

Description: An introduction to the new PowerShell enhancements in 10.96.

General Requirement: Basic knowledge of GENESIS64 and its architecture. Basic knowledge of Windows PowerShell.

Improvements to Existing Commands

The "name" parameter, used to identify an entity in many Workbench commands, has been enhanced to allow a wider variety of formats. Now when supplying a value for the "name" parameter you can choose to use:

- A string representing the name of the entity
- A string representing the full path of the entity (for example, "Assets/Equipments/Alarms")

" for examples of how to use some of these new commands.

PowerShell caches commands, so users who previously used ICONICS PowerShell modules before installing this beta will not immediately have access to any new or updated PowerShell commands. This can be solved by running this command:

```
Get-Module -ListAvailable -Refresh
```

The command only needs to be executed once after the beta is installed.

Log in or out of ICONICS Security

The user can now log in or out of ICONICS security via PowerShell. There are synchronous and asynchronous options for each command.

- Set-WbUserLogin
- Set-WbUserLoginAsync
- Set-WbUserLogout
- Set-WbUserLogoutAsync

Copy, Paste, and Multiply

Users can now use PowerShell to invoke the Workbench's Copy, Paste, and Multiply features.

- Set-WbCopyPaste
- New-WbMultiplyProperty
- Set-WbMultiply

- An integer representing the identifier of the entity in the database
- The IEntityKey representing the key of the entity
- The IEntity representing the entity itself

As an additional enhancement, the "name" parameter now supports wildcards.

New Commands

Here are the most notable new PowerShell commands added to Workbench in 10.96.

- See "Appendix: PowerShell Command Examples"

Project Management

We now have commands to work with applications, application projects, application servers, audit messages, available file paths, and SQL Server connections.

- Projects
 - Get-WbAppProjects
 - Get-WbAppProject
 - New-WbAppProject
 - Set-WbAppProject
 - Remove-WbAppProject
- Servers
 - Get-WbAppServers
 - Get-WbAppServersByParent
 - New-WbAppServer
 - Remove-WbAppServer
- Applications ("Configure Application(s) Settings" page)
 - Get-WbApplications
 - Get-WbApplicationsByParent
 - New-WbApplication
 - Remove-WbApplication
- Audit Log
 - Get-WbAuditMessages
 - Set-WbEnableAuditLog
- Available File Paths (Project tab > "File Configuration")
 - Get-WbFileInfos
 - Get-WbFileInfo
 - New-WbFileInfo
 - Set-WbFileInfo
 - Remove-WbFileInfo
- SQL Server info (Project tab > SQL Configuration)

- Get-WbSqlServerInfoes
- Get-WbSqlServerInfo
- New-WbSqlServerInfo
- Set-WbSqlServerInfo
- Remove-WbSqlServerInfo

Task Management

These commands allow you to interact with the Recent Tasks panel of Workbench. This allows you to retrieve exported files and cancel tasks.

- Get-WbTasks
- Get-WbTask
- Get-WbExportFile (downloaded an exported file)
- Get-WbPackFile (download a pack-and-go file)
- Set-WbCancelTask
- Remove-WbTasks

Database Commands

These new commands allow you to perform the same database actions as in Workbench.

- Get-WbSqlServerInfo

- Set-WbBackupDatabase
- Get-WbRestorePoints
- Set-WbRestoreDatabase
- Set-WbAttachDatabase
- Set-WbDetachRemoveDatabase

Find and Replace

Find and Replace is a new feature of Workbench that will only be accessible via PowerShell for version 10.96. Users can search across their entire project to do text replacements. We plan to add native user interface support for Find and Replace after the release of 10.96.

- New-WbFindOnOption
- New-WbType
- Set-WbFind
- Set-WbReplace

Recursively Delete

Objects and all of their children can now be deleted at once:

- Remove-WbRecursive

Appendix: PowerShell Command Examples

Log In and Log Out

```
# Display a login dialog
Set-WbUserLogin
Set-WbUserLoginAsync
```

```
# Log out the current user
Set-WbUserLogout
Set-WbUserLogoutAsync
```

```
# Log in with a given user and password
# This example uses a user named "MyUser" with a password of "myPassword"
$secpasswd = ConvertTo-SecureString "myPassword" -AsPlainText -Force
$mycreds = New-Object System.Management.Automation.PSCredential ("MyUser", $secpasswd)
Set-WbUserLoginAsync -Credentials $mycreds
```

Multiply Global Aliases in a Folder

```
# Before running, go to Platform Services > Global Aliasing > Aliases.
# Create a folder named Folder, and within it create an alias named Alias.
```

```
# showing the original data
$parent = Get-GasFolder Folder
Get-GasAliasesByParent $parent
```

Workbench – New PowerShell Enhancements in 10.96



Application Note

November 2019

```
# creating the multiply properties
$multiplyProperty = New-WbMultiplyProperty -PropertyName Name -BaseValue 'Alias' -FirstNumber 1 -
NumericPlaces 3
$multiplyProperties = @($multiplyProperty)

# getting the source
$source = Get-GasAlias Alias

# running the multiply
Set-WbMultiply -Provider GlobalAliasing -SourceKey $source.Key -Count 10 -IsDeepCopy 1 -
MultiplyProperties $multiplyProperties

# show data again
Get-GasAliasesByParent $parent

# Your folder should now have ten more aliases in it.
```

Project Management

```
# File Infoes
Get-WbFileInfos
New-WbFileInfo -Name Test -Path 'C:\Users\username\Desktop\' -Server ServerName -IncludeSubfolders
1
Get-WbFileInfo Test
Set-WbFileInfo -Name Test -IncludeSubFolder 0
Get-WbFileInfo Test
Remove-WbFileInfo Test
Get-WbFileInfo Test

# Sql Server Info
Get-WbSqlServerInfos
Get-WbSqlServerInfo "(local)\SQLEXPRESS2017"
New-WbSqlServerInfo -Name ServerName -Username sa -Password a -UseWindowsSecurity 0
Get-WbSqlServerInfo ServerName
Set-WbSqlServerInfo -Name ServerName -UseWindowsSecurity 1 -Password $null -Username $null
Get-WbSqlServerInfo ServerName
Remove-WbSqlServerInfo ServerName
Get-WbSqlServerInfo ServerName

# Audit Log
Set-WbEnableAuditLog -IsEnabled 1
Get-WbAuditMessages

# Application Projects
Get-WbAppProjects
$testProject = New-WbAppProject -Name TestProject -Author AuthorName -Version '01.00.000.00'
Get-WbAppProject $testProject
Set-WbAppProject -Name $testProject -Description Description

# Application Servers
Get-WbAppServers
Get-WbAppServersByParent $testProject
$testAppServer = New-WbAppServer -ParentName $testProject.Key -Name localhost
Get-WbAppServer $testAppServer

# Applications
Get-WbApplications
Get-WbApplicationsByParent $testAppServer
$sqlServerInfo = Get-WbSqlServerInfo '(local)'
$testApplication = New-WbApplication -ParentName $testAppServer.Key -ProviderID ALERT -Name
'Alertworx' -Catalog IcoUnifiedConfig -DataSourceID $sqlServerInfo.Key
Get-WbApplicationsByParent $testAppServer
```

Workbench – New PowerShell Enhancements in 10.96



Application Note

November 2019

```
# deleting all
Remove-WbApplication $testApplication
Remove-WbAppServer $testAppServer
Remove-WbAppProject $testProject
```

Task Management

```
# downloading a file
Get-WbTasks
$task = Get-WbTask 1
Get-WbExportFile -Task $task -TargetPath C:\Users\UserName\Desktop\Export.csv

# canceling a task
Get-WbTasks
$task = Get-WbTask 2
Set-WbCancelTask $task

# removing the tasks
$tasks = Get-WbTasks | select -ExpandProperty Key
$tasks
Remove-WbTasks -Keys $tasks
```

Database Manipulation

```
# Example assumes you have a database named "Test" in the "(local)\SQLEXPRESS2017" instance.
# getting the SQL Server instance
$sqlServer = Get-WbSqlServerInfo '(local)\SQLEXPRESS2017'

# getting the restore points
Get-WbRestorePoints -SqlServerKey $sqlServer.Key -DatabaseName Test

# creating the backup
Set-WbBackupDatabase -SqlServerKey $sqlServer.Key -DatabaseName Test -BackupName Backup001 -
BackupDescription 'Description'

# getting the restore points
Get-WbRestorePoints -SqlServerKey $sqlServer.Key -DatabaseName Test

# getting the first restore point
$restorePoint = (Get-WbRestorePoints -SqlServerKey $sqlServer.Key -DatabaseName Test)[0]

# restoring the database
Set-WbRestoreDatabase -SqlServerKey $sqlServer.Key -DatabaseName Test -BackupName
$restorePoint.Path

# detach the database
Set-WbDetachRemoveDatabase -SqlServerKey $sqlServer.Key -DatabaseName Test -ShouldDetach 1

# attach the database
Set-WbAttachDatabase -SqlServerKey $sqlServer.Key -DatabaseName Test -FileToAttach 'C:\Program
Files\Microsoft SQL Server\MSSQL13.MSSQLSERVER\MSSQL\DATA\Test.mdf'

# remove the database
Set-WbDetachRemoveDatabase -SqlServerKey $sqlServer.Key -DatabaseName Test -ShouldDetach 0
```

Find and Replace

```
# Replace "Signal" with "Signals" in all global aliasing folders and aliases.
# find option 1
$type_1 = New-WbType -TypeName GasAlias
$findOn_1 = New-WbFindOnOption -EntityType $type_1 -PropertyName Name
```

Workbench – New PowerShell Enhancements in 10.96



Application Note

November 2019

```
# find option 2
$type_2 = New-WbType -TypeName GasFolder
$findOn_2 = New-WbFindOnOption -EntityType $type_2 -PropertyName Name

# find options
$findOn = @($findOn_1, $findOn_2)

# parent key
$parentKey = New-WbRootKey

# requested property names
$requestedPropertyNames = @('Name')

# find
$findResults = Set-WbFind -Provider GlobalAliasing -Findwhat Signal -FindOn $findOn -ParentKey
$parentKey -RequestedPropertyNames $requestedPropertyNames

# replace on
$replaceOn = $findResults | select -ExpandProperty Entity | select -ExpandProperty Key

# replace
$replaceResults = Set-WbReplace -Provider GlobalAliasing -Findwhat Signal -FindOn $findOn -
ParentKey $parentKey -RequestedPropertyNames $requestedPropertyNames -ReplaceWith Signals -
ReplaceOn $replaceOn
```