

# Performing Tasks on Remote Machines using CIM

---



**Liam Cleary**

CEO / MICROSOFT MVP / MICROSOFT CERTIFIED TRAINER

@shareplicity [www.shareplicity.com](http://www.shareplicity.com) | @helloitsliam [www.helloitsliam.com](http://www.helloitsliam.com)



# Overview



**Common Tasks using CIM**

**Performing Tasks on Remote Computers**

**Managing Scheduled Tasks using CIM**

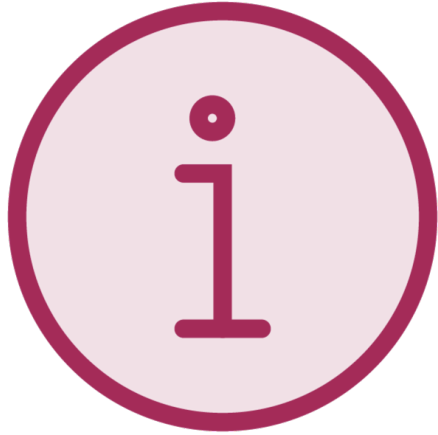


# Common Tasks using CIM

---



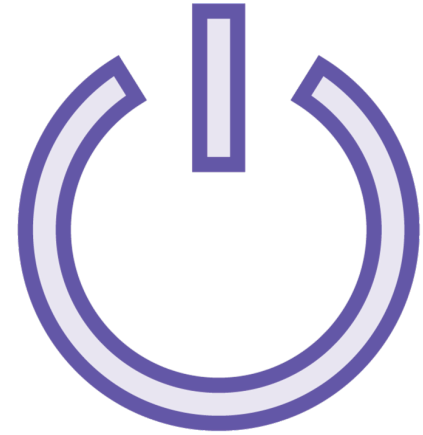
# Common CIM Tasks



**Retrieve Information  
from Windows Clients  
and Servers**

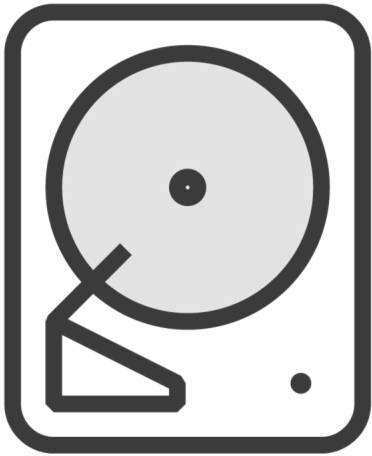
K	V

**Set Registry Key  
Values within  
Windows Clients and  
Servers**

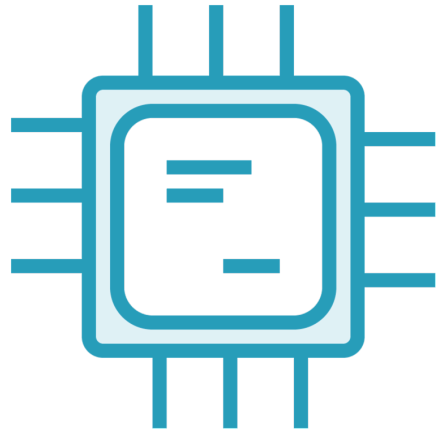


**Start, Stop and Restart  
Services within  
Windows Clients and  
Servers**

# Common Computer CIM Information



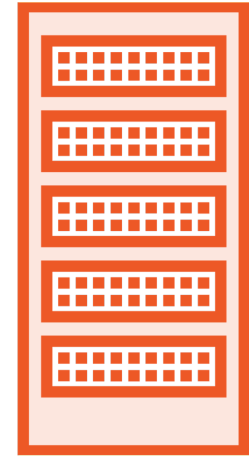
Disk Information



Processor  
Information

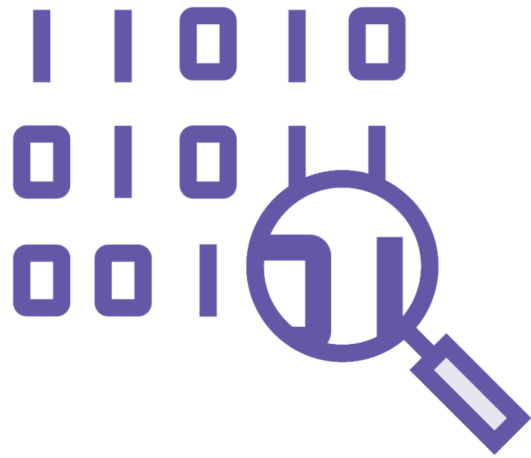


Memory  
Information



Operating  
System  
Information

# Finding CIM and Win32 Classes



**Launch PowerShell Terminal**

**Type: `Get-CimInstance Win32_` or `CIM_`**

**Press: Control + Space**

**Select: Y**

**Review the list of returned Classes**

# Retrieve CIM Information Locally

## # Retrieve Disk Information (Win32 and CIM)

```
Get-CimInstance -ClassName Win32_LogicalDisk
```

```
Get-CimInstance -ClassName CIM_LogicalDisk
```

## # Retrieve Processor Information (Win32 and CIM)

```
Get-CimInstance -ClassName Win32_Processor
```

```
Get-CimInstance -ClassName CIM_Processor
```

## # Retrieve Memory Information (Win32 and CIM)

```
Get-CimInstance -ClassName Win32_PhysicalMemory
```

```
Get-CimInstance -ClassName CIM_PhysicalMemory
```

## # Retrieve Operating System Information (Win32 and CIM)

```
Get-CimInstance -ClassName Win32_OperatingSystem
```

```
Get-CimInstance -ClassName CIM_OperatingSystem
```



# Performing Tasks on Remote Computers

---





# Retrieve CIM Information Remotely

## # Set Remote Computer

```
$computer = "Trainer"
```

## # Retrieve Disk Information (Win32 and CIM)

```
Get-CimInstance -ClassName Win32_LogicalDisk -ComputerName $computer
```

```
Get-CimInstance -ClassName CIM_LogicalDisk -ComputerName $computer
```

## # Retrieve Processor Information (Win32 and CIM)

```
Get-CimInstance -ClassName Win32_Processor -ComputerName $computer
```

```
Get-CimInstance -ClassName CIM_Processor -ComputerName $computer
```

## # Retrieve Memory Information (Win32 and CIM)

```
Get-CimInstance -ClassName Win32_PhysicalMemory -ComputerName $computer
```

```
Get-CimInstance -ClassName CIM_PhysicalMemory -ComputerName $computer
```

## # Retrieve Operating System Information (Win32 and CIM)

```
Get-CimInstance -ClassName Win32_OperatingSystem -ComputerName $computer
```

```
Get-CimInstance -ClassName CIM_OperatingSystem -ComputerName $computer
```



# Identifying CIM Class Methods

```
# Retrieve CIM Class and Identify the Methods
```

```
$class = Get-CimClass -ClassName Win32_Process  
$class.CimClassMethods
```

```
# Retrieve CIM Processes Class, using a query for "Notepad"
```

```
$process = Get-CimInstance `  
    -Query 'Select * From Win32_Process Where Name LIKE "Notepad%"'
```

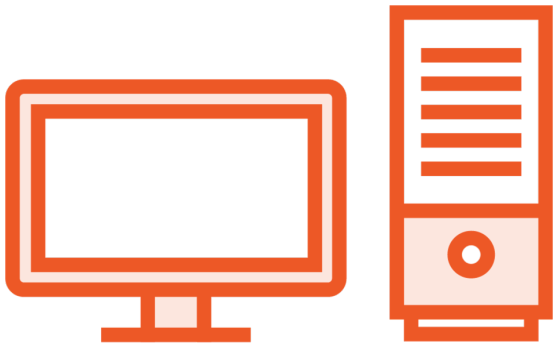
```
# Retrieve CIM Processes Class, using a filter for "Notepad"
```

```
$process = Get-CimInstance `  
    -ClassName Win32_Process `  
    -Filter "Name = 'Notepad.exe'"
```



# Invoke-CimMethod

Invokes a method of a CIM class or CIM instance using the name-value pairs specified as arguments



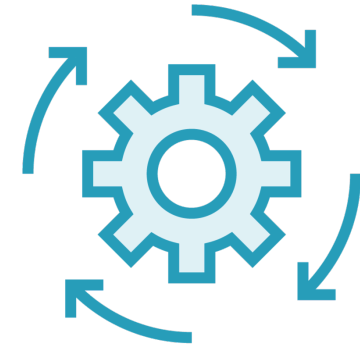
## Computer Name

If specified, will initiate a remote session and execute the command remotely



## Query

Used to filter the CIM Classes to the specified



## Method Name

Executes the specific method within the command for the specified CIM Class



# Executing a Method

**# Retrieve CIM Processes Class, filter to "Notepad", then use "Terminate" method**

```
$process = Get-CimInstance `
    -Query 'Select * From Win32_Process Where Name LIKE "Notepad%"'
```

**# Retrieve list of supported Methods**

```
$cimclass = $process.CimClass | Select-Object CimClassName
Get-CimClass -ClassName $cimclass.CimClassName | `
    Select-Object -ExpandProperty CimClassMethods
```

**# Terminate the process**

```
Invoke-CimMethod -InputObject $process -MethodName Terminate
```



# Demo



**Perform Tasks on Remote Computers  
using the Invoke-CimMethod Command**



# Managing Scheduled Tasks using CIM

---

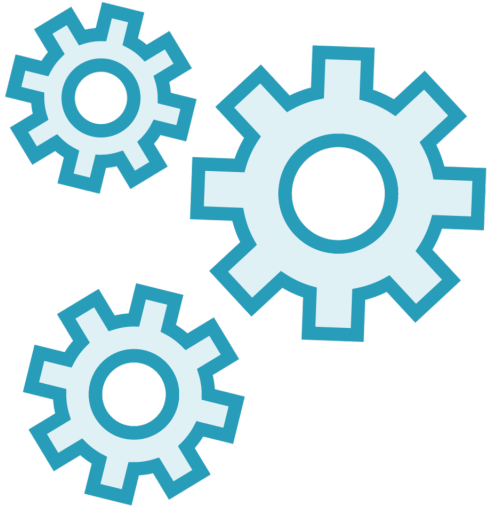


# Tasks

PowerShell CIM does not provide task or job management. Standard PowerShell commands for jobs can be executed within the CIM connection



# PowerShell Tasks and Jobs



Start-Job Command



"AsJob" Parameter



# Executing a Job Locally

**# Initiate Job Locally**

```
Start-Job -ScriptBlock { Get-CimInstance -ClassName Win32_ComputerSystem }
```

**# Retrieve CIM Processes Class, filter to "Notepad", then use "Terminate" method**

```
$process = Get-CimInstance `
    -Query 'Select * From Win32_Process Where Name LIKE "Notepad%">'
```

**# Initiate Job and return the results**

```
$job = Start-Job -ScriptBlock { Get-CimInstance `
    -Query 'Select * From Win32_Process Where Name LIKE "Notepad%">' }
Receive-Job -Id $job.Id
```



# Executing a Job Locally

## # Initiate Job Remotely

```
$computer = "Trainer"
$script = {
    $session = New-CimSession -ComputerName $computer
    Get-CimInstance -ClassName Win32_ComputerSystem -CimSession $session
}

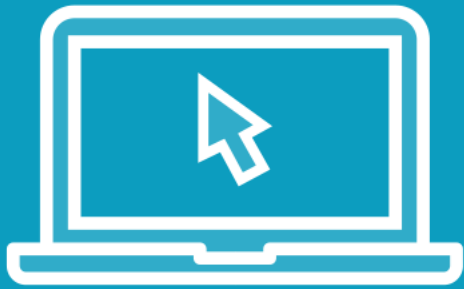
$job = Start-Job -ScriptBlock $script
```

## # Retrieve Remote Job Information

```
Receive-Job -Id $job.Id
```



# Demo



Execute Jobs Locally

Execute Jobs Remotely



# Summary



**Reviewed common Tasks that can be performed using CIM**

**Understood how CIM can be used to execute remote tasks**

**Utilized scheduled tasks with CIM**

