
AWS
re:Invent

EUC302

# Enhance, expedite, and simplify provisioning and management of Amazon EUC

Greg LaVigne
Sr. Specialist Solutions Architect
AWS

Max Stringer
Desktop On Demand Service Owner
Pfizer

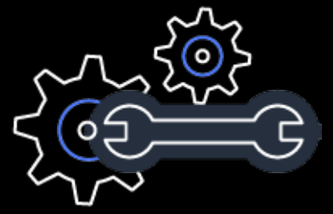
AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

# Session Objectives

- Understand the value proposition of automation and monitoring related to your AWS End User Computing (AWS EUC) service implementations
- Provide methodologies, tools, and examples to consider when automating
- Share implementations and resultant benefits from a “real world” user: WorkSpaces @ Pfizer

# Value Proposition: Well-Architected Framework



Operational excellence



Reliability



Performance efficiency



Cost optimization



Security



Amazon WorkSpaces

- Increases speed and agility and ultimately speed to market to your customers / users
- Repeatable, consistent and reliable
- Optimize costs – pay for only what you actually use
- Reduce support and management costs
- Empower and enable your end users on their terms



Amazon AppStream 2.0

# Image Creation Automation: Options



## AWS CLI and / or AWS Tools for PowerShell

Enable WinRM and configure via PowerShell or CLI



## AWS CloudFormation + Terraform or PowerShell / CLI

Start with AWS CloudFormation, supplement with Terraform and / or scripts

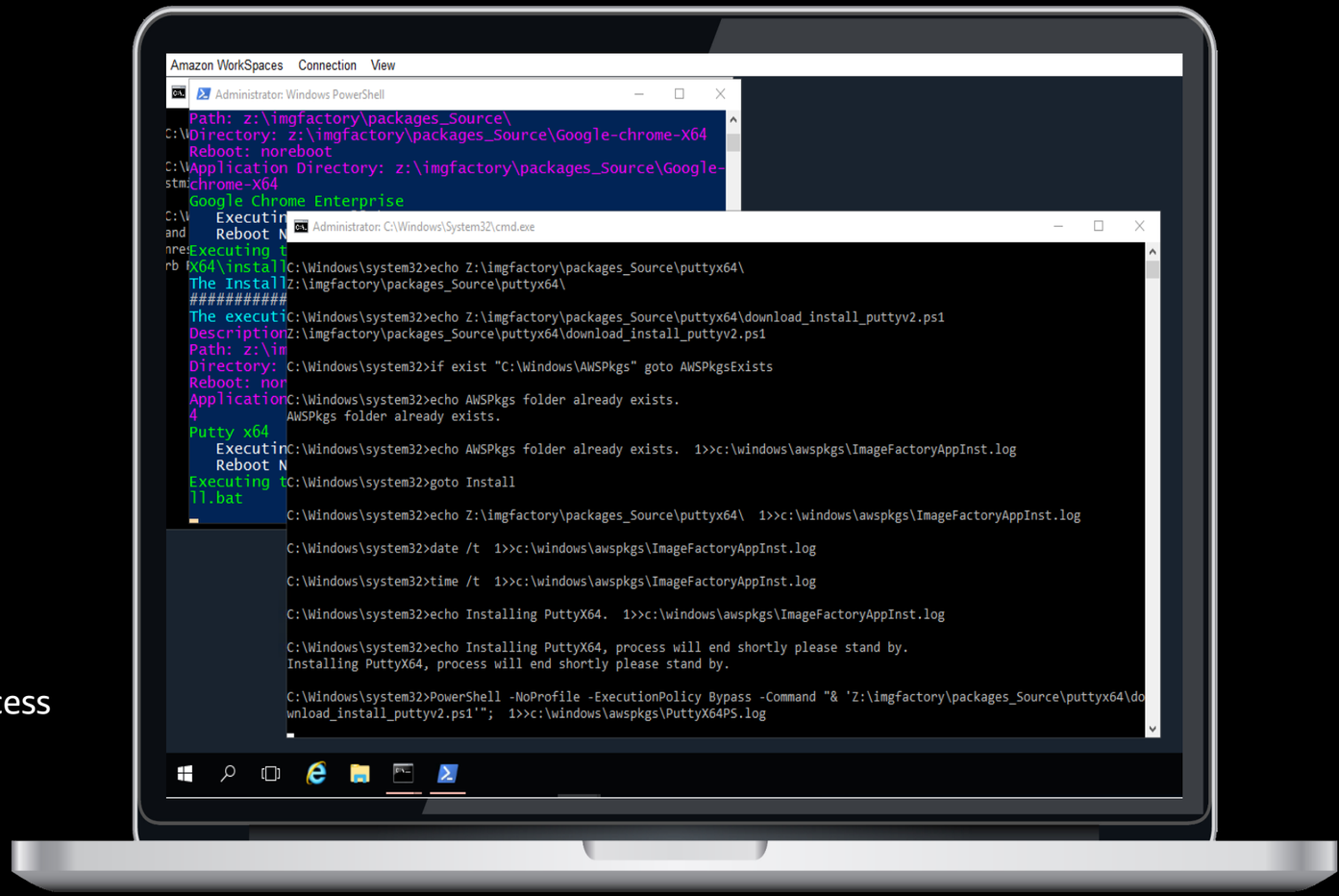


## Existing third-party tools

SCCM, Big Fix, Chocolatey, etc.

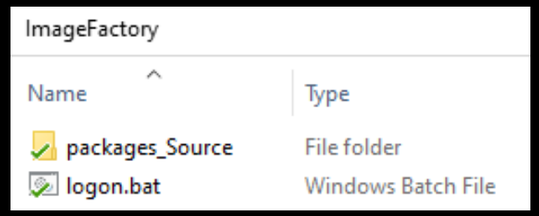
# Light Touch Image Creation: Components

- Automation Tools Leveraged
  - PowerShell
  - AWS Tools for PowerShell
  - .bat/.cmd
  - Group Policy
- Bundle Type
  - Windows 10, Performance, 80 GB C:/50GB D:
- Active Directory Infrastructure
  - Domain Role IDs
    - Map drive to FSx for Windows deployment share
  - Group Policy = Sets login script to kick off automated process
- File Share / Folder Structure
- “Engine” and Manifest Files

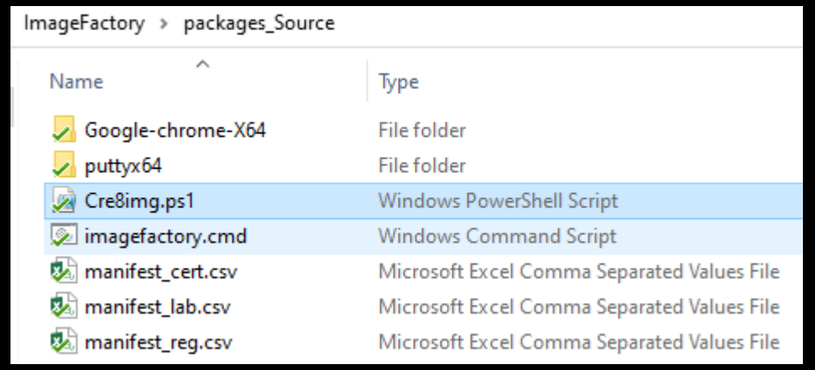


# Light Touch Image Creation: Folder Structure

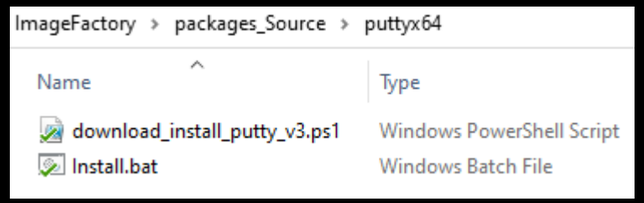
- FSx for Windows share
  - Ex: \\amznfsx.x.y.com\ImageFactory

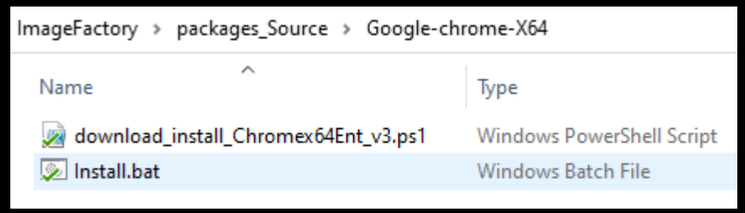



- Nested Package_Source folder
  - Source folder for each application
  - Manifest.csv files
  - Image Creation .ps1 script




- Nested application folders
  - Install.bat
  - Calls app.ps1 to download and install application
  - Uses installer native flags



# Light Touch Image Creation: “The Engine”



- The Manifest.csv defines which app folders to loop into and execute
  - Unique .csv associated to image creation role ID name
    - Manifest_cert.csv = img_cre8_cert ID
    - Manifest_lab.csv=img_cre8_lab ID

```
desc,path,appdir,reboot
Add PSWinUpdates Module,\\AMZNFSXK1UA8G6S\ImageFactory\packages_Source\,PSWinup_Mod,noreboot
Run Windows Updates,\\AMZNFSXK1UA8G6S\ImageFactory\packages_Source\,RunPSWinUps,noreboot
FireFox ESR,\\AMZNFSXK1UA8G6S\ImageFactory\packages_Source\,Firefox_ESR,noreboot
Google Chrome Enterprise,\\AMZNFSXK1UA8G6S\ImageFactory\packages_Source\,Google-chrome-X64,noreboot
Putty x64,\\AMZNFSXK1UA8G6S\ImageFactory\packages_Source\,puttyx64,noreboot
Dual Monitor Tools 2.7,\\AMZNFSXK1UA8G6S\ImageFactory\packages_Source\,dual-monitor-tools,noreboot
Adobe Reader DC,\\AMZNFSXK1UA8G6S\ImageFactory\packages_Source\,AdobeReaderDC,noreboot
VCplusplus2015,\\AMZNFSXK1UA8G6S\ImageFactory\packages_Source\,VCplusplus2015,noreboot
AWS Powershell Tools,\\AMZNFSXK1UA8G6S\ImageFactory\packages_Source\,AWS_tools_ps1,noreboot
MySQL Workbench,\\AMZNFSXK1UA8G6S\ImageFactory\packages_Source\,MySqlWB,noreboot
Notepad++,\\AMZNFSXK1UA8G6S\ImageFactory\packages_Source\,Notepad++,noreboot
WinSCP,\\AMZNFSXK1UA8G6S\ImageFactory\packages_Source\,WinSCP,noreboot
VS Code,\\AMZNFSXK1UA8G6S\ImageFactory\packages_Source\,VSCode,noreboot
AWS_Wallpaper,\\AMZNFSXK1UA8G6S\ImageFactory\packages_Source\,AWS_Wallpaper,noreboot
DualMonitorConfig,\\AMZNFSXK1UA8G6S\ImageFactory\packages_Source\,DualMonitorConfig,noreboot
Win10Optimizations,\\AMZNFSXK1UA8G6S\ImageFactory\packages_Source\,Win10Opts,noreboot
Run Final master image Cleanup,\\AMZNFSXK1UA8G6S\ImageFactory\packages_Source\,Cleanup,noreboot
```

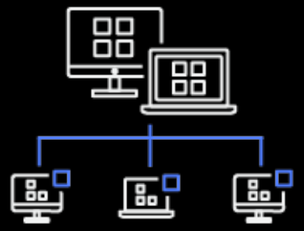
- Image Create .PS1 reads the manifest_*.csv and loops through nested folders calling an install.bat within each to download and / or install applications
  - .ps1 called via domain login script

# Image Creation Automation: Options



AWS CLI and / or AWS Tools for PowerShell*

Enable WinRM and configure via CLI or PS1



AWS Systems Manager**

Leverage AWS Systems Manager Run Command Feature

* https://aws.amazon.com/blogs/desktop-and-application-streaming/creating-an-image-programmatically-with-appstream-2-0-image-assistant-cli-operations/

** https://aws.amazon.com/blogs/desktop-and-application-streaming/how-to-use-ssm-run-to-perform-programmatic-actions-on-appstream-2-0-image-builders/

# Mass Provisioning Automation: PowerShell



Tools leveraged

- PowerShell
- AWS Tools for PowerShell
- Active Directory module for Windows PowerShell

```
param(
    [parameter(Mandatory=$true)]
    [ValidateScript({test-path $_})]
    [string]
    $inputFile
)

$errors = @()
$input = import-csv $inputFile
foreach ($line in $input){
    $icount = 1
    while ($icount -le ($line.count)){
        $username = "$($line.UserPrefix)$($icount.ToString("000"))"
        $wsdir = $line.wsdirectory
        $wsreg = $line.wsregion
        $wsbundle = $line.wsbundleid

    $icount++
    Write-Host $username, $wsdir, $wsreg, $wsbundle, $wstagkey1, $wstagval1 -ForegroundColor yellow
         Try{New-WKSWorkspace -Workspace @{"BundleID" = $wsbundle; "DirectoryId" = $wsdir; "UserName" = $username}-region $wsreg -ErrorAction Stop}
            catch{$errors += "Error creating WorkSpace for: $username "}
            }
    write-host "########################"  -ForegroundColor cyan
}
$errors |Out-GridView
```

# Mass Provisioning Automation: .CSV



1- Provision Domain Users (userinput.csv)*

```
UserPrefix,count,Pass,path
aws-cert-r-,8,wudntUlike2know,"OU=A,OU=Cert,OU=Users,OU=AWS,DC=example,DC=domain,DC=com"
aws-cert-p-,112,wudntUlike2know,"OU=P,OU=Cert,OU=Users,OU=AWS,DC=example,DC=domain,DC=com"
aws-hol-,200,wudntUlike2know,"OU=LAB,OU=Users,OU=AWS,DC=example,DC=domain,DC=com“
aws-spot-p-,125,wudntUlike2know,"OU=SPOT,OU=Users,OU=AWS,DC=example,DC=domain,DC=com”
```

2-Provision Amazon WorkSpaces (wsinput.csv)

```
UserPrefix,count,wsdirectory,wsregion,wsbundleid
aws-cert-r-,8,d-9xxxyyzzz3,us-west-2,wsb-vbsjd64y6
aws-cert-a-,1020,d-9xxxyyzzz3,us-west-2,wsb-72c29fpxq
aws-hol-,312,d-9xxxyyzzz0,us-west-2,wsb-5gsrznp1d
aws-spot-,145,d-9xxxyyzzzb,us-west-2,wsb-5gsrznp1d
```

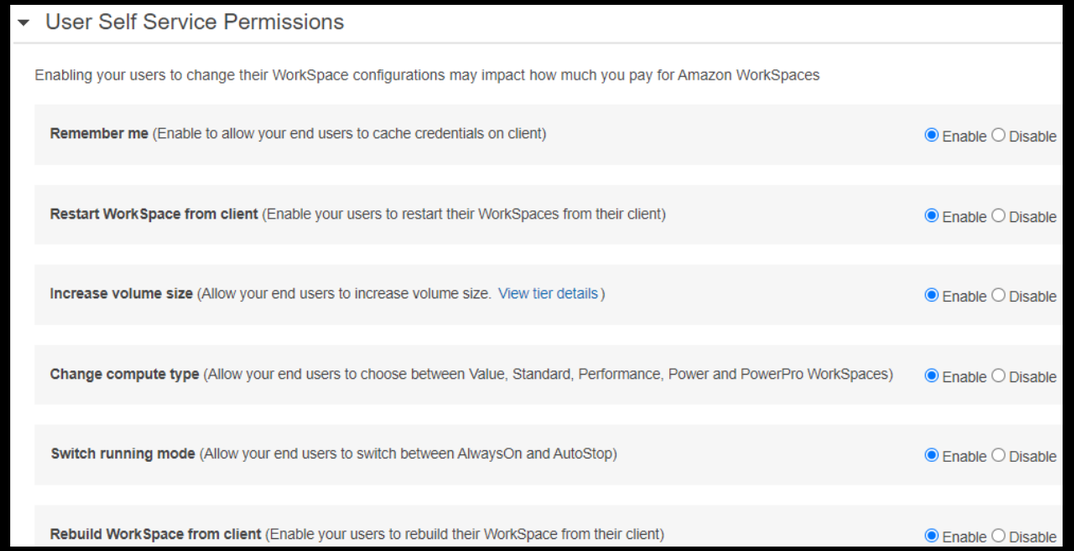
# Day-to-Day Management

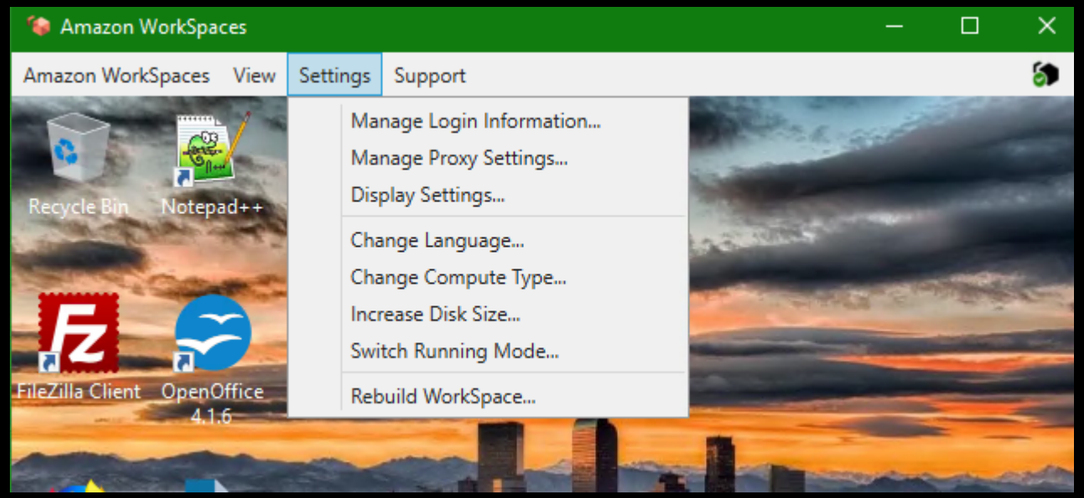


- Proactive monitoring with automated resolution
  - Unhealthy WorkSpaces force reboot
- Service Utilization results in cost optimization
  - Utilization reporting – optimized scaling policies
  - Monitoring WorkSpaces usage – pay for only what's actually being used
- Multi-Region Image Management
  - Use PowerShell and / or the API to automate the copying of updated images and bundles
  - Use PowerShell and / or the API to automate the copying of updated images and fleets
- AWS EUC Support Dashboard – single pane of glass view simplifies support
  - https://aws.amazon.com/getting-started/hands-on/euc-dashboard/

# Self-Service Capabilities: Built in

1. Administrator enables via the Directory Services configuration within WorkSpaces

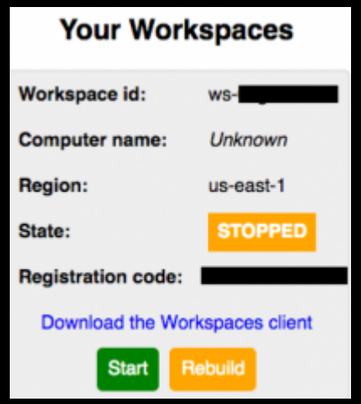



2. Users see enabled options via the WorkSpaces Client menu

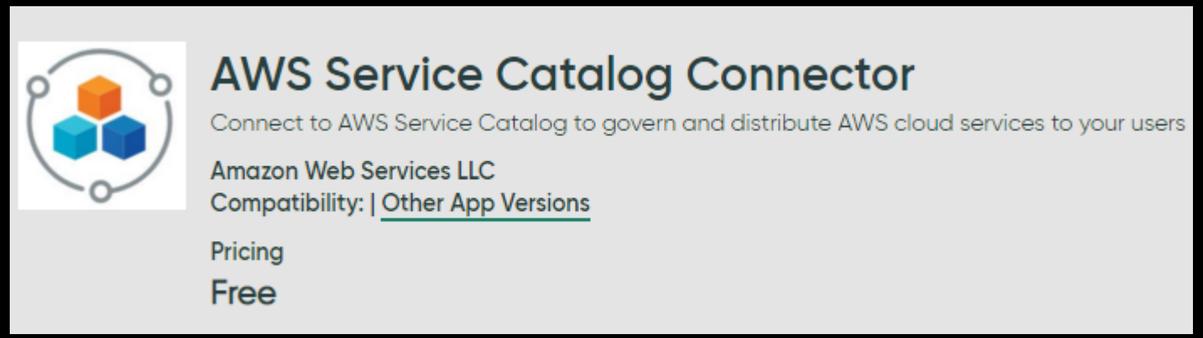
# Self-Service Capabilities: Build or Integrate

Self-Service Portal




https://aws.amazon.com/blogs/desktop-and-application-streaming/creating-a-self-service-portal-for-amazon-workspaces-end-users/

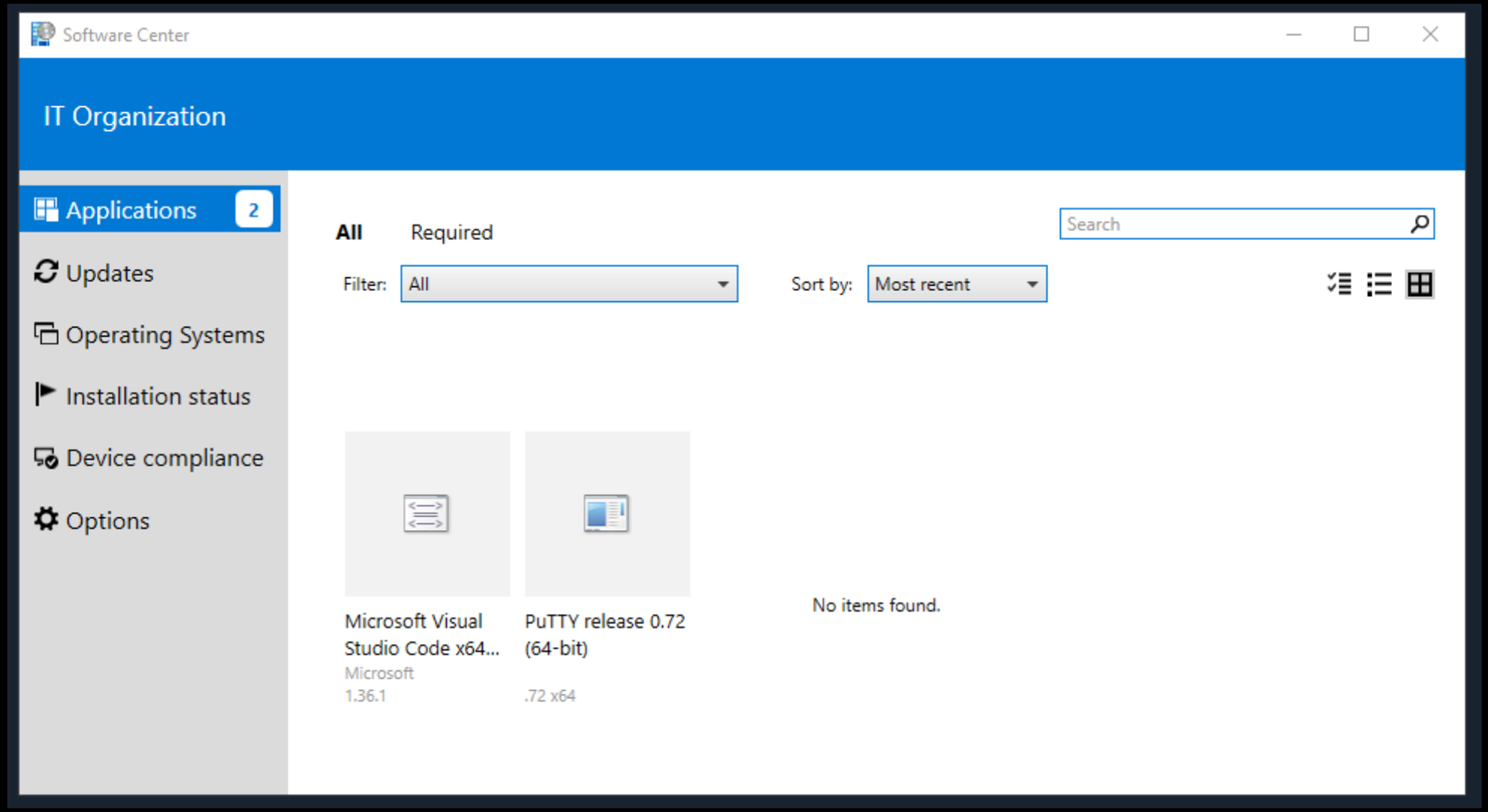
AWS Service Catalog Connector for ServiceNow




https://aws.amazon.com/blogs/mt/how-to-enable-self-service-amazon-workspaces-by-using-aws-service-catalog-connector-for-servicenow/

# Self-Service Capabilities: Existing Tools

Third-party tool capabilities (SCCM Software Center, BigFix, etc.)



# WorkSpaces @ Pfizer



AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

# Context Setting

- One of the largest Pharmaceutical companies in the world
- Globally distributed workforce
- A DaaS user for over a decade
- Primary use case is a contingent worker
- Consistent user experience

# Automation – Why?

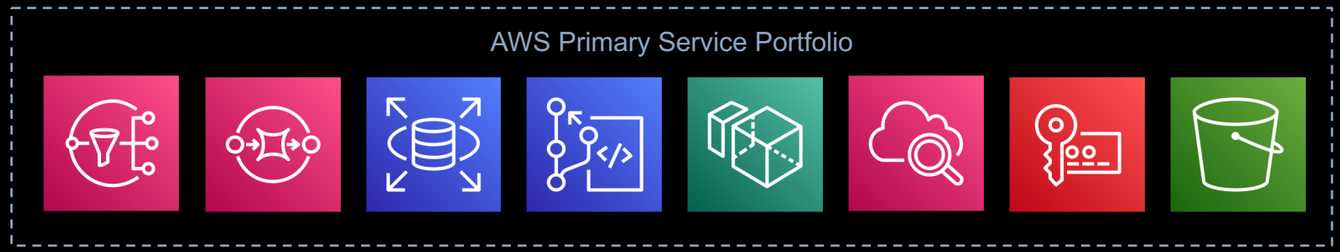
- Integration with existing systems, services, and processes
- Agility
- Reduce support cost
- Improved customer experience

# Automations

- WorkSpace Creation – more than just clicking Launch WorkSpace
- Release to Customer – chicken and egg problem
- Restart
- WorkSpace Termination – scavenge to optimize
- Rebuild
- Compute and / or Storage Upgrade – not so simple

# Key Enablers

- voqDispatch – closed-loop workloads
- Location Services – find a WorkSpace any way / anywhere
- The CHiMP – hello, I'm here
- Brokers
- AWS Enabling Stars – WorkSpaces, SQS, and SNS



# Session recap

- Review of today's objectives
- Next steps
  - Amazon AppStream 2.0 Getting Started Guide – https://bit.ly/35BKG4E
  - Amazon WorkSpaces Getting Started Project – https://bit.ly/3oI5ccy
  - Free Tier
- The End User Computing Blog

  https://aws.amazon.com/products/end-user-computing/blog/
- Want more? Related sessions
  - EUC303 – Scaling and optimizing Amazon WorkSpaces
  - EUC304 – Scaling and optimizing Amazon AppStream 2.0

# Thank you!



AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.



# Please complete the session survey

AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.