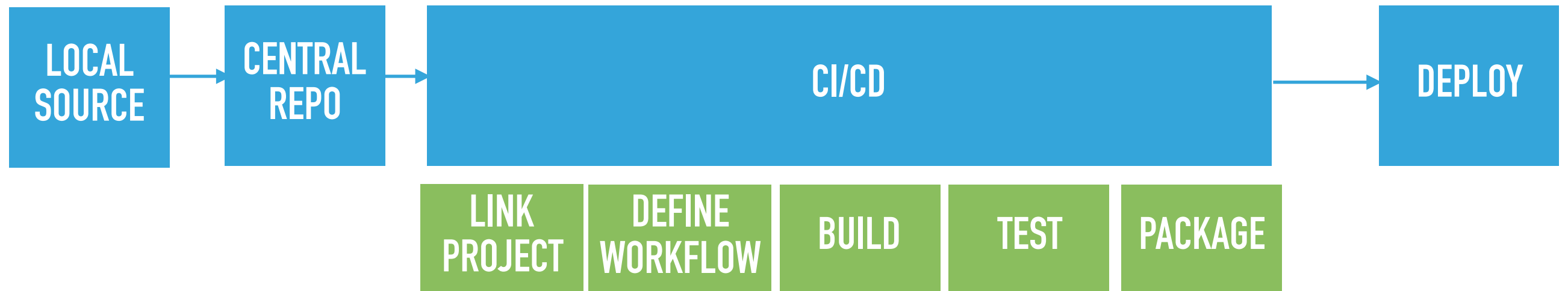
https://github.com/ebrucucen/PSModules/blob/master/sample.psd1

https://github.com/ebrucucen/PSModules/blob/master/PSEventLogEntry/PSEventLogEntry.psd1

WHAT IS THE BIG PICTURE?
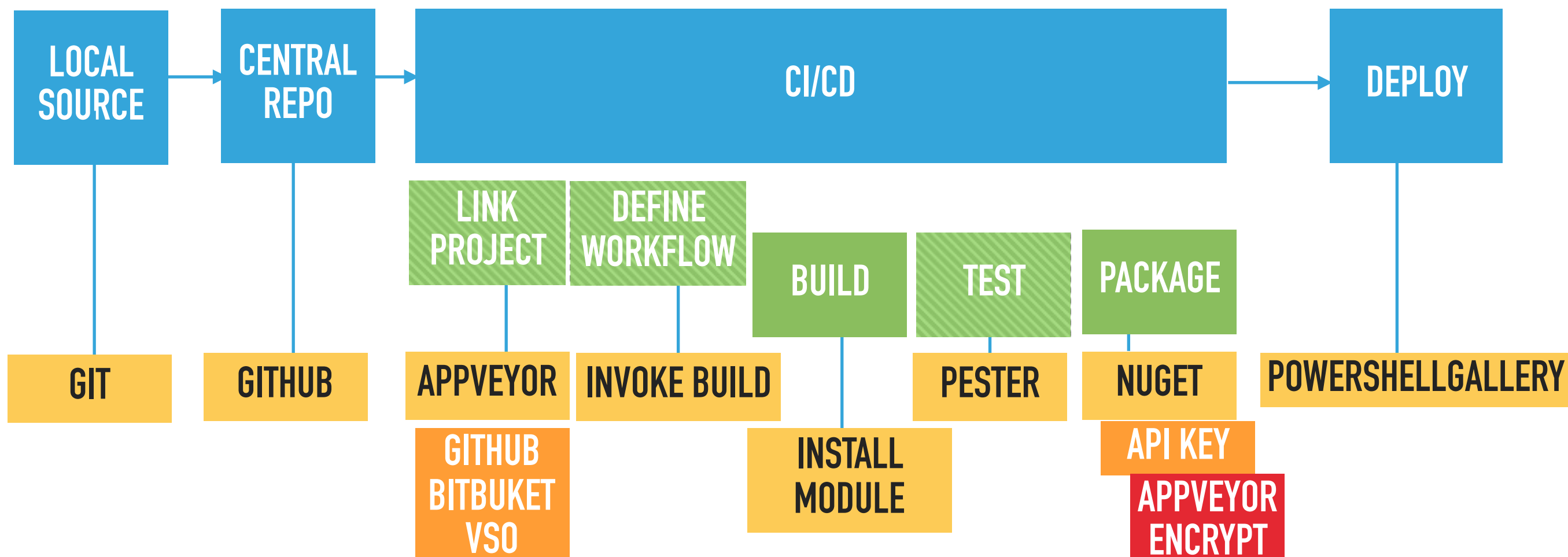
POWERSHELL LIFECYCLE

| LOCAL SOURCE | → | CENTRAL REPO | → | CI/CD | → | DEPLOY |

CI/CD:
| LINK PROJECT | DEFINE WORKFLOW | BUILD | TEST | PACKAGE |

SELECTED TOOLSET FOR POWERSHELL

LINK PROJECT

▸ Step1: Sign in:

   ▸ https://ci.appveyor.com/login

▸ Step2: Login with with developer/AppVeyor account

LINK PROJECT

▸ Step3: Select the project to link

LINK PROJECT

▸ Step4: Add appveyor.yml

```
appveyor.yml  ×

1
2    # This is appveyor yml configuration to tell
3    # what to do for each logical build/deployment phase:
4    # all the options are here: https://www.appveyor.com/docs/appveyor-yml/
5
6    # we want disable the MSBuild
7    build: false
8
9    #Kick off the CI/CD pipeline
10   before_build:
11    - echo "started"
12    - ps: install-module invokebuild
13    - echo "installed"
14    - ps: import-module invokebuild
15    - echo "imported"
16
17   #define the build script instead of automated MSBuild
18   build_script:
19      - ps: invoke-build
20
```

https://github.com/ebrucucen/PSModules/blob/master/appveyor.yml

APPVEYOR REFERENCE*

GENERAL	ENVIRONMENT	BUILD	TEST	PACKAGE	DEPLOY	NOTIFICATION	GLOBAL HANDLERS
skip_commits:	init:	before_build:	before_test:	artifacts:	before_deploy:	provider:	on_success:
	image:	build:	test:		deploy:	on_build_success:	on_failure:
	hosts:	build_script:	test_script:		deploy_script:	on_build_failure:	on_finish:
	install:	after_build:	after_test:		after_deploy:	on_build_status_changed:	

- message:
- author:

- git config

no need MSBuild

- build: off

we'll run Pester

- test: off

- path:

we'll run Invoke-Build script

- build:
  - ps:

alternative to PSDeploy

- provider:
- artifact:

providers:
Email, HipChat, Slack,
Campfire, Webhook

providers:
Local, FTP, WebDeploy, AzureCS,
AzureBlob, S3, NuGet, Environment

- deploy: off/on

*These are options relevant for a PowerShell lifecycle

DEFINE WORKFLOW

▸ Invoke-Build calls

   ▸ *.build.ps1

    [from the same level]

▸ a template on the right hand side:

▸ the last line calls the tasks in order:

   task . Init, Build, Test, Clean, Deploy

```
# Define properties such as:
# Test/Output Directories ...

task Init {
    # Initial tasks such as:
    # Install/Import packages ...
}
task Build -Depend Init{
    #Build tasks such as:
    #Generate the resources you want ...
}
task Test{
    #Test tasks such as:
    #Run Invoke-Pester ...
}
task Deploy{
    #Deploy tasks such as:
    #Run PSDeploy.ps1
}
task Clean {
    #Clean tasks such as:
    #Remove test files
}

#Define the order you want to call:
task .  Init, Build, Test, Clean, Deploy
```

TEST EXAMPLE

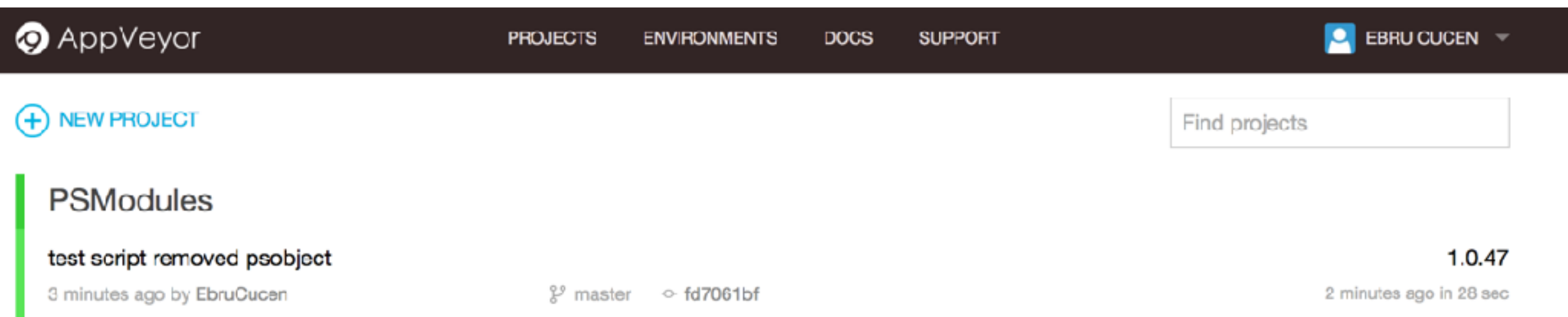▶ main.Build.ps1 Test task :

```powershell
#Test Task
task Test{
    $lines
    'TDD: Tests first! '
    #Get the test files:
    $TestFiles= Get-ChildItem -Path $TestLocation -Filter "*.Tests.*"
    foreach ($testFile in $testFiles){
        $testOutputFileName= Join-path -path $testfile.Directory -ChildPath "$($testfile.basename)_$timestamp.xml"
        $testResult=Invoke-Pester -Script $testFile.Fullname  -OutputFile $testOutputFileName -OutputFormat NUnitXml
    }
    #upload to Appveyor
    (New-Object 'System.Net.WebClient').UploadFile("https://ci.appveyor.com/api/testresults/nunit/$($env:APPVEYOR_JOB_ID)", `
            (Resolve-Path $testOutputFileName))
    if($testResult.FailedCount -gt 0)
    {
        Write-Error "Failed '$($testResult.FailedCount)' tests, build failed"
    }
    $lines
}
```

ENJOY AUTOMATED CI OUTPUT

▸ README.md on Github site can display the status:

▸ Appveyor website will have all build history, logs …

BUILD OUTPUT [1: APPVEYOR.YML CALLED]

BUILD OUTPUT 2: [INIT TASK IN MAIN.BUILD.PS1 CALLED]

```
14 Task /./Init
15 ------------------------------------------------------------------
16 Build System Details:
17
18
19 Name   : BHBranchName
20 Value : master
21
22 Name   : BHProjectName
23 Value : PSEventLogEntry
24
25 Name   : BHPSModuleManifest
26 Value : C:\projects\psmodules\PSEventLogEntry\PSEventLogEntry.psd1
27
28 Name   : BHBuildNumber
29 Value : 49
30
31 Name   : BHPSModulePath
32 Value : C:\projects\psmodules\PSEventLogEntry
33
34 Name   : BHBuildSystem
35 Value : AppVeyor
36
37 Name   : BHProjectPath
38 Value : C:\projects\psmodules
39
40 Name   : BHCommitMessage
41 Value : extra tidy up on init task
42
43
44
45
46
47 ------------------------------------------------------------------
48 Done /./Init 00:00:09.6815866
```

```
49 Task /./PreTest
50 ------------------------------------------------------------------------
51
52
53     Directory: C:\projects\psmodules\PSEventlogEntry
54
55
56 Mode                LastWriteTime         Length Name
57 ----                -------------         ------ ----
58 -a----         3/24/2017   8:25 PM          595 PSEventlogEntry.psm1
59 Sucess import
60 ------------------------------------------------------------------------
61 Done /./PreTest 00:00:00.0781290
```

```
62 Task /./Test
63 ------------------------------------------------------------------------
64 TDD: Tests first!
65 Executing all tests in C:\projects\psmodules\PSEventLogEntry\Test\PSEventLogEntry.Tests.ps1
66
67 Executing script C:\projects\psmodules\PSEventLogEntry\Test\PSEventLogEntry.Tests.ps1
68
69   Describing Get-EventLogEntry Function Basic Tests
70
71     Context Invalid Parameter Set
72       [+] Should Throw Error for invalid ServerName 3.35s
73       [+] Should Throw Error for invalid LogName 145ms
74
75     Context Valid Parameter Set
76       [+] Should Not Throw An Error for the Valid Logname, Start/End Time inputs 189ms
77 Tests completed in 3.69s
78 Tests Passed: 3, Failed: 0, Skipped: 0, Pending: 0, Inconclusive: 0
79 ------------------------------------------------------------------------
80 Done /./Test 00:00:04.7536498
```

```
81 Done /. 00:00:14.5446171
82 Build succeeded. 4 tasks, 0 errors, 0 warnings 00:00:14.6540113
83
84
85 Discovering tests...OK
86 Build success
```

REFERENCES

My examples:

https://github.com/ebrucucen/PSModules

https://ci.appveyor.com/project/ebrucucen/psmodules

How many badges?

https://github.com/lvermeulen/Nanophone/blob/master/build/build.ps1

Appveyor:

https://www.appveyor.com/docs/appveyor-yml/

https://ci.appveyor.com/tools/encrypt

https://github.com/PowerShell/DscResource.Tests/blob/dev/AppVeyor.psm1

Devops:

https://xebialabs.com/periodic-table-of-devops-tools