

Create Smart Icon for local .EXE

Last Modified on 04.16.26

This article describes how to create a Smart Icon for Microsoft Word and Excel using the [Application Workspace Powershell Module](#).



This code is made available as is, without any warranty of any kind. The entire risk of the use or the results of the use of this code remains with the user.

Microsoft Word

```
# Import the Liquit Workspace module from its default location
$moduleName = "Liquit.Server.PowerShell"
$moduleInstalled = Get-Module -ListAvailable | Where-Object { $_.Name -eq $moduleName }

if ($moduleInstalled) {
    Write-Host "The '$moduleName' module is already installed." -NoNewLine -ForegroundColor green
    Import-Module -Name "Liquit.Server.PowerShell"
    Write-Host ""
} else {
    Write-Host "The '$moduleName' module is not installed. Please install the Application Workspace PowerShell Module" -NoNewLine -ForegroundColor yellow
}

# Set up the credentials for the connection to the Application Workspace
$liquitZoneUsername = "local\admin"
$liquitZonePassword = Read-Host "Enter Password" -AsSecureString
$liquitCredentials = New-Object -TypeName System.Management.Automation.PSCredential -ArgumentList $liquitZoneUsername, $liquitZonePassword
$liquitZone = Read-Host "Enter the uri of your Application Workspace zone"

# Create a connection to the Application Workspace
$liquitContext = Connect-LiquitWorkspace -URI $liquitZone -Credential $liquitCredentials

# Create package information
$wordIconURL = "https://www.recastsoftware.com/ico/word.ico"
$wordIconPath = "c:/temp/tempword.ico"
$packageName = "Word"
$packageDisplayName = "Word"
$packageDescription = "Word"

$webClient = New-Object System.Net.WebClient
$webClient.DownloadFile($wordIconURL, $wordIconPath)

$iconContent = New-LiquitContent -Path $wordIconPath

$package = New-LiquitPackage -Name $packageName -Type "Launch" -DisplayName $packageDisplayName -Description $packageDescription -Priority 100 -Enabled $true -Offline $True -Web $false -Icon $iconContent

# Create a new snapshot
$snapshot = New-LiquitPackageSnapshot -Package $package

# Create a new launch action set
$actionset = New-LiquitActionSet -snapshot $snapshot -Type Launch -name 'Launch' -Enabled $true -Frequency Always -Process StopAtFirstEffectiveAction

# Create the first launch action
```

Recast

```
# Create the first launch action
$actionset_action1 = New-LiquitAction -ActionSet $actionset -Name 'Start Word (x86)' -Type 'processstart' -Enabled $true -IgnoreErrors $false -Settings @{name = '${ProgramFiles32}Microsoft Office\root\Office16\WINWORD.EXE'; parameters = ""; } -Context User

# Define the filter set for the first action
$filterset1 = New-LiquitFilterSet -Action $actionset_action1

# Add a filter to the first action
new-LiquitFilter -FilterSet $filterset1 -type fileexists -Settings @{path = '${ProgramFiles32}Microsoft Office\root\Office16\WINWORD.EXE'; } -Value "true"

# Set the operator for the entity filter to AND
Set-LiquitEntityFilter -Action $actionset_action1 -Operator Or

# Create the second launch action
$actionset_action2 = New-LiquitAction -ActionSet $actionset -Name 'Start Word (x64)' -Type 'processstart' -Enabled $true -IgnoreErrors $false -Settings @{name = '${ProgramFiles}Microsoft Office\root\Office16\WINWORD.EXE'; parameters = ""; } -Context User

# Create the filter set for the second action
$filterset2 = New-LiquitFilterSet -Action $actionset_action2

# Add a filter to the second action
new-LiquitFilter -FilterSet $filterset2 -type fileexists -Settings @{path = '${ProgramFiles}Microsoft Office\root\Office16\WINWORD.EXE'; } -Value "true"

# Set the operator for the entity filter to AND
Set-LiquitEntityFilter -Action $actionset_action1 -Operator Or

# Create the third launch action
New-LiquitAction -ActionSet $actionset -Name 'Word Online'-Type 'openurl' -Enabled $true -IgnoreErrors $false -Settings @{ url = 'https://office.live.com/start/Word.aspx'; browser = 3; } -Context User

# Publish the package
Publish-LiquitPackageSnapshot -Snapshot $snapshot -stage Production

# Set the entitlement
$identity = Get-LiquitUser -ID 00000000-0000-0000-0000-000000000000
New-LiquitPackageEntitlement -Package $package -Identity $identity -Publish Workspace
```

Microsoft Excel

```
# Import the Application Workspace module from its default location
Import-Module "C:\Program Files (x86)\Liquit Workspace\PowerShell\3.0\Liquit.Server.PowerShell.dll" -Prefix "RecastSoftware"

$liquitZoneUsername = "localadmin"
$liquitZonePassWord = Read-Host "Enter Password" -AsSecureString
$liquitCredentials = New-Object -TypeName System.Management.Automation.PSCredential -ArgumentList $liquitZoneUsername, $liquitZonePassWord
$liquitZone = Read-Host "Enter the uri of your Application Workspace zone"

$ExcelIconURL = "https://www.recastsoftware.com/ico/excel.ico"
$ExcelIconPath = "c:\Temp\tempExcel.ico"

$packageName = "Excel"
$packageDisplayName = "Excel"
$packageDescription = "Excel"

Connect-LiquitWorkspace -URI $liquitZone -Credential $liquitCredentials
```

```
$webClient = New-Object System.Net.WebClient $webClient.DownloadFile($ExcellconURL,$ExcellconPath)

$iconContent = New-LiquitContent -Path $ExcellconPath
$package = New-LiquitPackage -Name $packageName -Type "Launch" -DisplayName $packageDisplayName Descripti
on $packageDescription -Priority 100 -Enabled $true -Offline $True -Web $false -Icon $iconContent

# Create the snapshot for this package #
$snapshot = New-LiquitPackageSnapshot -Package $package

# Define launch action set
$actionset = New-LiquitActionSet -snapshot $snapshot -Type Launch -name 'Launch' -Enabled $true -Frequency Alway
s ` -Process StopAtFirstEffectiveAction

# Define the first launch action
$actionset_action1 = New-LiquitAction -ActionSet $actionset -Name 'Start Excel (x86)' -Type 'processstart' -Enabled $t
rue -IgnoreErrors $false -Settings @{{name = '${ProgramFiles32}Microsoft Office16Excel.EXE'; parameters
= ""};} -Context User

# Define the filter set for the first action
New-LiquitFilterSet -Action $actionset_action1

# Add a filter to the first action
New-LiquitFilter -FilterSet $filterset1 -type fileexists -Settings @{{path = '${ProgramFiles32}Microsoft Office
16Excel.EXE';} -Value "true"

# Set the operator for the entity filter to AND
Set-LiquitEntityFilter -Action $actionset_action1 -Operator Or

# Create the second launch action
$actionset_action2 = New-LiquitAction -ActionSet $actionset -Name 'Start Excel (x64)' -Type 'processstart' -Enabled $t
rue -IgnoreErrors $false -Settings @{{name = '${ProgramFiles}Microsoft Office16Excel.EXE'; parameters = "
";} -Context User

# Create the filter set for the first action
$filterset2 = New-LiquitFilterSet -Action $actionset_action2

# Add a filter to the first action
$filter2 = new-LiquitFilter -FilterSet $filterset2 -type fileexists -Settings @{{path = '${ProgramFiles}Microsoft Office16Excel.EXE';} -Value "true"

# Set the operator for the entity filter to AND
Set-LiquitEntityFilter -Action $actionset_action1 -Operator Or

# Create the third launch action
New-LiquitAction -ActionSet $actionset -Name 'Excel Online' -Type 'openurl' -Enabled $true -IgnoreErrors $false -Settin
gs @{{ url = 'https://office.live.com/start/excel.aspx'; browser = 3; } -Context User

# Publish the package
Publish-LiquitPackageSnapshot -Snapshot $snapshot -stage Production

# Set the entitlement
$identity = Get-LiquitUser -ID 00000000-0000-0000-0000-000000000000
New-LiquitPackageEntitlement -Package $package -Identity $identity -Publish Workspace
```