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

TEXT

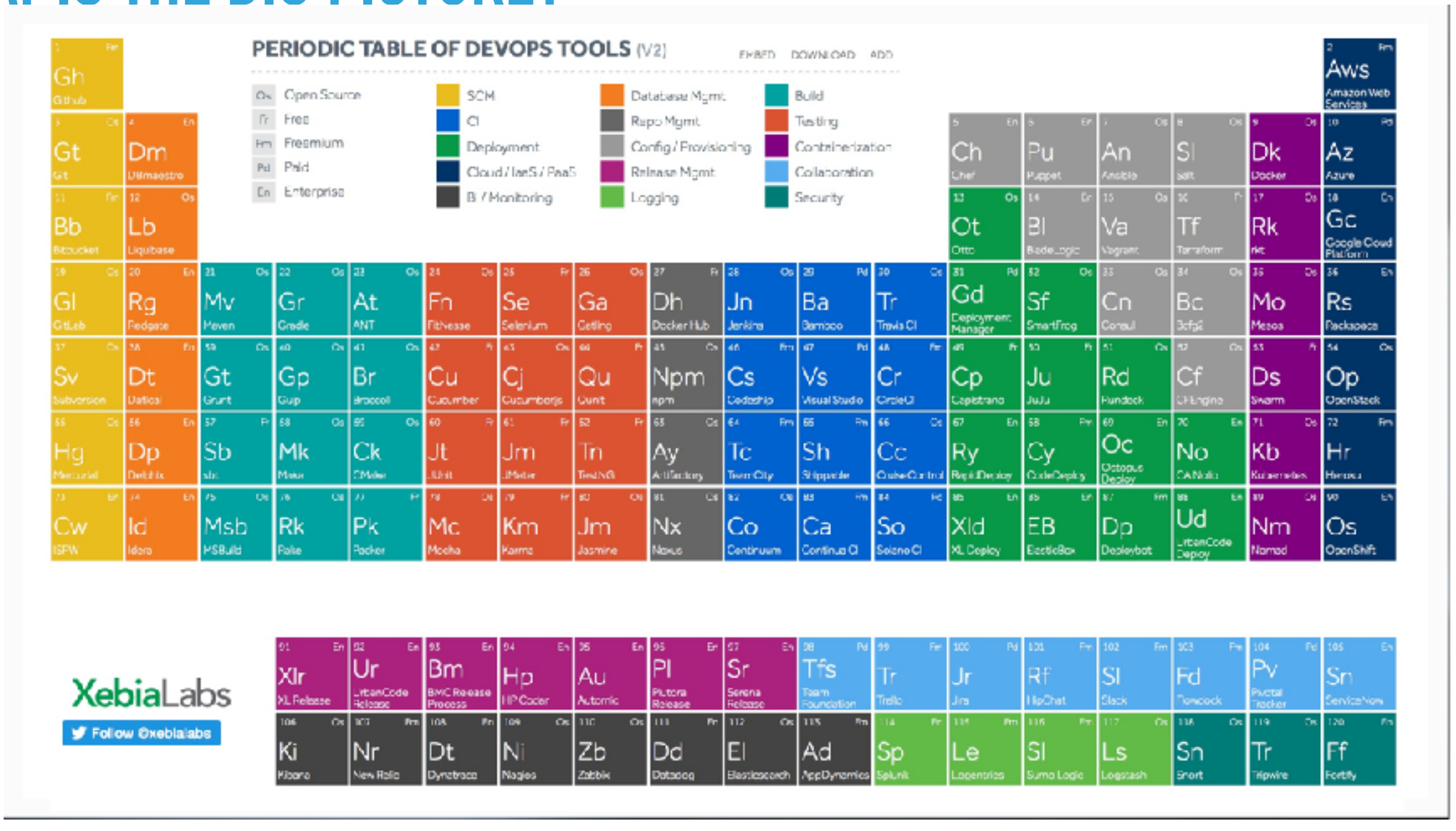
SCRIPT MODULE [DESCRIBE]

```
New-ModuleManifest -path .\sample.psd1
```

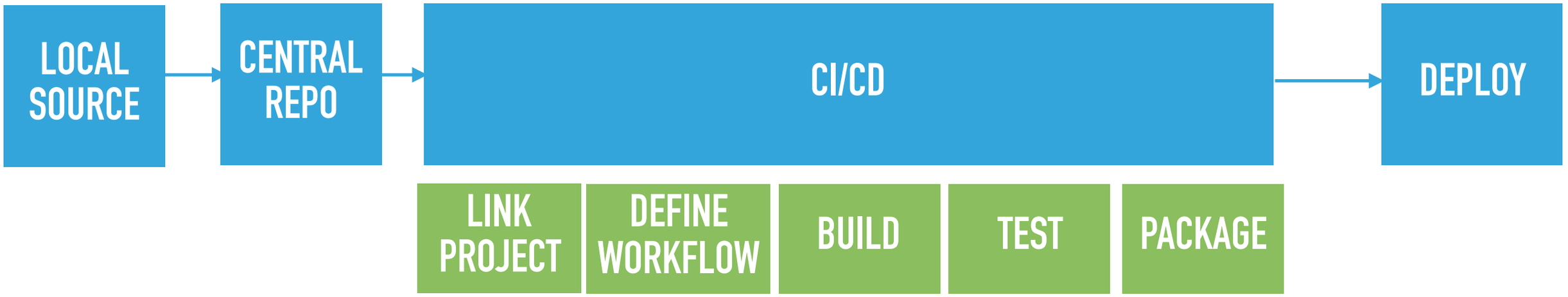
https://github.com/ebrucucen/PSModules/blob/master/sample.psd1

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

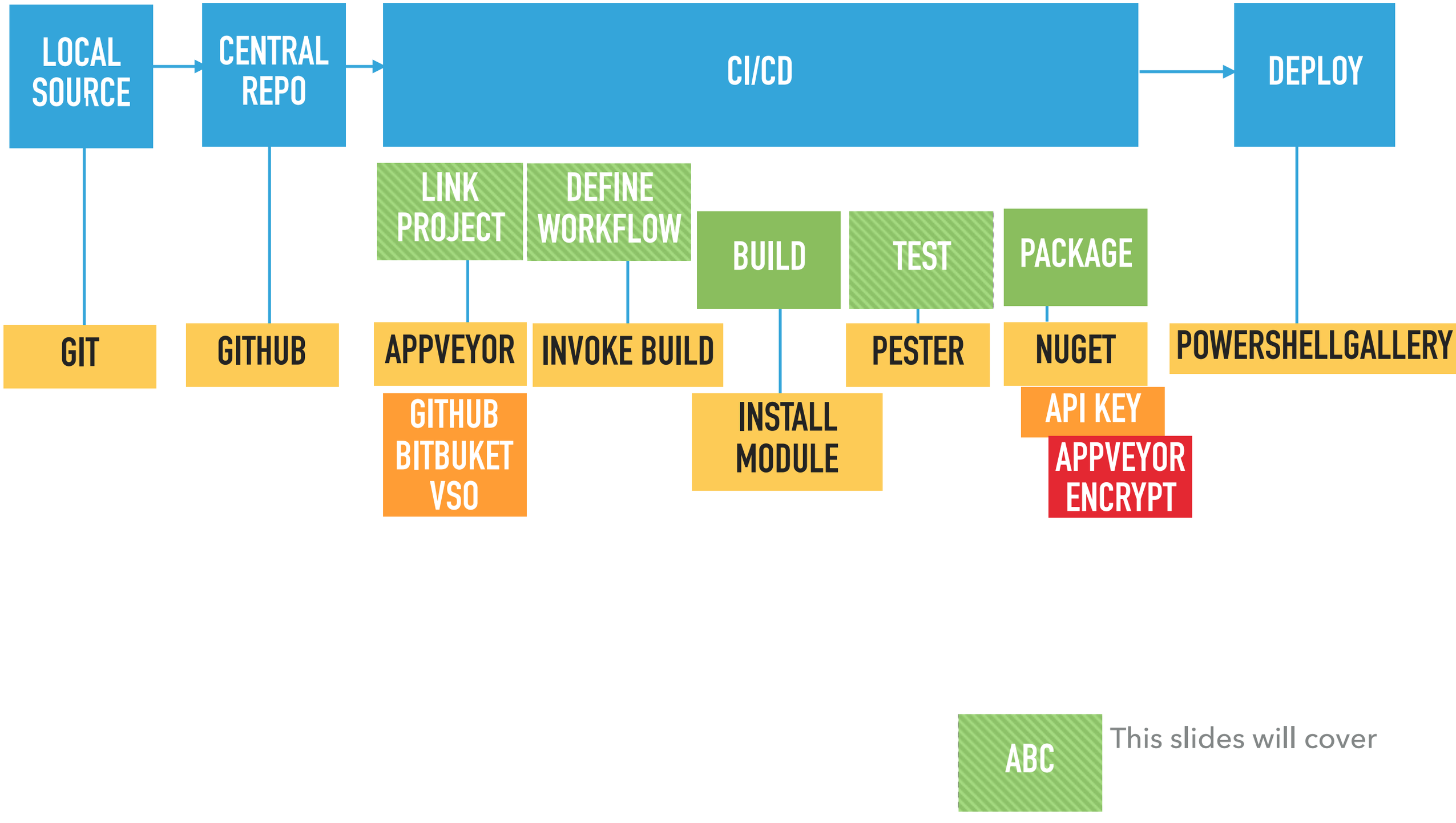
WHAT IS THE BIG PICTURE?



POWERSHELL LIFECYCLE

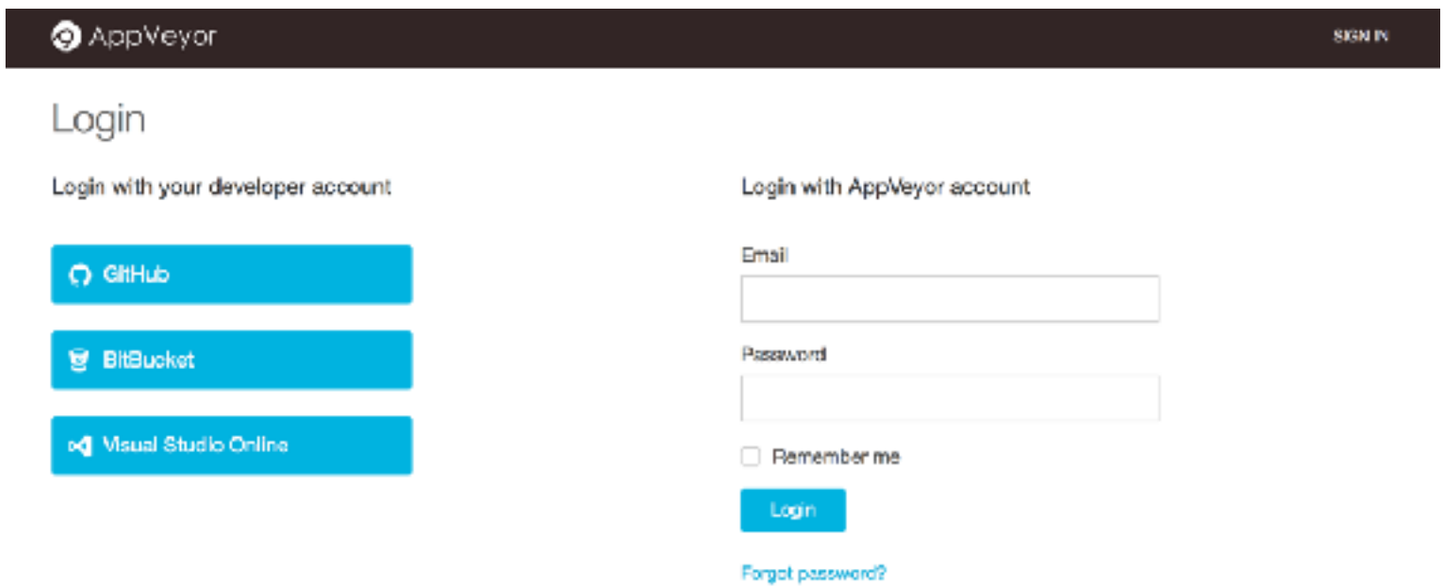


SELECTED TOOLSET FOR POWERSHELL



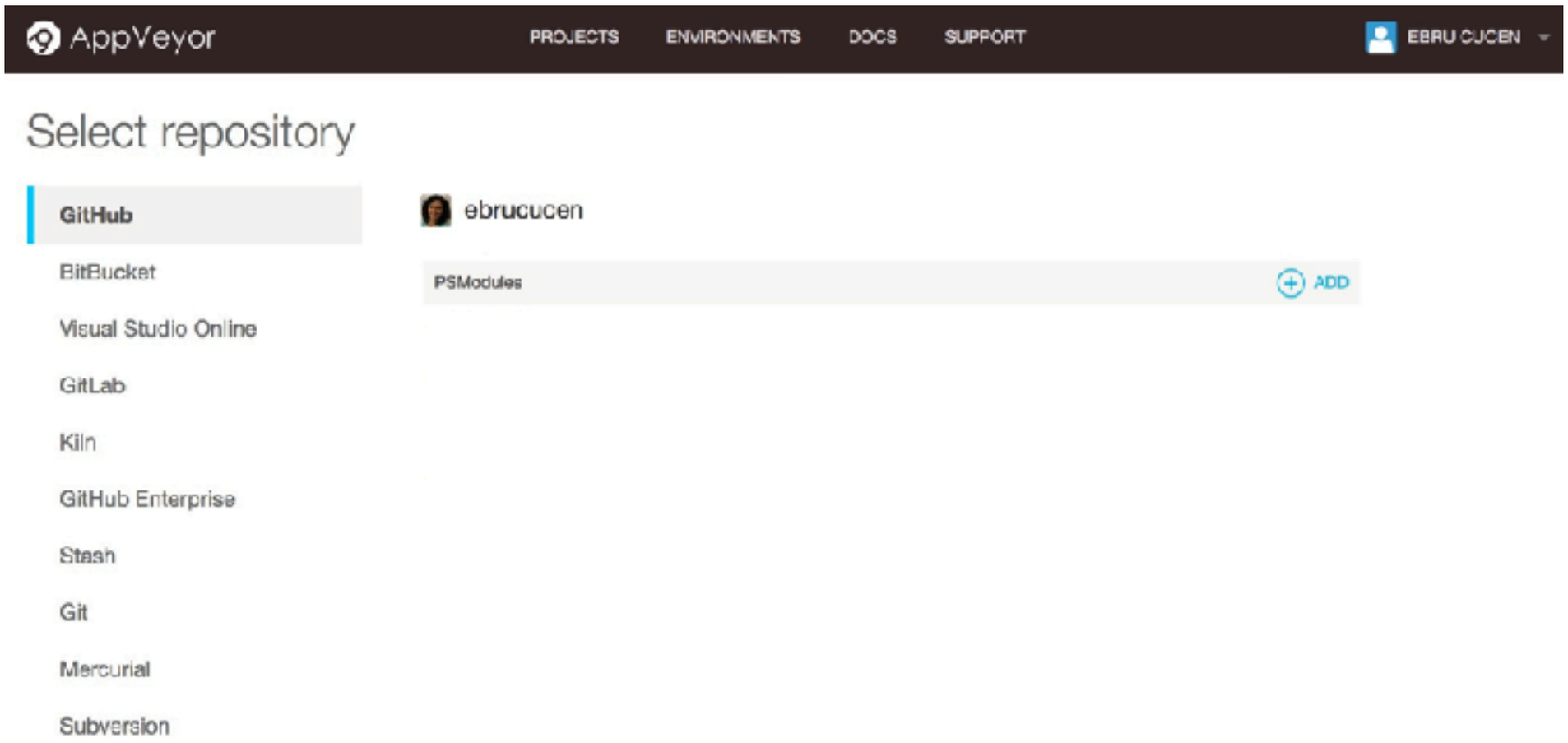
LINK PROJECT

- 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

```yaml

# This is appveyor yml configuration to tell
# what to do for each logical build/deployment phase:
# all the options are here: https://www.appveyor.com/docs/appveyor-yml/

# we want disable the MSBuild
build: false

#Kick off the CI/CD pipeline
before_build:
 - echo "started"
 - ps: install-module invokebuild
 - echo "installed"
 - ps: import-module invokebuild
 - echo "imported"

#define the build script instead of automated MSBuild
build_script:
  - ps: invoke-build

```

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

APPVEYOR REFERENCE*

GENERAL

skip_commits:

- message:
- author:

- git config

ENVIRONMENT

init:

image:

hosts:

install:

no need MSBuild

BUILD

before_build:

build:

build_script:

after_build:

- build: off

we'll run Invoke-Build script

- build:
  - ps:

TEST

before_test:

test:

test_script:

after_test:

we'll run Pester

- test: off

PACKAGE

artifacts:

- path:

DEPLOY

before_deploy:

deploy:

deploy_script:

after_deploy:

alternative to PSDeploy

- provider:
- artifact:

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

- deploy: off/on

NOTIFICATION

provider:

on_build_success:

on_build_failure:

on_build_status_changed:

providers:
Email, HipChat, Slack,
Campfire, Webhook

GLOBAL HANDLERS

on_success:

on_failure:

on_finish:

*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

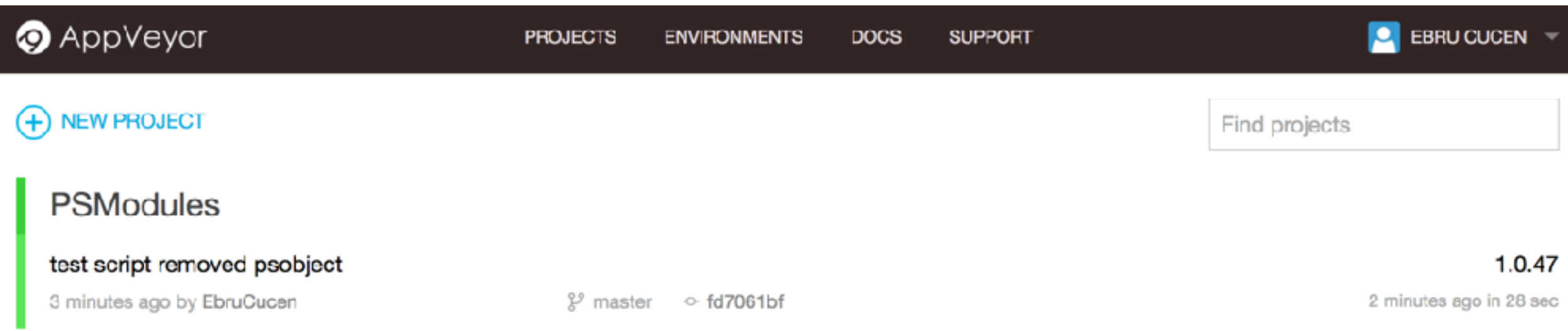
- main.Build.ps1 Test task :

```
#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]

AppVeyor PROJECTS ENVIRONMENTS DOCS SUPPORT EBRU CUCEN

PSModules

LATEST BUILD HISTORY DEPLOYMENTS SETTINGS NEW BUILD RE-BUILD COMMIT DEPLOY LOG

extra tidy up on init task 1.0.49

a few seconds ago by EbruCucen master 5821101b in a few seconds in 28 sec

CONSOLE MESSAGES TESTS 3 ARTIFACTS

```
1 Build started
2 git clone -q --branch=master https://github.com/ebrucucen/PSModules.git C:\projects\psmodules
3 git checkout -qf 5821101b0fef6c430c265166072e4afd8c881f6e
4 echo "started"
5 "started"
6 install-module invokebuild
7 echo "installed"
8 "installed"
9 import-module invokebuild
10 echo "imported"
11 "imported"
12 invoke-build
13 Build . C:\projects\psmodules\main.build.ps1
```

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

AppVeyor PROJECTS ENVIRONMENTS DOCS SUPPORT EBRU CUCEN

```
Task /./Init
----------------------------------------------------------------------
Build System Details:


Name  : BHBranchName
Value : master

Name  : BHProjectName
Value : PSEventLogEntry

Name  : BHPSModuleManifest
Value : C:\projects\psmodules\PSEventLogEntry\PSEventLogEntry.psd1

Name  : BHBuildNumber
Value : 49

Name  : BHPSModulePath
Value : C:\projects\psmodules\PSEventLogEntry

Name  : BHBuildSystem
Value : AppVeyor

Name  : BHProjectPath
Value : C:\projects\psmodules

Name  : BHCommitMessage
Value : extra tidy up on init task




----------------------------------------------------------------------
Done /./Init 00:00:09.6815866
```

BUILD OUTPUT 3: [PRETEST, TEST]

```
Task /./PreTest
----------------------------------------------------------------------


    Directory: C:\projects\psmodules\PSEventlogEntry


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----        3/24/2017   8:25 PM            595 PSEventlogEntry.psm1
Sucess import
----------------------------------------------------------------------
Done /./PreTest 00:00:00.0781290
Task /./Test
----------------------------------------------------------------------
TDD: Tests first!
Executing all tests in C:\projects\psmodules\PSEventLogEntry\Test\PSEventLogEntry.Tests.ps1

Executing script C:\projects\psmodules\PSEventLogEntry\Test\PSEventLogEntry.Tests.ps1

  Describing Get-EventLogEntry Function Basic Tests

    Context Invalid Parameter Set
      [+] Should Throw Error for invalid ServerName 3.35s
      [+] Should Throw Error for invalid LogName 145ms

    Context Valid Parameter Set
      [+] Should Not Throw An Error for the Valid Logname, Start/End Time inputs 189ms
Tests completed in 3.69s
Tests Passed: 3, Failed: 0, Skipped: 0, Pending: 0, Inconclusive: 0
----------------------------------------------------------------------
Done /./Test 00:00:04.7536498
Done /. 00:00:14.5446171
Build succeeded. 4 tasks, 0 errors, 0 warnings 00:00:14.6540113


Discovering tests...OK
Build success
```

REFERENCES

Invoke-build:
https://github.com/nightroman/Invoke-Build

My examples:
https://github.com/ebrucucen/PSModules
https://ci.appveyor.com/project/ebrucucen/psmodules

More github badges?
https://shields.io

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