

# Microsoft System Center 2012 R2

## **Cmdlet Reference for System Center 2012 R2 Service Manager (Data Warehouse)**

---

Microsoft Corporation

Published: November 1, 2013

### **Applies To**

System Center 2012 R2 Service Manager

### **Feedback**

Send suggestions and comments about this document to [sc2012docs@microsoft.com](mailto:sc2012docs@microsoft.com).

# Copyright

---

This document is provided "as-is". Information and views expressed in this document, including URL and other Internet website references, may change without notice.

Some examples depicted herein are provided for illustration only and are fictitious. No real association or connection is intended or should be inferred.

This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes. You may modify this document for your internal, reference purposes.

© 2013 Microsoft Corporation. All rights reserved.

Microsoft, Active Directory, Bing, Excel, Hyper-V, Internet Explorer, Silverlight, SQL Server, Windows, Windows Intune, Windows PowerShell, Windows Server, and Windows Vista are trademarks of the Microsoft group of companies. All other trademarks are property of their respective owners.

## Revision History

Release Date	Changes
November 1, 2013	Initial release of this document.

---

## Contents

Disable-SCDWJob .....	5
Disable-SCDWJobCategory .....	9
Disable-SCDWJobSchedule .....	13
Disable-SCDWSource .....	16
Enable-SCDWJob .....	20
Enable-SCDWJobCategory .....	24
Enable-SCDWJobSchedule .....	28
Enable-SCDWSource .....	31
Get-SCDWEntity .....	35
Get-SCDWJob .....	40
Get-SCDWJobModule .....	44
Get-SCDWJobSchedule .....	49
Get-SCDWRetentionPeriod .....	52
Get-SCDWSource .....	56
Get-SCDWSourceType .....	59
Get-SCDWWatermark .....	61
New-SCDWSourceType .....	64
Register-SCDWSource .....	67
Set-SCDWJobSchedule .....	72
Set-SCDWRetentionPeriod .....	77
Set-SCDWSource .....	82
Set-SCDWWatermark .....	87
Start-SCDWJob .....	91
Stop-SCDWJob .....	94
Unregister-SCDWManagementPack .....	98



# Disable-SCDWJob

---

## Disable-SCDWJob

Disables a data warehouse job to prevent it from running in the future.

### Syntax

Parameter Set: Default

```
Disable-SCDWJob [-JobName] <String> [-ComputerName <String> ] [-Credential <PSCredential> ]  
[-Confirm] [-WhatIf] [ <CommonParameters>]
```

### Detailed Description

The **Disable-SCDWJob** cmdlet disables a job, which prevents it from running in the future. When a job is disabled, any remaining modules that have not run yet will complete. After the job completes its current run, it will not run according to its specified schedule and you cannot run it manually.

To re-enable a disabled job, use the **Enable-SCDWJob** cmdlet.

### Parameters

#### **-ComputerName<String>**

Specifies the name of the computer on which the System Center Data Access service is running. The user account that is defined in the **Credential** parameter must have access rights to the specified computer. You can omit this parameter only if the System Center Data Access Service is running on the same computer that has Service Manager installed.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

---

## **-Credential<PSCredential>**

Specifies the credentials to use when you are connecting to the server on which the System Center Data Access service is running. The user account that is provided must have access to that server.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-JobName<String>**

Specifies the Data Warehouse job to disable. The **JobName** parameter is a mandatory parameter.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-Confirm**

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

---

## -WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## <CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see [about\\_CommonParameters](#).

## Inputs

The input type is the type of the objects that you can pipe to the cmdlet.

- **None.**You cannot pipe input to this cmdlet.

## Outputs

The output type is the type of the objects that the cmdlet emits.

- **None.**This cmdlet does not generate any output.

## Examples

### ----- EXAMPLE 1 -----

This command disables the Extract\_Contoso job.

```
PS C:\> Disable-SCDWJob -JobName Extract_Contoso -ComputerName serverDW7
```

### ----- EXAMPLE 2 -----

This command lists the jobs according to their **IsEnabled** state.

```
PS C:\> Get-SCDWJob -ComputerName serverDW7 | Sort-Object -Property IsEnabled,Name | Format-Table -Property Name,IsEnabled -AutoSize
```

---

### ----- EXAMPLE 3 -----

The first command obtains user credentials to store in a variable for the *Credential* parameter. The second command disables all the jobs using the **ForEach-Object** cmdlet and the specified credentials.

```
PS C:\> $creduser1 = Get-Credential
PS C:\> Get-SCDWJob -ComputerName serverDW7 -credential $creduser1 | ForEach-Object
{$_ .DisableJob() }
```

## Related topics

[Enable-SCDWJob](#)

[Get-SCDWJob](#)

[Start-SCDWJob](#)



---

# Disable-SCDWJobCategory

---

## Disable-SCDWJobCategory

Disables all jobs within a job category.

### Syntax

Parameter Set: Default

```
Disable-SCDWJobCategory [-JobCategoryName] <String> [-ComputerName <String> ] [-Credential <PSCredential> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

### Detailed Description

The **Disable-SCDWJobCategory** cmdlet disables all jobs within a job category.

### Parameters

#### **-ComputerName<String>**

Specifies the name of the computer on which the System Center Data Access service is running. The user account that is defined in the **Credential** parameter must have access rights to the specified computer. You can omit this parameter only if the System Center Data Access Service is running on the same computer that has Service Manager installed.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

#### **-Credential<PSCredential>**

Specifies the credentials to use when you are connecting to the server on which the System Center Data Access service is running. The user account that is provided must have access to that server.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-JobCategoryName<String>**

Specifies the name of the job category. You can use the **Get-SCDWJob** cmdlet to retrieve this value. Examples of valid job categories are Maintenance, Extract, Load, and Transform.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-Confirm**

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-WhatIf**

Describes what would happen if you executed the command without actually executing the command.

---

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## <CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see [about\\_CommonParameters](#).

## Inputs

The input type is the type of the objects that you can pipe to the cmdlet.

- **None.**You cannot pipe input to this cmdlet.

## Outputs

The output type is the type of the objects that the cmdlet emits.

- **None.**This cmdlet does not generate any output.

## Examples

### ----- EXAMPLE 1 -----

This command disables the jobs in the *CubeProcessing* category.

```
PS C:\> Disable-SCDWJobCategory -ComputerName serverDW7 -JobCategoryName CubeProcessing
```

### ----- EXAMPLE 2 -----

The first command stores user credentials for the *Credential* parameter.

The second command disables the jobs in the *CubeProcessing* category using the specified credentials.

```
PS C:\> $credUser = Get-Credential
```

```
PS C:\> Disable-SCDWJobCategory -ComputerName serverDW7 -JobCategoryName CubeProcessing -  
Credential $credUser
```

---

### ----- EXAMPLE 3 -----

This command provides information about enabled and disabled jobs in the job categories. If all the jobs in the category are either enabled or disabled, the value in the **IsEnabled** column in the output reflects that value.

If at least one job in a job category has an opposite **IsEnabled** value from the other jobs in the same category, the job category name is repeated with that value (as is the case for the Load category in the example output). This is achieved by using the **Unique** parameter with the **Select-Object** command.

```
PS C:\> Get-SCDWJob -ComputerName serverDW7 | Select-Object -Property CategoryName,
IsEnabled -Unique | Format-Table -Property CategoryName, IsEnabled -AutoSize
```

---

# Disable-SCDWJobSchedule

---

## Disable-SCDWJobSchedule

Disables a Data Warehouse job schedule, which causes the job schedule to stop initiating jobs.

### Syntax

Parameter Set: Default

```
Disable-SCDWJobSchedule [-JobName] <String> [-ComputerName <String> ] [-Confirm] [-WhatIf] [  
<CommonParameters>]
```

### Detailed Description

The **Disable-SCDWJobSchedule** cmdlet disables a data warehouse job schedule, which causes the job schedule to stop initiating jobs. Disabling the job schedule retains the job schedule settings. To modify the job schedule settings, use the **Set-SCDWJobSchedule** cmdlet.

### Parameters

#### **-ComputerName<String>**

Specifies the name of the computer on which the System Center Data Access service is running. You can omit this parameter only if the System Center Data Access Service is running on the same computer that has Service Manager installed.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

#### **-JobName<String>**

Specifies the Data Warehouse job for which the schedule will be disabled.

---

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-Confirm**

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-WhatIf**

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **<CommonParameters>**

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see [about\\_CommonParameters](#).

---

## Inputs

The input type is the type of the objects that you can pipe to the cmdlet.

- **None.** You cannot pipe input to this cmdlet.

## Outputs

The output type is the type of the objects that the cmdlet emits.

- **None.** This cmdlet does not generate any output.

## Examples

### ----- EXAMPLE 1 -----

This command disables the job schedule for the Transform.Common job.

```
PS C:\> Disable-SCDWJobSchedule -ComputerName serverDW7 -JobName Transform.Common
```

### ----- EXAMPLE 2 -----

This command shows the all the jobs and whether their schedule is enabled or disabled.

```
PS C:\> Get-SCDWJobSchedule -ComputerName serverDW7 | Format-Table -Property Name,
ScheduleEnabled
-AutoSize
```

## Related topics

[Enable-SCDWJobSchedule](#)

[Get-SCDWJobSchedule](#)

[Set-SCDWJobSchedule](#)

---

# Disable-SCDWSource

---

## Disable-SCDWSource

Disables all jobs that are affiliated with the specified data source.

### Syntax

Parameter Set: Default

```
Disable-SCDWSource [-DataSourceTypeName] <String> [-DataSourceName] <String> [-ComputerName <String> ] [-Credential <PSCredential> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

### Detailed Description

The **Disable-SCDWSource** cmdlet disables all jobs that are affiliated with the specified data source.

### Parameters

#### **-ComputerName<String>**

Specifies the name of the computer on which the System Center Data Access service is running. The user account that is defined in the **Credential** parameter must have access rights to the specified computer. You can omit this parameter only if the System Center Data Access Service is running on the same computer that has Service Manager installed.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

#### **-Credential<PSCredential>**

Specifies the credentials to use when you are connecting to the server on which the System Center Data Access service is running. The provided user account must have access to that server.



Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-DataSourceName<String>**

Specifies the name of the data source to disable. You can use the **Get-SCDWSource** cmdlet to retrieve registered data sources.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-DataSourceTypeName<String>**

Specifies the name of the type of the data source. You can use the **Get-SCDWSourceType** cmdlet to retrieve **TypeNames**.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

---

## -Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## -WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## <CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see [about CommonParameters](#).

## Inputs

The input type is the type of the objects that you can pipe to the cmdlet.

- **None.** You cannot pipe input to this cmdlet.

## Outputs

The output type is the type of the objects that the cmdlet emits.

- **None.** This cmdlet does not generate any output.

---

## Examples

### ----- EXAMPLE 1 -----

This command disables the SM12 data source and all the jobs that use it.

```
PS C:\> Disable-SCDWSource -ComputerName serverDW7 -DataSourceTypeName ServiceManager -
DataSourceName SM12
```

### ----- EXAMPLE 2 -----

The first command stores user credentials for the *Credential* parameter.

The second command disables the data source using the specified credentials.

```
PS C:\> $credUser = Get-Credential
PS C:\> Disable-SCDWSource -ComputerName serverDW7 -DataSourceTypeName ServiceManager -
DataSourceName SM12 -Credential $credUser
```

## Related topics

[Enable-SCDWSource](#)

[Get-SCDWSource](#)

[Set-SCDWSource](#)

[Register-SCDWSource](#)

[Unregister-SCDWSource](#)

---

# Enable-SCDWJob

---

## Enable-SCDWJob

Enables a data warehouse job, which allows it to run according to its specified schedule or to be started manually with the Start-SCDWJob cmdlet.

### Syntax

Parameter Set: Default

```
Enable-SCDWJob [-JobName] <String> [-ComputerName <String> ] [-Credential <PSCredential> ]  
[-Confirm] [-WhatIf] [ <CommonParameters>]
```

### Detailed Description

The **Enable-SCDWJob** cmdlet enables a data warehouse job, which allows it to run according to its specified schedule or to be started manually by using the **Start-SCDWJob** cmdlet. To disable a job, use the **Disable-SCDWJob** cmdlet.

### Parameters

#### -ComputerName<String>

Specifies the name of the computer on which the System Center Data Access service is running. The user account that is defined in the **Credential** parameter must have access rights to the specified computer. You can omit this parameter only if the System Center Data Access Service is running on the same computer that has Service Manager installed.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

---

## **-Credential<PSCredential>**

Specifies the credentials to use when you are connecting to the server on which the System Center Data Access service is running. The provided user account must have access to that server.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-JobName<String>**

Specifies the Data Warehouse job to enable. The **JobName** parameter is a mandatory parameter.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-Confirm**

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

---

## -WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## <CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see [about\\_CommonParameters](#).

## Inputs

The input type is the type of the objects that you can pipe to the cmdlet.

- **None.** You cannot pipe input to this cmdlet.

## Outputs

The output type is the type of the objects that the cmdlet emits.

- **None.** This cmdlet does not generate any output.

## Examples

### ----- EXAMPLE 1 -----

This command enables the Extract\_Contoso job.

```
PS C:\> Enable-SCDWJob -JobName Extract_Contoso -ComputerName serverDW7
```

### ----- EXAMPLE 2 -----

This command lists the jobs sorted by their **IsEnabled** value.

```
PS C:\> Get-SCDWJob -ComputerName serverDW7 | Sort-Object -Property IsEnabled,Name | Format-Table -Property Name,IsEnabled -AutoSize
```

---

### ----- EXAMPLE 3 -----

This command shows only the jobs that are disabled.

```
PS C:\> Get-SCDWJob -ComputerName serverDW7 | Where-Object {$_.IsEnabled -eq $FALSE} | Sort-Object  
-Property IsEnabled,Name | Format-Table -Property Name,IsEnabled -AutoSize
```

### ----- EXAMPLE 4 -----

The first command stores user credentials in a variable for the *Credential* parameter.

The second command enables all the jobs using the **ForEach-Object** cmdlet.

```
PS C:\> $creduser1 = Get-Credential  
PS C:\> Get-SCDWJob -ComputerName serverDW7 -credential $creduser1 | ForEach-Object  
{$_ .EnableJob() }
```

## Related topics

[Disable-SCDWJob](#)

[Get-SCDWJob](#)

[Start-SCDWJob](#)

[Stop-SCDWJob](#)

---

# Enable-SCDWJobCategory

---

## Enable-SCDWJobCategory

Enables all data warehouse jobs within the specified category.

### Syntax

Parameter Set: Default

```
Enable-SCDWJobCategory [-JobCategoryName] <String> [-ComputerName <String> ] [-Credential  
<PSCredential> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

### Detailed Description

The **Enable-SCDWJobCategory** cmdlet enables all data warehouse jobs within the specified category.

### Parameters

#### **-ComputerName<String>**

Specifies the name of the computer on which the System Center Data Access service is running. The user account that is defined in the **Credential** parameter must have access rights to the specified computer. You can omit this parameter only if the System Center Data Access Service is running on the same computer that has Service Manager installed.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

#### **-Credential<PSCredential>**

Specifies the credentials to use when you are connecting to the server on which the System Center Data Access service is running. The provided user account must have access to that server.



Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-JobCategoryName<String>**

Specifies the job category. You can use the **Get-SCDWJob** cmdlet to retrieve job categories. Valid job categories are Maintenance, Extract, Load, and Transform.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-Confirm**

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-WhatIf**

Describes what would happen if you executed the command without actually executing the command.

---

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## <CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see [about\\_CommonParameters](#).

## Inputs

The input type is the type of the objects that you can pipe to the cmdlet.

- **None.** You cannot pipe input to this cmdlet.

## Outputs

The output type is the type of the objects that the cmdlet emits.

- **None.** This cmdlet does not generate any output.

## Examples

### ----- EXAMPLE 1 -----

This command enables the jobs in the CubeProcessing category.

```
PS C:\> Enable-SCDWJobCategory -ComputerName serverDW7 -JobCategoryName CubeProcessing
```

### ----- EXAMPLE 2 -----

This command provides information about enabled and disabled jobs in the job categories. If all the jobs in the category are enabled or disabled, the value in the **IsEnabled** column in the output reflects that value. If at least one job in a job category has an opposite **IsEnabled** value from the other jobs in the same category, the job category name is repeated with that job's opposite **IsEnabled** value (as is the case for the Load category shown in the example output). This is achieved by using the **Unique** parameter for the **Select-Object** command.

```
PS C:\> Get-SCDWJob -ComputerName serverDW7 | Select-Object -Property CategoryName,
IsEnabled -Unique | Format-Table -Property CategoryName, IsEnabled -AutoSize
```

---

CategoryName	IsEnabled
Synchronization	True
CubeProcessing	True
Transform	True
Maintenance	True
Extract	True
Load	True
Load	False

---

# Enable-SCDWJobSchedule

---

## Enable-SCDWJobSchedule

Enables a job schedule so that a job runs according to its specified schedule.

### Syntax

Parameter Set: Default

```
Enable-SCDWJobSchedule [-JobName] <String> [-ComputerName <String> ] [-Confirm] [-WhatIf] [  
<CommonParameters>]
```

### Detailed Description

The **Enable-SCDWJobSchedule** cmdlet enables a job schedule so that a job runs according to its specified schedule. If the job schedule was previously disabled, enabling the job schedule retains the job's schedule settings. To disable the job schedule, use the **Disable-SCDWJobSchedule** cmdlet; to modify the job's schedule, use the **Set-SCDWJobSchedule** cmdlet.

### Parameters

#### **-ComputerName<String>**

Specifies the name of the computer on which the System Center Data Access service is running. You can omit this parameter only if the System Center Data Access Service is running on the same computer that has Service Manager installed.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

---

## **-JobName<String>**

Specifies the Data Warehouse job for which you want to enable the schedule.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-Confirm**

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-WhatIf**

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

---

## <CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see [about\\_CommonParameters](#).

## Inputs

The input type is the type of the objects that you can pipe to the cmdlet.

- **None.** You cannot pipe input to this cmdlet.

## Outputs

The output type is the type of the objects that the cmdlet emits.

- **None.** This cmdlet does not generate any output.

## Examples

### ----- EXAMPLE 1 -----

This command enables the job schedule for the Transform.Common job.

```
PS C:\> Enable-SCDWJobSchedule -ComputerName serverDW7 -JobName Transform.Common
```

### ----- EXAMPLE 2 -----

The following command shows the all the jobs and whether their schedule is enabled or disabled.

```
PS C:\> Get-SCDWJobSchedule -ComputerName serverDW7 | Format-Table -Property Name,
ScheduleEnabled
-AutoSize
```

## Related topics

[Disable-SCDWJobSchedule](#)

[Set-SCDWJobSchedule](#)

[Get-SCDWJobSchedule](#)

[Job Schedule and Frequency \(About Managing the Data Warehouse\)](#)

---

# Enable-SCDWSource

---

## Enable-SCDWSource

Enables all jobs that are associated with the specified data source.

### Syntax

Parameter Set: Default

```
Enable-SCDWSource [-DataSourceTypeName] <String> [-DataSourceName] <String> [-ComputerName <String> ] [-Credential <PSCredential> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

### Detailed Description

The **Enable-SCDWSource** cmdlet enables all jobs that are associated with the specified data source.

### Parameters

#### **-ComputerName<String>**

Specifies the name of the computer on which the System Center Data Access service is running. The user account that is defined in the **Credential** parameter must have access rights to the specified computer. You can omit this parameter only if the System Center Data Access Service is running on the same computer that has Service Manager installed.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

#### **-Credential<PSCredential>**

Specifies the credentials to use when you are connecting to the server on which the System Center Data Access service is running. The provided user account must have access to that server.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-DataSourceName<String>**

Specifies the name of the data source to disable. You can use the **Get-SCDWSource** cmdlet to retrieve registered data sources.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-DataSourceTypeName<String>**

Specifies the type of the data source. You can use the **Get-SCDWSourceType** cmdlet to retrieve **TypeNames**.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false



---

## -Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## -WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## <CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see [about CommonParameters](#).

## Inputs

The input type is the type of the objects that you can pipe to the cmdlet.

- **None.** You cannot pipe input to this cmdlet.

## Outputs

The output type is the type of the objects that the cmdlet emits.

- **None.** This cmdlet does not generate any output.

---

## Examples

### ----- EXAMPLE 1 -----

This first command stores user credentials in the `$userCred` variable for the *Credential* parameter.

The second command enables the SM12 data source, using the stored credentials.

```
PS C:\> $userCred = Get-Credential
```

```
PS C:\> Enable-SCDWSource -ComputerName serverDW7 -DataSourceTypeName ServiceManager -  
DataSourceName SM12 -Credential $userCred
```

## Related topics

[Disable-SCDWSource](#)

[Get-SCDWSource](#)

[Set-SCDWSource](#)

[Register-SCDWSource](#)

[Unregister-SCDWSource](#)

---

# Get-SCDWEntity

---

## Get-SCDWEntity

Gets the list of fact tables, dimensions, tables, and outriggers that exist in a data warehouse.

### Syntax

Parameter Set: DatamartConnectionSet

```
Get-SCDWEntity [-ComputerName <String> ] [-Credential <PSCredential> ] [-  
DatamartComputerName <String> ] [-DatamartDatabaseName <String> ] [ <CommonParameters>]
```

### Detailed Description

The **Get-SCDWEntity** cmdlet gets the list entity names and their types. Entities include act tables, dimensions, tables, and outriggers that exist in the data warehouse.

Entity names are required for setting watermarks, and for setting and getting retention periods; by using the **Set-SCDWWatermark**, **Set-SCDWRetentionPeriod**, and **Get-SCDWRetentionPeriod** cmdlets.

### Parameters

#### **-ComputerName<String>**

Specifies the name of the computer on which the System Center Data Access service is running. The user account that is defined in the **Credential** parameter must have access rights to the specified computer. You can omit this parameter only if the System Center Data Access Service is running on the same computer that has Service Manager installed.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

---

## **-Credential<PSCredential>**

Specifies the credentials to use when you are connecting to the server on which the System Center Data Access service is running. The user account that is provided must have access to that server.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-DatamartComputerName<String>**

Specifies the name of the computer on which the datamart resides. Typically, this is the Structured Query Language (SQL) database server that the data warehouse uses. If no value is provided, Service Manager inspects the configuration of the data warehouse and discovers the value.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-DatamartDatabaseName<String>**

Specifies the name of the database of the datamart. Typically, this is the SQL database server that the data warehouse uses. The default value is the data warehouse repository. Entity values can be retrieved from the **Staging** or from the **DataMart** databases.

Aliases	none
Required?	false
Position?	named
Default Value	none

---

Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## <CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see [about CommonParameters](#).

## Inputs

The input type is the type of the objects that you can pipe to the cmdlet.

- **None.** You cannot pipe input to this cmdlet.

## Outputs

The output type is the type of the objects that the cmdlet emits.

- **Microsoft.EnterpriseManagement.Warehouse.Cmdlets.GetSCDWEntity** A collection of Microsoft.EnterpriseManagement.Warehouse.Cmdlets.GetSCDWEntity objects.

## Examples

### ----- EXAMPLE 1 -----

This command displays the **EntityName** values and their **EntityType**, and displays the output in pages.

```
PS C:\> Get-SCDWEntity -ComputerName serverDW7 | Format-Table -AutoSize | Out-Host -Paging
```

### ----- EXAMPLE 2 -----

The first command stores user credentials for the Credential parameter.

The second command displays the entity data using the specified credentials.

```
PS C:\> $credUser = Get-Credential
```

```
PS C:\> Get-SCDWEntity -ComputerName serverDW7 -Credential $credUser | Format-Table -AutoSize  
| Out-Host -Paging
```

### ----- EXAMPLE 3 -----

This command displays the **EntityName** values sorted by their **EntityType**.

```
PS C:\> Get-SCDWEntity -ComputerName serverDW7 | Sort-Object -Property EntityType,  
EntityName | Format-Table -AutoSize | Out-Host -Paging
```

---

#### ----- EXAMPLE 4 -----

This command displays the **EntityName** values whose **EntityType** value is 'Fact'.

```
PS C:\> Get-SCDWEntity -ComputerName serverDW7 | Where-Object {$_.EntityType -eq "Fact"} |  
Sort-Object -Property EntityType, EntityName | Format-Table -AutoSize
```

#### ----- EXAMPLE 5 -----

This command displays the entities in the OMDWDatamart database.

```
PS C:\> Get-SCDWEntity -ComputerName serverDW7 -DatamartComputerName serverDW7 -  
DatamartDatabaseName OMDWDatamart | Format-Table -AutoSize | Out-Host -Paging
```

#### ----- EXAMPLE 6 -----

This example, and the example that follows, show how to use hash tables to investigate the entity data. This example shows how to create hash tables of the entities according to their database and then how to derive hash tables from them to obtain specific data.

The first, third, and fifth command create hash tables.

The second, fourth, and sixth commands populate the hash tables with the entity names and types in the DWDatamart, OMDWDatamart, and CMDWDatamart databases. The key is the entity name, and the value is the entity type.

```
PS C:\> $DWDatamart = @{}  
PS C:\> Get-SCDWEntity -ComputerName serverDW7 -DatamartComputerName serverDW7 -  
DatamartDatabaseName DWDatamart | ForEach-Object {$DWDatamart.Add($_.EntityName,  
$.EntityType)}  
PS C:\> $OMDWDatamart = @{}  
PS C:\> Get-SCDWEntity -ComputerName serverDW7 -DatamartComputerName serverDW7 -  
DatamartDatabaseName OMDWDatamart | ForEach-Object {$OMDWDatamart.Add($_.EntityName,  
$.EntityType)}  
PS C:\> $CMDWDatamart = @{}  
PS C:\> Get-SCDWEntity -ComputerName serverDW7 -DatamartComputerName serverDW7 -  
DatamartDatabaseName CMDWDatamart | ForEach-Object {$CMDWDatamart.Add($_.EntityName,  
$.EntityType)}
```

#### ----- EXAMPLE 7 -----

This example assumes you still have the \$DWDatamart hash table created in the previous example.

The first command creates a hash table named \$DWDatamart\_Outrigger.

The second command populates the \$DWDatamart\_Outrigger hash table with the Outrigger entities in the DWDatamart database.

---

```
PS C:\> $DWDatamart_Outtrigger = @()
PS C:\> foreach ($key in $DWDatamart.Keys) {
>>   if ($DWDatamart[$key] -eq "Outtrigger") {
>>     $DWDatamart_Outtrigger.Add($key,$DWDatamart[$key])
>>   }
>>}
```

---

# Get-SCDWJob

---

## Get-SCDWJob

Gets the list of all data warehouse jobs, displaying information such as the status of these jobs.

### Syntax

Parameter Set: JobNameSet

```
Get-SCDWJob [-ComputerName <String> ] [-Credential <PSCredential> ] [-JobName <String> ] [-NumberOfBatches <Int32> ] [ <CommonParameters>]
```

Parameter Set: JobIDSet

```
Get-SCDWJob -JobBatchID <Nullable`1> [-ComputerName <String> ] [-Credential <PSCredential> ] [ <CommonParameters>]
```

### Detailed Description

The **Get-SCDWJob** cmdlet gets the list of all data warehouse jobs, displaying information such as the status of these jobs.

### Parameters

#### **-ComputerName<String>**

Specifies the name of the computer on which the System Center Data Access service is running. The user account that is defined in the **Credential** parameter must have access rights to the specified computer. You can omit this parameter only if the System Center Data Access Service is running on the same computer that has Service Manager installed.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false



---

## **-Credential<PSCredential>**

Specifies the credentials to use when you are connecting to the server on which the System Center Data Access service is running. The provided user account must have access to that server.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-JobBatchID<Nullable`1>**

Specifies the batch ID of the job run to return. A job batch ID is generated by the system for each individual run of a job. This integer value is unique across job runs of all types. This parameter cannot be used in conjunction with the **JobName** parameter.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-JobName<String>**

Specifies the name of the Data Warehouse job instance for which you want to get status. This parameter cannot be used in conjunction with the **JobBatchID** parameter.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false

Accept Wildcard Characters?	false
-----------------------------	-------

## **-NumberOfBatches<Int32>**

Specifies the number of batches for which you want to display status. This value must be greater than or equal to 1.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **<CommonParameters>**

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see [about CommonParameters](#).

## **Inputs**

The input type is the type of the objects that you can pipe to the cmdlet.

- **None.** You cannot pipe input to this cmdlet.

## **Outputs**

The output type is the type of the objects that the cmdlet emits.

- **Microsoft.EnterpriseManagement.OrchestrationJob**

## **Examples**

### **----- EXAMPLE 1 -----**

The command in this example lists data warehouse jobs.

```
PS C:\> Get-SCDWJob -ComputerName serverDW7 -NumberOfBatches 4
```

---

## ----- EXAMPLE 2 -----

This command gets Transform.Common jobs.

```
PS C:\> Get-SCDWjob -ComputerName serverDW7 -JobName Transform.Common -NumberOfBatches 2
```

## ----- EXAMPLE 3 -----

This command gets jobs that have a status of 'Not Started' and that have a batch ID greater than 800.

```
PS C:\> Get-SCDWJob -ComputerName ServerDW7 -NumberOfBatches 4 | Where-Object {$_.Status -eq "Not Started" -and $_.BatchID -gt 800}
```

## ----- EXAMPLE 4 -----

This command gets all the job property values and saves the results to a CSV file.

```
PS C:\> Get-SCDWJobModule -ComputerName serverDW7 -NumberOfBatches 4 | Select-Object -Property BatchId, CategoryId, CategoryName, CreationTime, Description, EndTime, Id, IsEnabled, IsRecurring, LastRunTime, ManagementGroup, ModifiedTime, Modules, Name, NextRuntime, Schedule, ScheduleRuleID, SourceTypeName, StartTime, Status, StatusID, WorkflowRuleID | Export-Csv C:\DWData\JobData.csv
```

## Related topics

[Enable-SCDWJob](#)

[Get-SCDWJobModule](#)

[Start-SCDWJob](#)

---

# Get-SCDWJobModule

---

## Get-SCDWJobModule

Returns detailed status information about the modules of the specified job.

### Syntax

Parameter Set: JobNameSet

```
Get-SCDWJobModule [-ComputerName <String> ] [-JobName <String> ] [-NumberOfBatches <Int32> ]  
[ <CommonParameters>]
```

Parameter Set: JobIDSet

```
Get-SCDWJobModule -JobBatchID <Nullable`1> [-ComputerName <String> ] [-ModuleID <Nullable`1> ]  
[ -ModuleName <String> ] [ <CommonParameters>]
```

Parameter Set: JobNameModuleIDSet

```
Get-SCDWJobModule [-ComputerName <String> ] [-JobName <String> ] [-ModuleID <Nullable`1> ]  
[-NumberOfBatches <Int32> ] [ <CommonParameters>]
```

Parameter Set: JobNameModuleNameSet

```
Get-SCDWJobModule [-ComputerName <String> ] [-JobName <String> ] [-ModuleName <String> ] [-  
NumberOfBatches <Int32> ] [ <CommonParameters>]
```

### Detailed Description

The **Get-SCDWJobModule** cmdlet returns detailed status information about the modules of the specified job. Each data warehouse job comprises multiple subprocesses, called *modules*, which perform the individual work items that are associated with the parent job.

For updated information about this cmdlet, see [Get-SCDWJobModule](#).

### Parameters

#### **-ComputerName<String>**

Specifies the name of the computer on which the System Center Data Access service is running. You can omit this parameter only if the System Center Data Access Service is running on the same computer that has Service Manager installed.

Aliases	none
---------	------

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-JobBatchID<Nullable`1>**

Specifies the job's batch ID for which to retrieve all associated modules. You can use the **Get-SCDWJob** cmdlet to retrieve a job's batch ID. By specifying the **JobBatchID** parameter, you retrieve all the modules of the specified job.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-JobName<String>**

Specifies the name of the specific Extract, Transform, or Load job for which you want to get status.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-ModuleID<Nullable`1>**

Specifies the ID of the job module for which to retrieve status. This must be a valid module ID for the specified job. This parameter cannot be used in conjunction with the **JobModule** parameter.

---

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-ModuleName<String>**

Specifies the name of the job module for which to retrieve status. This name must be a valid module name for the specified job. This parameter cannot be used in conjunction with the **ModuleID** parameter.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-NumberOfBatches<Int32>**

Specifies the number of batches for which to return status.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

---

## <CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see [about\\_CommonParameters](#).

## Inputs

The input type is the type of the objects that you can pipe to the cmdlet.

- **None.** You cannot pipe input to this cmdlet.

## Outputs

The output type is the type of the objects that the cmdlet emits.

- **Microsoft.EnterpriseManagement.OrchestrationJobModule**

## Examples

### ----- EXAMPLE 1 -----

This command displays module information about the Transform.common job.

```
PS C:\>Get-SCDWJobmodule -ComputerName serverDW7 -JobName Transform.common -NumberOfBatches 4
```

### ----- EXAMPLE 2 -----

This command displays module information for jobs using a specified job batch ID.

```
PS C:\>Get-SCDWJobModule -ComputerName serverDW7 -JobBatchID 4126 -NumberOfBatches 4
```

### ----- EXAMPLE 3 -----

This command displays module information for a specific job and module by its module ID.

```
PS C:\> Get-SCDWJobModule -ComputerName IXSMTTestDW -JobName DWMaintenance -ModuleID 7 -NumberOfBatches 12
```

### ----- EXAMPLE 4 -----

This command displays module information for a specific job and module by its module name.

```
PS C:\> Get-SCDWJobModule -ComputerName IXSMTTestDW -JobName MPSyncJob -ModuleName "Clean Up" -NumberOfBatches 12
```

---

## ----- EXAMPLE 5 -----

This command lists module error counts, but only if an error count is greater than zero.

```
PS C:\> Get-SCDWJobModule -ComputerName serverDW7 -NumberOfBatches 12 | Where-Object  
{$_ .ModuleErrorCount -gt 0} | Format-Table -Property JobName, ModuleID, ModuleErrorCount -  
AutoSize
```

## ----- EXAMPLE 6 -----

This command gets all the job module property values and saves the results to a CSV file.

```
PS C:\> Get-SCDWJobModule -ComputerName serverDW7 -NumberOfBatches 4 | Select-Object -  
Property BatchId, CategoryId, CategoryName, CreationTime, Description, JobId, JobModuleId,  
JobName, ManagementGroup, ManagementGroupId, ModuleCreationTime, ModuleDescription,  
ModuleErrorCount, ModuleErrorSummary, ModuleId, ModuleModifiedTime, ModuleName,  
ModuleRetryCount, ModuleStartTime, ModuleTypeId, ModuleTypeName, NotToBePickedBefore, Status  
| Export-Csv C:\TEMP\JobModuleData.csv
```

## Related topics

[Get-SCDWJob](#)



---

# Get-SCDWJobSchedule

---

## Get-SCDWJobSchedule

Gets scheduling information for data warehouse jobs.

### Syntax

Parameter Set: Default

```
Get-SCDWJobSchedule [-ComputerName <String> ] [-JobName <String> ] [ <CommonParameters>]
```

### Detailed Description

The **Get-SCDWJobSchedule** cmdlet provides scheduling information for data warehouse jobs. You can either use the **JobName** parameter to specify a specific job or omit that parameter to display scheduling information for all data warehouse jobs.

### Parameters

#### **-ComputerName<String>**

Specifies the name of the computer on which the System Center Data Access service is running. You can omit this parameter only if the System Center Data Access Service is running on the same computer that has Service Manager installed.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

#### **-JobName<String>**

Specifies the job name for which to display scheduling information.

---

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## <CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see [about\\_CommonParameters](#).

## Inputs

The input type is the type of the objects that you can pipe to the cmdlet.

- **None.** You cannot pipe input to this cmdlet.

## Outputs

The output type is the type of the objects that the cmdlet emits.

- **Microsoft.EnterpriseManagement.OrchestrationJob**

## Examples

### ----- EXAMPLE 1 -----

This command displays the job schedule for the Transform.Common job.

```
PS C:\> Get-SCDWJobSchedule -ComputerName serverDW7 -JobName Transform.Common
```

### ----- EXAMPLE 2 -----

This command displays all jobs and their run times.

```
PS C:\> Get-SCDWJobSchedule -ComputerName serverDW7 | Format-Table -Property Name, LastRunTime, NextRunTime -AutoSize
```

## Related topics

[Enable-SCDWJobSchedule](#)

---

[Disable-SCDWJobSchedule](#)

[Set-SCDWJobSchedule](#)

---

# Get-SCDWRetentionPeriod

---

## Get-SCDWRetentionPeriod

Gets the data retention period, in minutes, for either a specific fact table within a specific data warehouse database or the default for fact tables within the database.

### Syntax

Parameter Set: EntitySet

```
Get-SCDWRetentionPeriod [-ComputerName <String> ] [-Credential <PSCredential> ] [-EntityName <String> ] [ <CommonParameters>]
```

Parameter Set: DatamartConnectionSet

```
Get-SCDWRetentionPeriod [-DatamartComputerName] <String> [-DatamartDatabaseName] <String> [-ComputerName <String> ] [-Credential <PSCredential> ] [-EntityName <String> ] [ <CommonParameters>]
```

### Detailed Description

The **Get-SCDWRetentionPeriod** cmdlet gets the data retention period, in minutes, for either a specific fact table within a specific data warehouse database or for the default fact table within the database.

The Data Warehouse grooms out rows after a predefined retention period. Data that is eligible for grooming and that is older than the retention period will be groomed out of the database.

### Parameters

#### **-ComputerName<String>**

Specifies the name of the computer on which the System Center Data Access service is running. The user account that is defined in the **Credential** parameter must have access rights to the specified computer. You can omit this parameter only if the System Center Data Access Service is running on the same computer that has Service Manager installed.

Aliases	none
Required?	false
Position?	named
Default Value	none

Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-Credential<PSCredential>**

Specifies the credentials to use when you are connecting to the server on which the System Center Data Access service is running. The user account that is provided must have access to that server.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-DatamartComputerName<String>**

Specifies the name of the computer on which the datamart resides. This is usually the SQL database server that the data warehouse uses. By default, Service Manager discovers the value by inspecting the data warehouse configuration.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-DatamartDatabaseName<String>**

Specifies the name of the database of the datamart. This is usually the SQL database server that the data warehouse uses. The default value is the data warehouse repository. It is possible to retrieve the entities from the Staging or DataMart databases.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-EntityName<String>**

Specifies the Dimension, Fact, or Outrigger on which to set the watermark. You can use the **Get-SCDWEntity** cmdlet to retrieve the entities.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **<CommonParameters>**

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see [about\\_CommonParameters](#).

## **Inputs**

The input type is the type of the objects that you can pipe to the cmdlet.

- **None.** You cannot pipe input to this cmdlet.

## **Outputs**

The output type is the type of the objects that the cmdlet emits.

- **None.** This cmdlet does not generate any output.

---

## Examples

### ----- EXAMPLE 1 -----

This command gets the data retention period for a specified entity name.

```
PS C:\> Get-SCDWRetentionPeriod -ComputerName serverDW7 -DatamartComputerName serverDW7 -  
DatamartDatabaseName DWDataMart -EntityName IncidentStatusDurationFact
```

### ----- EXAMPLE 2 -----

The first command creates a variable for a hash table.

The second, third, and fourth commands populate the hash table with the retention period for each entity in three databases (DWDatamart, CMDWDatamart, and OMDWDatamart). The key for the hash table is the entity name with its database name in parenthesis, and the value is the retention period.

```
PS C:\> $AllEntitiesRP = @{}  
PS C:\> Get-SCDWEntity -ComputerName serverDW7 -DatamartComputerName serverDW7 -  
DatamartDatabaseName DWDatamart | ForEach-Object {  
  >> $ent = $_.EntityName  
  >> $rp = Get-SCDWRetentionPeriod -ComputerName serverDW7 -DatamartComputerName  
serverDW7 -DatamartDatabaseName DWDatamart -Entity $ent  
  >> $AllEntitiesRP.Add($ent + " (DWDatamart)", $rp.RetentionPeriodInMinutes)  
  >> }  
>>  
PS C:\> Get-SCDWEntity -ComputerName serverDW7 -DatamartComputerName serverDW7 -  
DatamartDatabaseName CMDWDatamart | ForEach-Object {  
  >> $ent = $_.EntityName  
  >> $rp = Get-SCDWRetentionPeriod -ComputerName serverDW7 -DatamartComputerName  
serverDW7 -DatamartDatabaseName CMDWDatamart -Entity $ent  
  >> $AllEntitiesRP.Add($ent + " (CMDWDatamart)", $rp.RetentionPeriodInMinutes)  
  >> }  
>>  
PS C:\> Get-SCDWEntity -ComputerName serverDW7 -DatamartComputerName serverDW7 -  
DatamartDatabaseName OMDWDatamart | ForEach-Object {  
  >> $ent = $_.EntityName  
  >> $rp = Get-SCDWRetentionPeriod -ComputerName serverDW7 -DatamartComputerName  
serverDW7 -DatamartDatabaseName OMDWDatamart -Entity $ent  
  >> $AllEntitiesRP.Add($ent + " (OMDWDatamart)", $rp.RetentionPeriodInMinutes)  
  }  
}
```

---

# Get-SCDWSource

---

## Get-SCDWSource

Gets specific instances of data sources which are registered to the data warehouse.

### Syntax

Parameter Set: Default

```
Get-SCDWSource [[-DataSourceTypeName] <String> ] [-ComputerName <String> ] [-Credential <PSCredential> ] [ <CommonParameters>]
```

### Detailed Description

The **Get-SCDWSource** cmdlet gets specific instances of data sources that are registered to the data warehouse.

For updated information about this cmdlet, see [Get-SCDWSource](#).

### Parameters

#### **-ComputerName<String>**

Specifies the name of the computer on which the System Center Data Access service is running. The user account that is defined in the **Credential** parameter must have access rights to the specified computer. You can omit this parameter only if the System Center Data Access Service is running on the same computer that has Service Manager installed.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false



---

## **-Credential<PSCredential>**

Specifies the credentials to use when you are connecting to the server on which the System Center Data Access service is running. The user account that is provided must have access to that server.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-DataSourceTypeName<String>**

Specifies the type of the data source. You can use the **Get-SCDWSourceType** cmdlet to retrieve **TypeNames**.

Aliases	none
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **<CommonParameters>**

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see [about\\_CommonParameters](#).

## **Inputs**

The input type is the type of the objects that you can pipe to the cmdlet.

- **None.** You cannot pipe input to this cmdlet.

## **Outputs**

The output type is the type of the objects that the cmdlet emits.

- **Microsoft.EnterpriseManagement.DataWarehouse.DataSource**

---

## Examples

### ----- EXAMPLE 1 -----

This command retrieves all data sources from the serverDW7 computer.

```
PS C:\> Get-SCDWSource -ComputerName serverDW7
```

### ----- EXAMPLE 2 -----

This command retrieves all data sources from the serverDW7 computer, for which the data source type is ServiceManager.

```
PS C:\> Get-SCDWSource -DataSourceTypeName ServiceManager -ComputerName serverDW7
```

### ----- EXAMPLE 3 -----

This command finds any data sources that were registered before a specified date.

```
PS C:\> Get-SCDWSource -ComputerName serverDW7 | Where-Object {$_.DateRegistered -lt [System.DateTime]"June 12 2012"}
```

### ----- EXAMPLE 4 -----

This command finds any data sources that have been unregistered.

```
PS C:\> Get-SCDWSource -ComputerName serverDW7 | Where-Object {$_.DateUnRegistered -ne $Null}
```

## Related topics

[Enable-SCDWSource](#)

[Disable-SCDWSource](#)

[Set-SCDWSource](#)

[Register-SCDWSource](#)

[Unregister-SCDWSource](#)

---

# Get-SCDWSourceType

---

## Get-SCDWSourceType

Gets the types of data sources that can be registered to the data warehouse.

### Syntax

Parameter Set: Default

```
Get-SCDWSourceType [-ComputerName <String> ] [-Credential <PSCredential> ] [
<CommonParameters>]
```

### Detailed Description

The **Get-SCDWSourceType** cmdlet gets the types of data sources that can be registered to the data warehouse.

For updated information about this cmdlet, see [Get-SCDWSourceType](#).

### Parameters

#### **-ComputerName<String>**

Specifies the name of the computer on which the System Center Data Access service is running. The user account that is defined in the **Credential** parameter must have access rights to the specified computer. You can omit this parameter only if the System Center Data Access Service is running on the same computer that has Service Manager installed.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

---

## **-Credential<PSCredential>**

Specifies the credentials to use when you are connecting to the server on which the System Center Data Access service is running. The provided user account must have access to that server.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **<CommonParameters>**

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see [about\\_CommonParameters](#).

## **Inputs**

The input type is the type of the objects that you can pipe to the cmdlet.

- **None.** You cannot pipe input to this cmdlet.

## **Outputs**

The output type is the type of the objects that the cmdlet emits.

- **Microsoft.EnterpriseManagement.Warehouse.Cmdlets.GetSCDWSourceType+DataSourceType**

## **Examples**

### **----- EXAMPLE 1 -----**

This command gets the available source types.

```
PS C:\> Get-SCDWSourceType -ComputerName serverDW7
```

---

# Get-SCDWWatermark

---

## Get-SCDWWatermark

Gets the latest watermark for the specified job module.

### Syntax

Parameter Set: Default

```
Get-SCDWWatermark [-ModuleType] <String> [-ModuleName] <String> [-ComputerName <String> ] [-Credential <PSCredential> ] [ <CommonParameters>]
```

### Detailed Description

The **Get-SCDWWatermark** cmdlet gets the stage of completion of the specified module, along with any other modules that the current module depends on.

Modules depend on each other, and every job module processes data incrementally. This cmdlet provides a complete picture of data latency.

### Parameters

#### **-ComputerName<String>**

Specifies the name of the computer on which the System Center Data Access service is running. The user account that is defined in the **Credential** parameter must have access rights to the specified computer. You can omit this parameter only if the System Center Data Access Service is running on the same computer that has Service Manager installed.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

---

## **-Credential<PSCredential>**

Specifies the credentials to use when you are connecting to the server on which the System Center Data Access service is running. The provided user account must have access to that server.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-ModuleName<String>**

Specifies the name of the job module for which to retrieve the watermark. You can use the **Get-SCDWJobModule** cmdlet to retrieve the job module. Only modules that start with either Transform or Load will be retrieved.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-ModuleType<String>**

Specifies the type of job module for which to retrieve the watermark. Valid strings are Transform and Load.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false

---

Accept Wildcard Characters?
-----------------------------

false
-------

## <CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see [about\\_CommonParameters](#).

## Inputs

The input type is the type of the objects that you can pipe to the cmdlet.

- **None.** You cannot pipe input to this cmdlet.

## Outputs

The output type is the type of the objects that the cmdlet emits.

- **Microsoft.EnterpriseManagement.Warehouse.Cmdlets.GetSCDWatermark**

## Examples

### ----- EXAMPLE 1 -----

This command gets the watermarks for a specified module and module name.

```
PS C:\> Get-SCDWatermark -ComputerName serverDW7 -ModuleType Transform -Module  
Name TransformSoftwareUpdateDim
```

---

# New-SCDWSourceType

---

## New-SCDWSourceType

Creates a new data source type can be registered to the data warehouse.

### Syntax

Parameter Set: Default

```
New-SCDWSourceType [-SourceConfigFile] <String> [-ComputerName <String> ] [-Credential <PSCredential> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

### Detailed Description

The **New-SCDWSourceType** cmdlet creates a new data source type that can be registered to the data warehouse. Each data source type is defined by the classes and the relationships in the management pack bundle that is imported when the data source type is defined.

### Parameters

#### **-ComputerName<String>**

Specifies the name of the computer on which the System Center Data Access service is running. The user account that is defined in the *Credential* parameter must have access rights to the specified computer.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false



---

## -Credential<PSCredential>

Specifies the credentials to use when you are connecting to the server on which the System Center Data Access service is running. The user account that is provided must have access to that server.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## -SourceConfigFile<String>

Specifies the path to the management pack bundle (.mpb) file that contains the management packs and other resources that define the data source.

Each management pack in the management pack bundle must contain a class that derives from the **Microsoft.SystemCenter.DataWarehouse.DataSource** class, and an enumeration that derives from **DW.DataSourceType**. The base class and the enumeration must exist in the **Microsoft.SystemCenter.DataWarehouse.Base.mp** management pack.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByPropertyName)
Accept Wildcard Characters?	false

## -Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none

Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## -WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## <CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see [about\\_CommonParameters](#).

## Inputs

The input type is the type of the objects that you can pipe to the cmdlet.

- **None.** You cannot pipe input to this cmdlet.

## Outputs

The output type is the type of the objects that the cmdlet emits.

- **None.** This cmdlet does not generate any output.

## Examples

### ----- EXAMPLE 1 -----

This command creates a new data source type from a specified management pack bundle.

```
PS C:\> New-SCDWSourceTypes -ComputerName serverDW7 -SourceConfigFile "C:\Program Files\Microsoft System Center\Management Packs\CustomHumanRelationsDataSource.mpb"
```

---

# Register-SCDWSource

---

## Register-SCDWSource

Registers instances of data source types, such as Service Manager, Operations Manager, and Configuration Manager, to the data warehouse.

### Syntax

Parameter Set: Default

```
Register-SCDWSource [-DataSourceTypeName] <String> [-SourceComputerName] <String> [-AdditionalData <Object> ] [-ComputerName <String> ] [-Credential <PSCredential> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

### Detailed Description

The **Register-SCDWSource** cmdlet registers instances of data source types, such as Service Manager, Operations Manager, and Configuration Manager, to the data warehouse.

### Parameters

#### **-AdditionalData<Object>**

Specifies any additional data that a particular source might have to send during registration. For example, during a Configuration Manager source registration, this parameter provides the necessary credentials in the form of a credential object, which the Configuration Manager source uses to contact the Configuration Manager server.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

---

## **-ComputerName<String>**

Specifies the name of the data warehouse management server on which to run the **Register-SCDWSource** cmdlet. The user account specified in the *Credential* parameter must have access rights to the specified computer. You can omit this parameter only if the System Center Data Access Service is running on the same computer that has Service Manager installed.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-Credential<PSCredential>**

Specifies the credentials to use when you are connecting to the server on which the System Center Data Access service is running. The provided user account must have access to that server.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-DataSourceTypeName<String>**

Specifies the type of the data source. You can use the **Get-SCDWSourceType** cmdlet to retrieve **TypeNames**.

Aliases	none
Required?	true
Position?	1
Default Value	none

---

Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-SourceComputerName<String>**

Specifies the name of the Service Manager management server that is to be registered with the data warehouse.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-Confirm**

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-WhatIf**

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false

---

Accept Wildcard Characters?
-----------------------------

false
-------

## <CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see [about\\_CommonParameters](#).

## Inputs

The input type is the type of the objects that you can pipe to the cmdlet.

- **None.** You cannot pipe input to this cmdlet.

## Outputs

The output type is the type of the objects that the cmdlet emits.

- **None.** This cmdlet does not generate any output.

## Examples

### ----- EXAMPLE 1 -----

The first command prompts for your credentials and stores them in an object variable.

The second command registers a Service Manager management group to the data warehouse.

```
PS C:\> $CredUser1 = Get-Credential
```

```
PS C:\> Register-SCDWSource -ComputerName serverDW7 -DataSourceTypeName ServiceManager -  
SourceComputerName serverDW7 -Credential $CredUser1
```

### ----- EXAMPLE 2 -----

The first command prompts for your credentials and stores them in an object variable.

The second command registers an Operations Manager management group with the data warehouse.

```
PS C:\> $CredUser1 = Get-Credential
```

```
PS C:\> Register-SCDWSource -ComputerName serverDW7 -DataSourceTypeName OperationsManager -  
SourceComputerName om2012 -DataSourceDbName OperationsManager -DataSourceDbServerName om2012  
-FullPathToSourceManagementPackBundle "C:\Program Files\Microsoft System Center\Service  
Manager 2012\OperationsManagerMP.mpb" -Credential $CredUser1
```

### ----- EXAMPLE 3 -----

This command registers a Configuration Manager server with the data warehouse.

```
PS C:\> Register-SCDWSource -ComputerName serverDW7 -DataSourceType
```

```
ConfigurationManager.DataSource -DataSourceName serverDW7_sms_smq-SCCMDataSource -Properties
```

---

```
@{ ServerName = 'serverDW7'; DatabaseName = 'sms_smq'; SubjectList =  
'Microsoft.EnterpriseManagement.ServiceManager.Connector.Sms,Microsoft.SystemCenter.ConfigurationManager.Connector.PowerManagement'; SecureReferenceId = 'bb89ca43-3e64-07fc-a136-592183066184'; Version='bd3bc1fe-1546-5108-2faf-4d5c01d2bc6a' } -  
FullPathToSourceManagementPackBundle "C:\Program Files\Microsoft System Center\Service  
Manager 2012\SCCMDW.mpb" -AdditionalData @($input)[0]
```

## Related topics

[Enable-SCDWSource](#)

[Disable-SCDWSource](#)

[Get-SCDWSource](#)

[Set-SCDWSource](#)

[Unregister-SCDWSource](#)

---

# Set-SCDWJobSchedule

---

## Set-SCDWJobSchedule

Sets the schedule for a data warehouse job.

### Syntax

Parameter Set: Default

```
Set-SCDWJobSchedule [-JobName] <String> [-ComputerName <String> ] [-DailyFrequency <Nullable`1> ] [-DailyStart <Nullable`1> ] [-ScheduleType <ScheduleTypeEnum> ] [-WeeklyFrequency <List`1> ] [-WeeklyStart <Nullable`1> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

### Detailed Description

The **Set-SCDWJobSchedule** cmdlet configures the schedule for data warehouse jobs. You can configure the schedule on a daily basis or on a weekly basis.

For a daily schedule, you must specify the time interval at which the job recurs (**DailyFrequency**) and the time of day at which this recurrence starts (**DailyStart**).

For a weekly schedule, you must specify the time of day (**WeeklyStart**) at which the job should run and the list of days for which this start time is effective (**WeeklyFrequency**).

### Parameters

#### **-ComputerName<String>**

Specifies the name of the computer on which the System Center Data Access service is running. You can omit this parameter only if the System Center Data Access Service is running on the same computer that has Service Manager installed.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false



---

## **-DailyFrequency<Nullable`1>**

Specifies the recurrence frequency of the scheduled job for a daily schedule.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-DailyStart<Nullable`1>**

Specifies the time at which the scheduled job will start for a daily schedule. Specify the start time in the hh:mm format.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-JobName<String>**

Specifies the Data Warehouse job for which the schedule is being configured.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false

Accept Wildcard Characters?	false
-----------------------------	-------

### **-ScheduleType<ScheduleTypeEnum>**

Specifies the schedule as Daily or Weekly. The default value is Daily.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-WeeklyFrequency<List`1>**

Specifies the days on which the job will run for a weekly schedule. The job will start according to the time that is specified by the **WeeklyStart** parameter. Use a comma to separate multiple days (for example: Tuesday, Thursday).

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-WeeklyStart<Nullable`1>**

Specifies the time of day at which the job will start for a weekly schedule (for example: 8:00, 16:00). The job will begin at this time on the days that are specified by the **WeeklyFrequency** parameter. Specify the start time in the hh:mm format.

Aliases	none
Required?	false

Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-Confirm**

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-WhatIf**

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **<CommonParameters>**

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see [about\\_CommonParameters](#).

## **Inputs**

The input type is the type of the objects that you can pipe to the cmdlet.

- **None.** You cannot pipe input to this cmdlet.

---

## Outputs

The output type is the type of the objects that the cmdlet emits.

- **None.** This cmdlet does not generate any output.

## Notes

- Valid schedule parameter combinations are as follows:

**-DailyFrequency**HH:MM**-DailyStart**HH:MM**-ScheduleType Daily**

**-WeeklyFrequency**Day(s)**-WeeklyStart**HH:MM, **-ScheduleType Weekly**

For the weekly frequency, specify the day of the week such as Monday. For multiple days, separate with commas such as Monday, Tuesday, Wednesday. You cannot configure a job with both a weekly schedule and a daily schedule.

## Examples

### ----- EXAMPLE 1 -----

This command sets the schedule for the Transform.Common job, to run every 30 minutes starting at midnight.

```
PS C:\> Set-SCDWJobSchedule -ComputerName serverDW7 -JobName Transform.Common -  
DailyFrequency 00:30 -DailyStart 00:00
```

## Related topics

[Disable-SCDWJobSchedule](#)

[Enable-SCDWJobSchedule](#)

[Get-SCDWJobSchedule](#)

---

# Set-SCDWRetentionPeriod

---

## Set-SCDWRetentionPeriod

Sets the data retention period in minutes for either a specific fact table within a specific data warehouse database, or sets the default for fact tables within the database. Data that is eligible for grooming and older than the retention period will be groomed out of the database.

### Syntax

Parameter Set: EntitySet

```
Set-SCDWRetentionPeriod [-ComputerName <String> ] [-Credential <PSCredential> ] [-DurationInMinutes <Int32> ] [-EntityName <String> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Parameter Set: DatamartConnectionSet

```
Set-SCDWRetentionPeriod [-DatamartComputerName] <String> [-DatamartDatabaseName] <String> [-ComputerName <String> ] [-Credential <PSCredential> ] [-DurationInMinutes <Int32> ] [-EntityName <String> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

### Detailed Description

The **Set-SCDWRetentionPeriod** cmdlet sets the data retention period, in minutes, for either a specific fact table within a specific data warehouse database or the default for fact tables within the database. Data that is eligible for grooming and older than the retention period will be groomed out of the database.

### Parameters

#### **-ComputerName<String>**

Specifies the name of the computer on which the System Center Data Access service is running. The user account that is defined in the **Credential** parameter must have access rights to the specified computer. You can omit this parameter only if the System Center Data Access Service is running on the same computer that has Service Manager installed.

Aliases	none
Required?	false
Position?	named

Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-Credential<PSCredential>**

Specifies the credentials to use when you are connecting to the server on which the System Center Data Access service is running. The user account that is provided must have access to that server.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-DatamartComputerName<String>**

Specifies the name of the computer on which the datamart resides. This is usually the SQL database server that services the data warehouse. If a value is not provided, the value is discovered by inspecting the warehouse configuration.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-DatamartDatabaseName<String>**

Specifies the name of the datamart database. This is typically the SQL database server that services the data warehouse. The default value is the warehouse repository. You can retrieve the entities from the Staging or DataMart databases to provide a value for this parameter.

Aliases	none
Required?	true
Position?	3
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-DurationInMinutes<Int32>**

Specifies the duration, in minutes, before the entity is groomed.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-EntityName<String>**

Specifies the Dimension, Fact, or Outrigger on which to set the watermark. You can use the **Get-SCDWEntity** cmdlet to retrieve entity names.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-Confirm**

Prompts you for confirmation before executing the command.

---

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-WhatIf**

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **<CommonParameters>**

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see [about\\_CommonParameters](#).

## **Inputs**

The input type is the type of the objects that you can pipe to the cmdlet.

- **None.** You cannot pipe input to this cmdlet.

## **Outputs**

The output type is the type of the objects that the cmdlet emits.

- **None.** This cmdlet does not generate any output.

## **Examples**

### **----- EXAMPLE 1 -----**

This example sets a retention period.



---

```
PS C:\> Set-SCDWRetentionPeriod -ComputerName serverDW7 -DatamartComputerName serverDW7 -  
DatamartDatabaseName CMDWDataMart -EntityName  
ComputerHasSoftwareUpdateInstalledFact -DurationInMinutes 1576800
```

---

# Set-SCDWSource

---

## Set-SCDWSource

Updates the definition of classes and relationships that can be populated for an instance of a data source.

### Syntax

Parameter Set: Default

```
Set-SCDWSource [-DataSourceTypeName] <String> [-DataSourceName] <String> [-AdditionalData <Object> ] [-ComputerName <String> ] [-Credential <PSCredential> ] [-FullPathToSourceManagementPackBundle <String> ] [-Properties <Hashtable> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

### Detailed Description

The **Set-SCDWSource** cmdlet updates the definition of classes and relationships that can be populated for an instance of a data source.

### Parameters

#### **-AdditionalData<Object>**

Specifies any additional data that a particular source has to send during registration. Specifically, this is used during a **cmshort** source registration, to provide credentials in the form of a credential object that the **cmshort** source uses to contact the **cmshort** server.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

---

## **-ComputerName<String>**

Specifies the name of the computer on which the System Center Data Access service is running. The user account that is defined in the *Credential* parameter must have access rights to the specified computer. You can omit this parameter only if the System Center Data Access Service is running on the same computer that has Service Manager installed.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-Credential<PSCredential>**

Specifies the credentials to use when you are connecting to the server on which the System Center Data Access service is running. The user account that is provided must have access to that server.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-DataSourceName<String>**

Specifies the name of the data source to use. You can use the **Get-SCDWSource** cmdlet to retrieve names of registered data sources.

Aliases	none
Required?	true
Position?	2
Default Value	none

Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-DataSourceTypeName<String>**

Specifies the type of the data source. You can use the **Get-SCDWSourceType** cmdlet to retrieve data source type names.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-FullPathToSourceManagementPackBundle<String>**

Specifies the path to the management pack bundle that contains the management packs with the definitions for the new source types. This makes it possible for you to define the type system for the new source in the same manner that type systems are defined in management packs by declaring classes.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-Properties<Hashtable>**

Provides further arguments that will be used when a data source is set up. The hashtable contents vary according to the data source type.

---

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-Confirm**

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-WhatIf**

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **<CommonParameters>**

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see [about\\_CommonParameters](#).

## **Inputs**

The input type is the type of the objects that you can pipe to the cmdlet.

- 
- **None.** You cannot pipe input to this cmdlet.

## Outputs

The output type is the type of the objects that the cmdlet emits.

- **None.** This cmdlet does not generate any output.

## Examples

### ----- EXAMPLE 1 -----

The first command prompts for user credentials and stores them in a variable.

The second command updates the version and the synctype properties for a Configuration Manager data source.

```
PS C:\> $cred = Get-Credential
PS C:\> Set-SCDWSource -DataSourceType ConfigurationManager.DataSource -DataSourceName
MSITData -Properties @{ Version="D0925349-F1FC-1084-0761-EE60D21B1141"; SyncType="9BFF3B4B-
80D0-A263-2DAD-90A67C5FCA39"} -AdditionalData $cred -FullPathToSourceManagementPackBundle
"C:\NewCMSourceMPS.mpb" -ComputerName serverDW7
```

## Related topics

[Enable-SCDWSource](#)

[Disable-SCDWSource](#)

[Get-SCDWSource](#)

[Register-SCDWSource](#)

[Unregister-SCDWSource](#)

---

# Set-SCDWWatermark

---

## Set-SCDWWatermark

Sets the watermark from which subsequent data processing should continue.

### Syntax

Parameter Set: Default

```
Set-SCDWWatermark [-EntityName] <String> [-WaterMarkValue] <DateTime> [-ComputerName  
<String> ] [-Credential <PSCredential> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

### Detailed Description

The **Set-SCDWWatermark** cmdlet sets the watermark from which data processing should continue. If you need to extract and process data again, a watermark identifies the point from which to continue the data processing so that data loss is minimal.

### Parameters

#### **-ComputerName<String>**

Specifies the name of the computer on which the System Center Data Access service is running. The user account that is defined in the *Credential* parameter must have access rights to the specified computer. You can omit this parameter only if the System Center Data Access Service is running on the same computer that has Service Manager installed.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

---

## **-Credential<PSCredential>**

Specifies the credentials to use when you are connecting to the server on which the System Center Data Access service is running. The user account that is provided must have access to that server.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-EntityName<String>**

Specifies the Dimension, Fact, or Outrigger on which to set the watermark. You can use the **Get-SCDWEntity** cmdlet to retrieve the value for the **EntityName** parameter.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-WaterMarkValue<DateTime>**

Specifies the date that the watermark should be set to. The specified date must be in the past.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false



---

## -Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## -WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## <CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see [about\\_CommonParameters](#).

## Inputs

The input type is the type of the objects that you can pipe to the cmdlet.

- **None.** You cannot pipe input to this cmdlet.

## Outputs

The output type is the type of the objects that the cmdlet emits.

- **None.** This cmdlet does not generate any output.

---

## Examples

### ----- EXAMPLE 1 -----

The command in this example sets a watermark.

```
PS C:\>Set-SCDWWatermark -ComputerName serverDW7 -EntityName SoftwareUpdateDim  
-WaterMarkValue 1/1/2010
```

---

# Start-SCDWJob

---

## Start-SCDWJob

Starts a data warehouse job.

### Syntax

Parameter Set: Default

```
Start-SCDWJob -JobName <String> [-ComputerName <String> ] [-Credential <PSCredential> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

### Detailed Description

The **Start-SCDWJob** cmdlet starts a data warehouse job. When a job is started, its associated job modules run in the order that the job sets. To stop a job, use the **Stop-SCDWJob** cmdlet.

### Parameters

#### **-ComputerName<String>**

Specifies the name of the computer on which the System Center Data Access service is running. The user account that is defined in the *Credential* parameter must have access rights to the specified computer. You can omit this parameter only if the System Center Data Access Service is running on the same computer that has Service Manager installed.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

---

## **-Credential<PSCredential>**

Specifies the credentials to use when you are connecting to the server on which the System Center Data Access service is running. The user account that is provided must have access to that server.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-JobName<String>**

Specifies the job to be started. The **JobName** parameter is mandatory.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-Confirm**

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

---

## -WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## <CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see [about\\_CommonParameters](#).

## Inputs

The input type is the type of the objects that you can pipe to the cmdlet.

- **None.** You cannot pipe input to this cmdlet.

## Outputs

The output type is the type of the objects that the cmdlet emits.

- **None.** This cmdlet does not generate any output.

## Examples

### ----- EXAMPLE 1 -----

This command starts the Extract\_Contoso job.

```
PS C:\>Start-SCDWJob -ComputerName serverDW7 -JobName Extract_Contoso
```

## Related topics

[Disable-SCDWJob](#)

[Enable-SCDWJob](#)

[Get-SCDWJob](#)

[Stop-SCDWJob](#)

---

# Stop-SCDWJob

---

## Stop-SCDWJob

Stops a data warehouse job.

### Syntax

Parameter Set: Default

```
Stop-SCDWJob [-JobName] <String> [-ComputerName <String> ] [-Credential <PSCredential> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

### Detailed Description

The **Stop-SCDWJob** cmdlet stops a data warehouse job. Job modules that are currently running complete gracefully, and the rest of the job modules that have not run yet queue up.

### Parameters

#### **-ComputerName<String>**

Specifies the name of the computer on which the System Center Data Access service is running. The user account that is defined in the *Credential* parameter must have access rights to the specified computer. You can omit this parameter only if the System Center Data Access Service is running on the same computer that has Service Manager installed.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

---

## **-Credential<PSCredential>**

Specifies the credentials to use when you are connecting to the server on which the System Center Data Access service is running. The user account that is provided must have access to that server.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-JobName<String>**

Specifies the job to be stopped. The **JobName** parameter is mandatory.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-Confirm**

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

---

## -WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## <CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see [about\\_CommonParameters](#).

## Inputs

The input type is the type of the objects that you can pipe to the cmdlet.

- **None.** You cannot pipe input to this cmdlet.

## Outputs

The output type is the type of the objects that the cmdlet emits.

- **None.** This cmdlet does not generate any output.

## Examples

### ----- EXAMPLE 1 -----

This command stops the Extract\_Contoso job.

```
PS C:\>Stop-SCDWJob -ComputerName serverDW7 -JobName Extract_Contoso
```

### ----- EXAMPLE 2 -----

The first command prompts for user credentials and stores them in a variable.

The second command stops all jobs that do not have the status of stopped.

```
PS C:\> $cred = Get-Credential
PS C:\>Get-SCDWJob -ComputerName serverDW7 -Credential $cred | ForEach-Object {
  >> if ($_.Status -ne "Stopped") {
  >> Stop-SCDWJob -ComputerName IXSMTTestDW -JobName $_.Name
```



---

>> }  
>> }

## Related topics

[Disable-SCDWJob](#)

[Enable-SCDWJob](#)

[Get-SCDWJob](#)

[Start-SCDWJob](#)

---

# Unregister-SCDWMManagementPack

---

## Unregister-SCDWMManagementPack

Removes a management pack directly from the data warehouse.

### Syntax

Parameter Set: Default

```
Unregister-SCDWMManagementPack [-DataSourceTypeName] <String> [-DataSourceName] <String> [-ManagementPackBundle] <String> [-ComputerName <String> ] [-Credential <PSCredential> ] [-UninstallMP] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

### Detailed Description

The **Unregister-SCDWMManagementPack** cmdlet removes a management pack directly from the data warehouse.

### Parameters

#### **-ComputerName<String>**

Specifies the name of the computer on which the System Center Data Access service is running. The user account that is defined in the *Credential* parameter must have access rights to the specified computer. You can omit this parameter only if the System Center Data Access Service is running on the same computer that has Service Manager installed.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

---

## **-Credential<PSCredential>**

Specifies the credentials to use when you are connecting to the server on which the System Center Data Access service is running. The user account that is provided must have access to that server.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-DataSourceName<String>**

Species the name of the data source to unregister. You can use the **Get-SCDWSource** cmdlet to retrieve data source names.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-DataSourceTypeName<String>**

Specifies the type of the data source. You can use the **Get-SCDWSourceType** cmdlet to retrieve type name values.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

---

## **-ManagementPackBundle<String>**

Specifies the file that contains the management pack bundle that contains the management pack to unregister.

Aliases	none
Required?	true
Position?	3
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-UninstallMP**

Specifies that the management pack will be deleted, in addition to being unregistered.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-Confirm**

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

---

## -WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## <CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see [about\\_CommonParameters](#).

## Inputs

The input type is the type of the objects that you can pipe to the cmdlet.

- **None.** You cannot pipe input to this cmdlet.

## Outputs

The output type is the type of the objects that the cmdlet emits.

- **None.** This cmdlet does not generate any output.

## Examples

### ----- EXAMPLE 1 -----

This command unregisters the management packs in a specified management pack bundle and then uninstalls them.

```
PS C:\> Unregister-SCDWManagementPack -ComputerName serverDW7 -DataSourceType  
ConfigurationManager.DataSource -DataSourceName ADatum -ManagementPackBundle  
"C:\MPB\ADatumDataWarehouse.Library.mpb" -UninstallMP
```

---

# Unregister-SCDWSource

---

## Unregister-SCDWSource

Unregisters a data source from the data warehouse.

### Syntax

Parameter Set: Default

```
Unregister-SCDWSource [-DataSourceTypeName] <String> [-DataSourceName] <String> [-AdditionalData <Object> ] [-ComputerName <String> ] [-Credential <PSCredential> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

### Detailed Description

The **Unregister-SCDWSource** cmdlet unregisters a data source from the data warehouse. Any jobs that extract data from the unregistered data source are deleted. Any management packs that are synchronized from the unregistered data source are uninstalled. Any data that has already been retrieved from the data warehouse for the unregistered data source remains.

### Parameters

#### **-AdditionalData<Object>**

Specifies any additional data that a particular source has to send during registration. For example, a Configuration Manager source is registered, this parameter is used to pass a credential object that is later used by the Configuration Manager source to contact the Configuration Manager server.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

---

## **-ComputerName<String>**

Specifies the name of the computer on which the System Center Data Access service is running. The user account that is defined in the **Credential** parameter must have access rights to the specified computer. You can omit this parameter only if the System Center Data Access Service is running on the same computer that has Service Manager installed.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-Credential<PSCredential>**

Specifies the credentials to use when you are connecting to the server on which the System Center Data Access service is running. The user account that is provided must have access to that server.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

## **-DataSourceName<String>**

Specifies the name of the data source to use. Registered data sources may be retrieved by the **Get-SCDWSources** cmdlet.

Aliases	none
Required?	true
Position?	2
Default Value	none

Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-DataSourceTypeName<String>**

Specifies the name of the type of the data source. You can use the **Get-SCDWSourceType** cmdlet to retrieve **TypeNames**.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-Confirm**

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

### **-WhatIf**

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false



---

Accept Wildcard Characters?
-----------------------------

false
-------

## <CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see [about\\_CommonParameters](#).

## Inputs

The input type is the type of the objects that you can pipe to the cmdlet.

- **None.** You cannot pipe input to this cmdlet.

## Outputs

The output type is the type of the objects that the cmdlet emits.

- **None.** This cmdlet does not generate any output.

## Examples

### ----- EXAMPLE 1 -----

This command unregisters a specified data source.

```
PS C:\> UnRegister-SCDWSource -ComputerName serverDW7 -DataSourceTypeName "ServiceManager" -
DataSourceName "Contoso_2"
```

## Related topics

[Enable-SCDWSource](#)

[Get-SCDWSource](#)

[Set-SCDWSource](#)

[Register-SCDWSource](#)

[Disable-SCDWSource](#)