DocAve® 6 Service Pack 2

SDK and Management Shell

User Guide

Revision B

Issued February 2013

Table of Contents

About SDK and Management Shell 4
Configuration 4
Agents 4
Getting Started 5
Environment Requirement 5
DocAve 6 SDK 5
Supported Features in DocAve SDK 5
Breaking Changes in DocAve SDK 8
DocAve 6 Management Shell 10
Installation 10
Access DocAve 6 Management Shell 11
Upgrading DocAve SDK and Management Shell 12
DocAve Management Shell Cmdlet Reference 12
Log into DocAve Manager 12
Control Panel 13
Obtain a Date Object 13
Obtain a Time Object 13
Account Manager 14
Security Information Manager 18
Physical Device 21
Logical Device 21
Update Manager 22
Administration 24
Administrator 24
Content Manager 29
Deployment Manager 33
Replicator 34
Storage Optimization 36
BLOB Provider 36

Archiver 38
Real-Time Storage Manager 41
Job Monitor 44
Obtain a Specified Module's Job 44
Obtain a Job's Detailed Information 46
Obtain a Job's Summary Information 46
Download a Job Report 47
Migration 47
General SharePoint Migration Commands 47
SharePoint 2007 to 2010 Migration 50
SharePoint 2007 to 2013 Migration 54
SharePoint 2010 to 2013 Migration 60
File System Migration 66
eRoom Migration 70
Lotus Notes Migration 75
Data Protection 79
Granular Backup and Restore 79
Platform Backup and Restore 81
Notices and Copyright Information 83

About SDK and Management Shell

Refer to the sections below for system and farm requirements that must be in place prior to installing and using DocAve SDK and Management Shell.

Configuration

In order to use DocAve SDK and Management Shell, the DocAve 6 SP2 platform must be installed and configured properly on your farm. DocAve SDK and Management Shell will not function without DocAve 6 SP2.

Agents

DocAve Agents are responsible for running DocAve jobs and interacting with the SharePoint object model. DocAve Agents enable DocAve Manager to communicate with the respective servers, allowing for DocAve products to function properly. At the minimum, DocAve must have one agent installed on the SharePoint farm. The server that the agent is installed on must have Microsoft SharePoint Foundation Web Application service enabled as well. Otherwise, the Agent of the corresponding DocAve product cannot be loaded during agent installation and, consequently, the DocAve product functions cannot be used. Additional DocAve agents should be installed on other SharePoint Web front ends (WFEs) in order to provide redundancy in the event that a DocAve Agent goes down.

***Note**: The use of system resources on a server increases when the installed agent is performing actions. This may affect server performance. However, if the agent installed on a server is not being used, the use of system resources is very low and, therefore, the effect on server performance is negligible.

For instructions on installing DocAve Platform, the DocAve Manager, and DocAve module Agents, see the DocAve 6 Installation Guide.

Getting Started

Refer to the sections below for important information on getting started with SDK and Management Shell.

Environment Requirement

For **DocAve 6 SDK**, the .NET Framework version must be v3.5 SP1 or later.

For **DocAve 6 Management Shell**, the .NET Framework version must be 3.5 SP1 or later and the PowerShell version must be v2.0 or later.

DocAve 6 SDK

Build your customizable, extendable, and programmable usage of DocAve functions by invoking the program interfaces provided by **DocAve 6 SDK**.

To access detailed information on the commands available in the DocAve 6 SDK:

On the server with DocAve Manager/Agent installed, go to **Start** > **All Programs** > **AvePoint DocAve 6** > **DocAve 6 SDK**.

Double-click the help file named **DocAve SDK Documentation** to view this information. You can also find the help file in the following path:
…\AvePoint\DocAve6\Shell\DocAveModules\DocAveModule\DocAveSDK.chm

Supported Features in DocAve SDK

DocAve Version	Module	Supported Features
DocAve 6.0	Content Manager	Run a plan
	Deployment Manager	Run a plan
	Replicator	Run a plan
	Granular Backup and Restore	• Create a new plan • Update the settings of an existing plan • Delete a plan • Get all of the existing plans • Run a plan
	Platform Backup and Restore	Run a plan

DocAve Version	Module	Supported Features
	Archiver	• Get all of the existing Archiver rules • Get the rules that have been applied rules on a SharePoint object • Reuse an existing rule on a SharePoint object • Apply an existing rule on a SharePoint object • Run an Archiver plan
	Job Monitor	• Get the desired jobs • Get the summary information of a job • Get the detailed information of a job • Download the report of a job
DocAve 6 SP1	Administrator	• Clone user/group's permission • Scan dead account • Delete dead accounts • Search for users or groups with specific permissions
	Content manager	• Create/update /delete an online mode plan; • Get online mode plans
	Deployment manager	• Get an existing plan • Update the destination of a design element mapping of an existing plan • Create a new plan
	Archiver	• Check and change a SharePoint object's inheriting status • Get all of the existing Archiver profiles • Get the profiles that have been applied rules on a SharePoint object • Apply a profile on a SharePoint object • Retract an already applied profile
	Storage Optimization BLOB Provider Settings	• Get the configurations of a stub database • Configure the stub database settings • Get the RBS settings • Configure the RBS settings • Get the EBS settings • Configure the EBS settings
	Real-time Storage Manager	• Create a real-time storage manager rule • Update the settings of an existing real-time storage manager rule • Delete an existing real-time storage manager rule • Get all of the real-time storage manager rule • Reuse a real-time storage manager rule on a SharePoint object • Get the real-time storage manager rules that have been applied on a SharePoint object

DocAve Version	Module	Supported Features
		• Retract a real-time storage manager rule
	Scheduled Storage Manager	• Get all of the existing storage manager profiles • Reuse a storage manager profile on a SharePoint site collection • Retract a storage manager profile from a SharePoint site collection
	SharePoint 2007 to 2010 migration (online mode)	• Get all of the existing plans • Create a new plan • Update the settings of an existing plan • Delete a plan • Run a plan
	Update Manager	• Get all of the available update • Upload an update • Test an update • Install an update
	Storage Configuration	• Get all of the existing physical devices • Create a physical device of the CIFS Share type • Get all of the existing logical devices
	Account Manager	• Add a local user • Update the settings of an existing local system user • Delete a local system user • Add an active directory user • Update the settings of an existing active directory user • Delete an active directory user • Get all of the existing local users • Get all of the active directory users/groups
	Vault	• Get all of the existing profiles • Get the profiles that have been applied on a SharePoint object • Apply a profile on a SharePoint object • Check and change a SharePoint object's inheriting status
DocAve 6 SP2	SharePoint 2007 to 2010 Migration	• Download a profile • Upload a profile
	SharePoint 2007 to 2013 Migration	• Get all of the existing plans • Create a new plan • Update the settings of an existing plan • Delete a plan • Run a plan • Download profile • Upload profile
	SharePoint 2010 to 2013	• Get all of the existing plans

DocAve Version	Module	Supported Features
	Migration	• Create a new plan • Update the settings of an existing plan • Delete a plan • Run a plan • Download profile • Upload profile
	File System Migration	• Generate the metadata file(s) for the migrated contents • Get a blank plan • Get all of the existing plans • Create a new plan • Update the settings of an existing plan • Delete a plan • Run a plan • Update a plan
	eRoom Migration	• Get a blank plan • Get all of the existing plans • Create a new plan • Update the settings of an existing plan • Delete a plan • Run a plan
	Lotus Notes Migration	• Get a blank plan • Get all of the existing plans • Create a new plan • Update the settings of an existing plan • Delete a plan • Run a plan

Breaking Changes in DocAve SDK

Breaking Changes are changes in the DocAve SDK that make certain operations and features behave or name differently from the DocAve 6.0 to DocAve 6 SP1.

The followings are the breaking changes in DocAve SDK from DocAve 6 to DocAve 6 SP1:

- One more library file is required: Storage.dll, which resides in *DocAveInstallDir**Shell**DocAveModules**DocAveModule* directory.
- The namespace is rearranged in this service pack.
- For some methods of job monitor service, the type of integer parameters is changed from *long* to *int*.
- To avoid ambiguity and confusion with the native SharePoint API types, rename type names under ***AvePoint.DocAve.API.Objects*** as the table displays.

Original Object Name	Change To
SPList	SharePointList
SPObject	SharePointObject
SPSite	SharePointSite
SPTree	SharePointTree
SPWeb	SharePointWeb
SPWebApplication	SharePointWebApplication

- To make it clear, the *ContentManagerService* is divided into *ContentManagerOnlineService* and *ContentManagerOfflineService* two services.
- The type *GranularBackupAndRestoreService* is renamed to *GranularBackupService*, and the property name of *DataProtectionService* is changed accordingly.
  The type *PlatformBackupAndRestoreService* is renamed to *PlatformBackupService*, and the property name of *DataProtectionService* is changed accordingly.

DocAve 6 Management Shell

DocAve 6 provides a command line management interface based on Windows PowerShell. You can perform the DocAve operations using the provided cmdlet.

Installation

By default, **DocAve 6 Management Shell** is installed together with DocAve Manager/Agent. You can find it in **Start** > **All Programs** > **AvePoint DocAve 6** > **DocAve 6 Management Shell.**

You can also install DocAve 6 Management Shell on the server that does not have DocAve Manager/Agent installed. To install **DocAve 6 Management Shell** separately, see the following steps.

1. In the DocAve installation path (…\ *AvePoint**DocAve6*), copy the folder named **Shell** to the server where you want to install DocAve Management Shell.
2. Right-click **ShellInstallation.exe** under …*Shell**Setup* and select **Run as administrator**.
3. Specify a local installation path or press **Enter** to use the default location (which is *C:\Program File**AvePoint*).

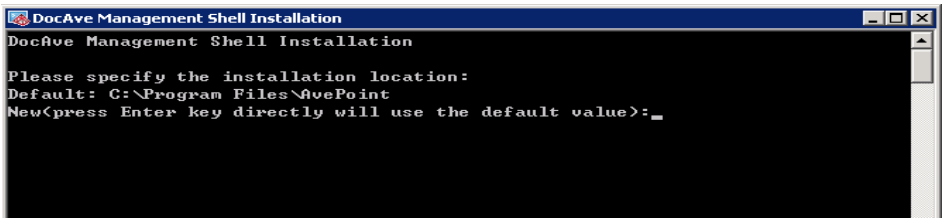

Figure 1: Specify the installation location.

4. Press any key to exit when the installation completes.

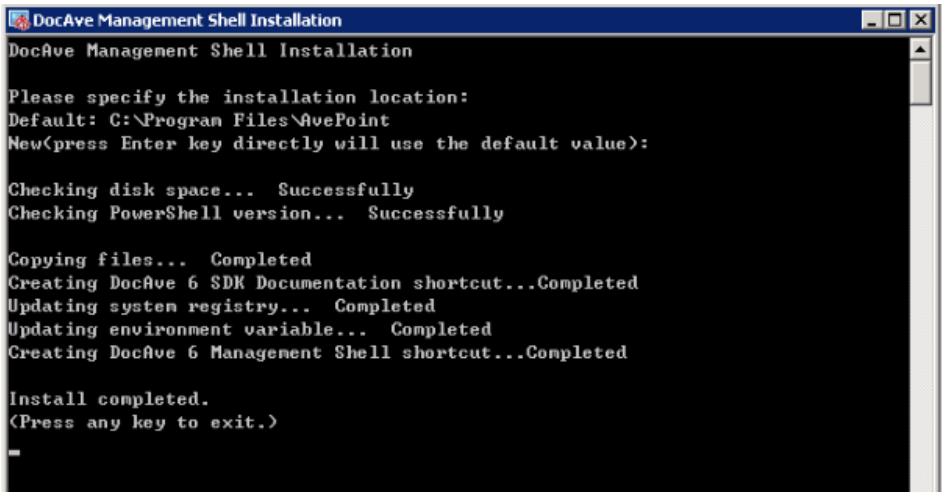

Figure 2: Exit the tool.

Access DocAve 6 Management Shell

After installing **DocAve 6 Management Shell**, refer to the following two methods to access it.

- Go to **Start** > **All Programs** > **AvePoint DocAve 6** >**DocAve6 Management Shell** and click **DocAve 6 Management Shell** to open the interface.
- Use **Windows PowerShell**. Click the Windows PowerShell icon on the left of Windows taskbar to open it. Then, import the DocAve 6 PowerShell module by executing the command: **Import-Module -Name DocAveModule -DisableNameChecking**

***Note**: Before using any DocAve cmdlet, you must log into DocAve using **Login-DAManager** cmdlet first. You can get the detailed information on using this cmdlet entering **Get-Help Login-DAManager**.

To get the overview of all of the commands available for DocAve 6, execute the command **Get-Command -Module DocAveModule**.

To search all of the commands related to a module, use a “fuzzy” query. For example, execute the command **Get-Command -Name *GranularBackup*** to get all of the commands about the **Granular Backup** module.

Upgrading DocAve SDK and Management Shell

The DocAve SDK and Management Shell must be upgraded together with the DocAve platform. The upgraded DocAve SDK and Management Shell support more functions and contain optimizations and improvements.

For detailed steps regarding how to upgrade the **DocAve 6 SDK**, refer to **DocAve SDK Documentation**; it is available in the following path by default: *…\AvePoint\DocAve6\Shell\DocAveModules\DocAveModule\DocAveSDK.chm*.

For detailed steps regarding how to upgrade the **DocAve 6 Management Shell**, refer to **Conceptual Topic about Update** in the **DocAve 6 Management Shell**.

DocAve Management Shell Cmdlet Reference

DocAve Management Shell provides a rich set of commands based on the DocAve platform. These commands allow you to perform various operations including run a plan, delete a plan, delete dead account etc. The section below lists all of the DocAve Management Shell cmdlets, a corresponding example of each cmdlet, as well as the descriptions of the parameters in each cmdlet.

Log into DocAve Manager

To log into DocAve manager, run the command below:

Login-DAManager -ControlHost <String> -ControlPort <Int32> -Credential <PSCredential> [<CommonParameters>]

The examples of the command are:

C:\PS>$Credential = Get-Credential
C:\PS>Login-DAManager -ControlHost localhost -ControlPort 14000 -Credential $credential

Or
C:\PS>$SecurePassword = Read-Host -AsSecureString
C:\PS>Login-DAManager -ControlHost localhost -ControlPort 14000 -Username admin -Password $SecurePassword

Or
C:\PS>Login-DAManager -ControlHost localhost -ControlPort 14000 -Username admin -PlainTextPassWord admin

To configure the command parameters, see the table below.

Parameter	Type	Comment
-ControlHost	Required	Specifies the address of control host.
-ControlPort	Required	Specifies the control port.
-Credential	Required	Specifies the credential.
-Username	Required	Specifies the username.
-Password	Required	Specifies the password in the form of secure string.
-PlainTextPassword	Required	Specifies the password in the form of plain text string.

Control Panel

Obtain a Date Object

To obtain a date object used for defining a specific date in a scheduled plan, run the command below:

Get-DADate [-Year] <Int32> [-Month] <Int32> [-Day] <Int32> [-TimeZoneId] <SupportedTimeZoneId> [<CommonParameters>]

The example of the command is:

C:\PS>$TimeZoneID = [DocAve.API.Common.SupportedTimeZoneId]::PacificSAStandardTime
C:\PS>Get-DADate -Year 2012 -Month 03 -Day 20 –TimeZoneId $TimeZoneId

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Year	Required	Specifies the year that is displayed. Valid value is between 1 and 9999.
-Month	Required	Specifies the month that is displayed. Valid value is between 1 and 12.
-Day	Required	Specifies the day that is displayed. Valid value is between 1 and 31.
-TimeZoneId	Required	Specifies the time zone ID.

Obtain a Time Object

To get a time object used for defining a specific time in a scheduled plan, run the command below:

Get-DADateTime [-Year] <Int32> [-Month] <Int32> [-Day] <Int32> [-Hour] <Int32> [-Minute] <Int32> [-TimeZoneId] <SupportedTimeZoneId> [<CommonParameters>]

The example of the command is:

C:\PS>$TimeZoneID = [DocAve.API.Common.SupportedTimeZoneId]::PacificSAStandardTime
C:\PS>Get-DADateTime -Year 2012 -Month 03 -Day 20 -Hour 14 -Minute 30 -TimeZoneId $TimeZoneID

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Year	Required	Specifies the year that is displayed. Valid value is between 1 and 9999.
-Month	Required	Specifies the month that is displayed. Valid value is between 1 and 12.
-Day	Required	Specifies the day that is displayed. Valid value is between 1 and 31.
-Hour	Required	Specifies the hour that is displayed. Valid value is between 0 and 23.
-Minute	Required	Specifies the minute that is displayed. Valid value is between 0 and 59.
-TimeZoneId	Required	Specifies the time zone ID.

Account Manager

Get the Information of an Active Directory Group

To get the information of a specified active directory group, run the command below:

Get-DAActiveDirectoryGroup [[-Name] <String[]>] [<CommonParameters>]

The example of the command is:

C:\PS>$user = Get-DAActiveDirectoryGroup

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Name	Optional	Specifies the name(s) of the active directory group(s) you want to obtain. Wildcards and multiple values are supported, which are separated by commas. If no values are entered, all existing active directory groups will be obtained.

Get the Information of an Active Directory

To get the information of an active directory user, run the command below:

Get-DAActiveDirectoryUser [[-Name] <String[]>] [<CommonParameters>]

The example of the command is:

C:\PS>$user = Get-DAActiveDirectoryUser

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Name	Optional	Specifies the name(s) of the active directory user(s) you want to obtain. Wildcards and multiple values are supported, which are separated by commas. If no values are entered, all existing active directory users will be obtained.

Get the Information of a Local User

To get the information of a local user, run the command below:

Get-DALocalUser [[-Name] <String[]>] [<CommonParameters>]

The example of the command is:

C:\PS>Get-DALocalUser -Name user1

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Name	Optional	Specifies the name(s) of the local user(s) you want to obtain. Wildcards and multiple values are supported, which should be separated by commas. If no values are entered, all existing local users will be obtained.

Get the Blank Active Directory Group

To get the blank active directory group, run the command below:

Get-DABlankActiveDirectoryGroup [<CommonParameters>]

The example of the command is:

C:\PS>$user = Get-DABlankActiveDirectoryGroup

Get a Blank Active Directory User

To get the blank active directory user, run the command below:

Get-DABlankActiveDirectoryUser [<CommonParameters>]

The example of the command is:

C:\PS>Get-DABlankActiveDirectoryUser

Get a Blank Local User

To get a blank local user, run the command below:

Get-DABlankLocalUser [<CommonParameters>]

The example of the command is:

C:\PS>Get-DABlankLocalUser

Create a New User

To create a new user, run the command below:

New-DAUser [-User] <UserBase> [<CommonParameters>]

The examples of the command are:

C:\PS>$user = Get-DABlankLocalUser
C:\PS>$user.Name = 'user1'
C:\PS>$user.Email = 'user1@example.com'
C:\PS>$user.Password = Read-Host -AsSecureString

****
C:\PS>$user.Description = 'This is a Local User'
C:\PS>$user.Groups.Add('group1')
C:\PS>New-DAUser $user

Or
C:\PS>$user = Get-DABlankActiveDirectoryUser
C:\PS>$user.Name = 'example.com\user1
C:\PS>$user.Groups.Add('group1')
C:\PS>New-DAUser $user

To configure the command parameters, see the table below.

Parameter	Type	Comment
-User	Required	Specifies a user object to create a new user. For the structure and attributes of user, please refer to the **DocAve SDK Documentation** for more information.

Delete an Active Directory Group

To delete an active directory group, run the command below:

Remove-DAActiveDirectoryGroup [-Name] <String> [-WhatIf] [-Confirm] [<CommonParameters>]

The example of the command is:

C:\PS>Remove-DAActiveDirectoryGroup -Name 'example.com\group1'

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Name	Required	Specifies the name of an active directory group that you want to delete.
-WhatIf	Optional	Describes what would happen if you executed the command without actually executing the command.
-Confirm	Optional	Prompts you for confirmation before executing the command.

Delete an Active Directory User

To delete an active directory user, run the command below:

Remove-DAActiveDirectoryUser [-Name] <String> [-WhatIf] [-Confirm] [<CommonParameters>]

The example of the command is:

C:\PS>Remove-DAActiveDirectoryUser -Name 'example.com\user1'

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Name	Required	Specifies the name of an active directory user that you want to delete.
-WhatIf	Optional	Describes what would happen if you executed the command.
-Confirm	Optional	Prompts you for confirmation before executing the command.

Delete a Local User

To delete a local user, run the command below:

Remove-DALocalUser [-Name] <String> [-WhatIf] [-Confirm] [<CommonParameters>]

The example of the command is:

C:\PS>Remove-DALocalUser -Name 'user1'

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Name	Required	Specifies the name of a local user that you want to delete.
-WhatIf	Optional	Describes what would happen if you executed the command.
-Confirm	Optional	Prompts you for confirmation before executing the command.

Update the Information of a User in Account Manager

To update the information of an existing user, run the command below:

Update-DAUser [-User] <UserBase> [<CommonParameters>]

The example of the command is:

C:\PS>$user = Get-DALocalUser -Name 'user1'
C:\PS>$user.Email = 'user1@example.com'
C:\PS>Update-DAUser $user

To configure the command parameters, see the table below.

Parameter	Type	Comment
-User	Required	Specifies a user object where you want to edit the user.

Security Information Manager

Back Up a Database Encryption Key

To back up a database encryption key, run the command below:

Backup-DADatabaseEncryptionKey [-Path] <String> [-Overwrite] [<CommonParameters>]

The example of the command is:

C:\PS>Backup-DADatabaseEncryptionKey -Path 'C:\backup.zip'

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Path	Required	Specifies a path to store the backup file.
-Overwrite	Optional	Overwrites the backup file if it already exists in the specified path.

Back Up the Data Encryption Profile

To back up the database encryption profile, run the command below:

Backup-DADataEncryptionProfile [-Path] <String> [-Overwrite] [<CommonParameters>]

***Note:** Data encryption profile is renamed to security profile since DocAve 6 SP1.

The example of the command is:

C:\PS>Backup-DADataEncryptionProfile -Path 'C:\backup.zip'

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Path	Required	Specifies a path to store the backup file.
-Overwrite	Optional	Overwrites the backup file if it already exists in the specified path.

Restore a Database Encryption Key

To restore a database encryption key, run the command below:

Restore-DADatabaseEncryptionKey [-Path] <String> [-OutPut] <String> [-Overwrite] [[-Passphrase] <String>] [<CommonParameters>]

The examples of the command are:

C:\PS>Restore-DADatabaseEncryptionKey -Path 'C:\backupfile' -OutPut 'C:\sql'
C:\PS>Restore-DADatabaseEncryptionKey -Path 'C:\backupfile' -OutPut 'C:\sql' -Passphrase passphrase

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Path	Required	Specifies a path to load the backup file.
-OutPut	Required	Specifies a path to store the output file.
-Overwrite	Optional	Overwrites the output file if it already exists in the specified path.
-Passphrase	Optional	Specifies a passphrase value to encrypt DocAve database encryption key. If this parameter is not used, the cmdlet will read the passphrase from the local machine. Ensure that the passphrases are matched between the local machine and the DocAve database; otherwise, the DocAve system will not work.

Restore a Data Encryption Profile

To restore a data encryption profile, run the command below:

*Restore-DADataEncryptionProfile [-Path] <String> [<CommonParameters>]***Note:** Data encryption profile is renamed to security profile since DocAve 6 SP1.

The example of the command:

C:\PS>Backup-DADataEncryptionProfile -Path 'C:\backup.zip'

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Path	Required	Specifies a path to load the backup file.

Set Up the Status of a FipsAlgorithmPolicy

To set up the status of a specified FipsAlogrithmPolicy, run the command below:

Set-DAFipsAlgorithmPolicy [[-Disabled]] [<CommonParameters>]

Set-DAFipsAlgorithmPolicy [[-Enabled]] [<CommonParameters>]

The examples of the command are:

C:\PS>Set-DAFipsAlgorithmPolicy -Enabled
C:\PS>Set-DAFipsAlgorithmPolicy -Disabled

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Disabled	Optional	Disables the FipsAlogrithmPolicy.
-Enabled	Optional	Enables the FipsAlogrithmPolicy.

Set Up the Passphrase

To set up a new value for the passphrase, run the command below:

Set-DAPassPhrase [-PassPhrase] <SecureString> [-ConfirmPassPhrase] <SecureString> [<CommonParameters>]

Set-DAPassPhrase [-PassPhrase] <SecureString> [-LocalServerOnly] [<CommonParameters>]

The examples of the command are:

C:\PS>$passphrase = ConvertTo-SecureString 'PassPhrase' -AsPlainText -Force
C:\PS>$ConfirmPassPhrase = ConvertTo-SecureString 'PassPhrase' -AsPlainText -Force
C:\PS>Set-DAPassPhrase -PassPhrase $passphrase -ConfirmPassPhrase $ConfirmPassPhrase

Or

C:\PS>$passphrase = ConvertTo-SecureString 'PassPhrase' -AsPlainText -Force
C:\PS>Set-DAPassPhrase -PassPhrase $passphrase -LocalServerOnly

To configure the command parameters, see the table below.

Parameter	Type	Comment
-PassPhrase	Required	Specifies a new passphrase value.
-ConfirmPassPhrase	Required	Confirms the new passphrase value.
-LocalServerOnly	Optional	This parameter will only change the passphrase of this local server. If this parameter is not used, the passphrase change will be performed on all the servers in the DocAve system.

Physical Device

Get the Configuration Details of a Physical Device

To get the configuration details of a physical device, run the command below:

Get-DAPhysicalDevice [[-Name] <String[]>] [<CommonParameters>]

Get-DAPhysicalDevice [-Type] <StorageType> [<CommonParameters>]

The examples of the command are:

C:\PS>Get-DAPhysicalDevice -Name 'physical device1'
C:\PS>Get-DAPhysicalDevice

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Name	Optional	Specifies one or more physical device names. Multiple physical device names must be separated by commas. Wildcards are supported. If you do not type a value, the information of all physical devices will be returned.
-Type	Required	Specifies the physical device type.

Logical Device

Get the Configuration Details of a Logical Device

To get the configuration details of a logical device, run the command below:

Get-DALogicalDevice [[-Name] <String[]>] [<CommonParameters>]

The examples of the command are:

C:\PS>Get-DALogicalDevice -Name 'logical device1'
C:\PS>Get-DALogicalDevice

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Name	Optional	Specifies one or more logical device names. Multiple logical device names must be separated by commas. Wildcards are supported. If you do not type any value for the parameter, the information of all the logical devices will be returned.

Update Manager

Upload an Update in Update Manager

To upload a specified update, run the command below:

Upload-DAUpdate [-Path] <String> [-Overwrite] [<CommonParameters>]

The example of the command is:

C:\PS>Upload-DAUpdate -Path 'C:\Hotfix_6.0.0.51008601.zip' -Overwrite

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Path	Required	Specifies the path of the update file.
-Overwrite	Optional	Allows the cmdlet to overwrite an existing update.

Get an Update

To get all available updates for DocAve Manager, run the command below:

Get-DAUpdate [[-Name] <String[]>] [<CommonParameters>]

The example of the command is:

C:\PS>Get-DAUpdate -Name 'HotFix_6.0.0.51008601'

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Name	Optional	Specifies one or more updates' names. You can type multiple update names (separated by commas). Wildcards are supported.

Obtain Update Configuration Information

To obtain general configuration information of an update, run the command below:

Get-DAUpdateConfiguration [-UpdateName] <String[]> [<CommonParameters>]

The example of the command is:

C:\PS>Get-DAUpdateConfiguration -UpdateName Hotfix_6.0.0.51008601

To configure the command parameters, see the table below.

Parameter	Type	Comment
-UpdateName	Required	Specifies one or more updates' names. You can type multiple update names (separated by commas).

Test an Update

To test whether a specified update is valid before updating, run the command below:

Test-DAUpdate [-Configuration] <UpdateConfiguration> [<CommonParameters>]

The example of the command is:

C:\PS>$config = Get-DAUpdateConfiguration -UpdateName 'Hotfix_6.0.0.51008601'
C:\PS>$config.UpdateManager = $true
C:\PS>$config.ViewServiceForUpdate()
Category: 127.0.0.1
WillUpdate: True
Name: Control Service
Status: ACTIVE
Version: 6.0.0.51008601
Type: Control
Message: Ready for installing.
C:\PS>$config.UpdateNonFarmRelatedAgent = $true
C:\PS>$config.UpdateFarms.Add('Farm(SP2010_DB:SHAREPOINT_CONFIG)'
C:\PS>$config.UpdateSQLAgents.Add('SQL Agent'
C:\PS>Test-DAUpdate -Configuration $config True

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Configuration	Required	Specifies an update configuration object.

Install an Update

To install an update, run the command below:

Install-DAUpdate [-Configuration] <UpdateConfiguration> [-WhatIf] [-Confirm] [<CommonParameters>]Install-DAUpdate [-Configuration] <UpdateConfiguration> [-WhatIf] [-Confirm] [<CommonParameters>]

The example of the command is:

C:\PS>$config = Get-DAUpdateConfiguration -UpdateName 'Hotfix_6.0.0.51008601'
C:\PS>$config.UpdateManager = $true
C:\PS>$config.ViewServiceForUpdate()
C:\PS>$config.UpdateNonFarmRelatedAgent = $true
C:\PS>$config.UpdateFarms.Add('Farm(SP2010_DB:SHAREPOINT_CONFIG)')
C:\PS>$config.UpdateSQLAgents.Add('SQL Agent')
C:\PS>Install-DAUpdate -Configuration $config

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Configuration	Required	Specifies an update configuration object.
-WhatIf	Optional	Describes what would happen if user executes the command without actually executing the command.
-Confirm	Optional	Prompts user for confirmation before executing the command.

Administration

Administrator

Clone the SharePoint Group Permission

To clone the SharePoint Group Permission, run the command below:

Clone-DASPGroupPermission [-Scope] <SharePointObject> [-SourceGroup] <String> [-Destination] <String> [-Setting] <CloneGroupPermissionSetting> [<CommonParameters>]

Clone-DASPGroupPermission [-FarmName] <String> [-SourceGroup] <String> [-Destination] <String> [-Setting] <CloneGroupPermissionSetting> [<CommonParameters>]

The example of the command is:

C:\PS>$setting = Get-DASPBlankCloneGroupPermissionSetting

C:\PS>$setting.PermissionOption = 'Append'

C:\PS>$setting.IncludeDetailedReport = $true

C:\PS>

C:\PS>$webapplication1= New-Object DocAve.API.Objects.SharePointWebApplication ('Farm(SP2010_DB:SHAREPOINT_CONFIG)', 'http://SharePoint2010:10001/')

C:\PS>Clone-DASPGroupPermission -Scope $webapplication1 -SourceGroup 'group1' -Destination 'user1' -Setting $setting

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Scope	Required	Specifies the node(s) as a scope where you wish to search for the source group. You can specify the node(s) from web application level to list level. And they must be at the same level.
-SourceGroup	Required	Specifies a group as the source whose permissions will be cloned.
-Destination	Required	Specifies the username(s) or group name(s) as the destination that you can clone permissions to.
-Setting	Required	Configures the settings for cloning group permissions.
-FarmName	Required	Specifies the farm where you wish search for the source group.

Obtain a SharePoint Blank Security Search Setting Object

To get a SharePoint blank security search setting object, run the command below:

Get-DASPBlankSecuritySearchSetting [<CommonParameters>]

The example of the command is:

C:\PS>$setting = Get-DASPBlankSecuritySearchSetting

Delete a SharePoint Dead Account

To delete the SharePoint dead account(s), run the command below:

Remove-DASPDeadAccount [-DeadAccount] <DeadAccount[]> [-DeleteMySite] [-DeleteUserProfiles] [<CommonParameters>]

The example of the command is:

C:\PS>Remove-DASPDeadAccount -DeadAccount $deadAccountForRemoving -DeleteMySite -DeleteUserProfiles

To configure the command parameters, see the table below.

Parameter	Type	Comment
-DeadAccount	Required	Specifies the dead accounts to remove.
-DeleteMySite	Optional	Specifies whether or not to delete the My Site.
-DeleteUserProfiles	Optional	Specifies whether or not to delete user profiles.

Check Permission Inheriting Information

To check whether a specified SharePoint object inherits permissions from a higher level, run the command below:

Check-DASPInheritingPermissions [-SharePointWeb] <SharePointWeb> [<CommonParameters>]

Check-DASPInheritingPermissions [-SharePointList] <SharePointList> [<CommonParameters>]

The example of the command is:

C:\PS>$web1 = New-Object DocAve.API.Objects.SharePointWeb ('Farm(SP2010_DB:SHAREPOINT_CONFIG)','http://SharePoint:1 0001/sites/site1','/sites/site1')
C:\PS>Check-DASPInheritingPermissions -SharePointWeb $web1

To configure the command parameters, see the table below.

Parameter	Type	Comment
-SharePointWeb	Required	Specifies a SharePoint site object.
-SharePointList	Required	Specifies a SharePoint list object.

Get a Blank Clone SharePoint User Permission Setting Object

To get a blank clone SharePoint user permission setting object, run the command below:

Get-DASPBlankCloneUserPermissionSetting [<CommonParameters>]

The example of the command is:

C:\PS>Get-DASPBlankCloneUserPermissionSetting

Get a Blank Clone SharePoint Group Permission Setting Object

To get a blank clone SharePoint group permission setting object, run the command below:

Get-DASPBlankCloneGroupPermissionSetting [<CommonParameters>]

The example of the command is: *C:\PS>Get-DASPBlankCloneGroupPermissionSetting*

Clone SharePoint User Permission

To clone a specified SharePoint user's permissions to a group or a user, run the command below:

Clone-DASPUserPermission [-Scope] <SharePointObject> [-SourceUser] <String> [-Destination] <String> [-Setting] <CloneUserPermissionSetting> [<CommonParameters>]

Clone-DASPUserPermission [-FarmName] <String> [-SourceUser] <String> [-Destination] <String> [-Setting] <CloneUserPermissionSetting> [<CommonParameters>]

The example of the command is:

C:\PS>$setting = Get-DASPBlankCloneUserPermissionSetting
C:\PS>$setting.PermissionOption = 'Append'
C:\PS>$setting.IncludeDetailedReport = $true
C:\PS>Clone-DASPUserPermission -FarmName 'Farm(SP2010_DB:SHAREPOINT_CONFIG)' -SourceUser 'user1' -Destination 'user2' -Setting $setting

To configure the command parameters, see the table below.

Parameter	Type	Comment
-FarmName	Required	Specifies the farm where you wish to search for the source user.
-Scope	Required	Specifies the node(s) as a scope where you wish to search for the source user. You can specify the node(s) from Web application level to list level, but all objects must be at the same level.
-SourceUser	Required	Specifies a username as the source account whose permissions will be cloned.
-Destination	Required	Specifies the username(s) or group name(s) as the destination account(s) that you can clone permissions to.
-Setting	Required	Configures the settings for cloning user permissions.

Scan for Dead Accounts

To scan for dead accounts within a specified scope, run the command below:

Scan-DASPDeadAccount [-Scope] <SharePointObject[]> [<CommonParameters>]

Scan-DASPDeadAccount [-FarmName] <String> [<CommonParameters>]

The examples of the command are:

C:\PS>$site1 = New-Object DocAve.API.Objects.SharePointSite ('Farm(SP2010_DB:SHAREPOINT_CONFIG)','http://SharePoint2010:10001/sites/site1')
C:\PS>$site2 = New-Object DocAve.API.Objects.SharePointSite ('Farm(SP2010_DB:SHAREPOINT_CONFIG)','http://SharePoint2010:10001/sites/site2')
C:\PS>$result = Scan-DASPDeadAccount -Scope $site1,$site2
C:\PS>$result | fl
C:\PS>$result = Scan-DASPDeadAccount -FarmName 'Farm(SP2010_DB:SHAREPOINT_CONFIG)'
C:\PS>$result | fl

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Scope	Required	Specifies Web application(s), site collection(s), or site(s) as the object(s) where you want to scan the dead accounts. The objects you specify must be at the same level.
-FarmName	Required	Specifies a farm name.

Search Users or Groups with Specific Permissions

To search SharePoint users or groups who have specific permissions, run the command below:

Search-DASPPermission [-Scope] <SharePointObject[]> [-Setting] <SecuritySearchSetting> [<CommonParameters>]

Search-DASPPermission [-FarmName] <String> [-Setting] <SecuritySearchSetting> [<CommonParameters>]

The examples of the command are:

C:\PS>$setting = Get-DASPBlankSecuritySearchSetting
C:\PS>$setting.UserAndGroup.Add
C:\PS>$webapplication1 = New-ObjectDocAve.API.Objects.SharePointWebApplication
C:\PS>$webapplication2 = New-Object DocAve.API.Objects.SharePointWebApplica
tion('Farm(SP2010_DB:SHAREPOINT_CONFIG)','http://SharePoint2010:10002')
C:\PS>Search-DASPPermission -Scope $webapplication1,$webapplication2 -Setting $setting
C:\PS>$setting = Get-DASPBlankSecuritySearchSetting
C:\PS>$setting.IncludeSharePointUsersAndGroupsWithNoPermissions = $true
C:\PS>$setting.LimitedPermissions = 'FullControl'
C:\PS>$setting.OnlySearchForExplicitPermissions = $true
C:\PS>$setting.PermissionPolicy = 'ExactPermissionOrBetter' C:\PS>Search-DASPPermission -FarmName 'Farm(SP2010_DB:SHAREPOINT_CONFIG)' -Setting $setting

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Scope	Required	Specifies the scope with the node(s) you want to search.
-Setting	Required	Specifies a security search settings.
-FarmName	Required	Specifies a farm name.

Stop Inheriting Permissions from the Upper Level

To stop the permission inheritance relationship for a specified object from an upper level, run the command below:

Stop-DASPInheritingPermissions [-SharePointWeb] <SharePointWeb[]> [<CommonParameters>]

Stop-DASPInheritingPermissions [-SharePointList] <SharePointList[]> [<CommonParameters>]

The examples of the command are:

C:\PS>$web = New-Object DocAve.API.Objects.SharePointWeb ('Farm(SP2010_DB:SHAREPOINT_CONFIG)','http://SharePoint2010:10001/sites/site1','/sites/site1')
C:\PS>Stop-DASPInheritingPermissions -SharePointWeb $web
C:\PS>$list = New-Object DocAve.API.Objects.SharePointList ('Farm(SP2010_DB:SHAREPOINT_CONFIG)','http://SharePoint2010:10001/sites/site1','sites/site1','list1')
C:\PS>Stop-DASPInheritingPermissions -SharePointList $list

To configure the command parameters, see the table below.

Parameter	Type	Comment
-SharePointWeb	Required	Specifies the site(s) as the object(s) where you want to break the permission inheritance.
-SharePointList	Required	Specifies the list(s) as the object(s) where you want to break the permission inheritance.

Content Manager

Get a Blank Content Manager Online Plan

To get a blank Content Manager online plan, run the command below:

Get-DAContentManagerBlankOnlinePlan [<CommonParameters>]

The example of the command is:

C:\PS>Get-DAContentManagerBlankOnlinePlan

Get an Online Content Manager Plan's Configuration Details

To get the configuration details of an online Content Manager plan, run the command below:

Get-DAContentManagerOnlinePlan [[-Name] <String[]>] [<CommonParameters>]

The example of the command is:

C:\PS>Get-DAContentManagerOnlinePlan plan1

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Name	Optional	Specifies the name(s) of the existing online Content Manager plan(s) you want to obtain. Wildcards and multiple values are allowed, which

		should be separated by commas. If no values are entered here, the plan settings of all existing online Content Manager plan(s) will be obtained.

Create an Online Content Manager Plan

To create an online Content Manager plan, run the command below:

New-DAContentManagerOnlinePlan [-Plan] <ContentManagerOnlinePlan> [<CommonParameters>]

The example of the command is:

C:\PS>$plan = Get-DAContentManagerBlankOnlinePlan
C:\PS>$plan.SourceFarmName = 'Farm(SP2010_DB1:SHAREPOINT_CONFIG)'
C:\PS>$plan.DestinationFarmName = 'Farm(SP2010_DB2:SHAREPOINT_CONFIG)'
C:\PS>$plan.Name = 'plan1'
C:\PS>$webapplication1 = New-Object DocAve.API.Objects.SharePointWebApplication($plan.SourceFarmName, 'http://SharePoint2010:10001')
C:\PS>$webapplication2 = New-Object DocAve.API.Objects.SharePointWebApplication($plan.DestinationFarmName,'http://SharePoint2010:10002') C:\PS>$plan.SourceTree.IncludeSPObject($webapplication1)
C:\PS>$plan.DestinationTree.SelectSPObject($webapplication2)
C:\PS>$plan.Action = 'Merge'
C:\PS>New-DAContentManagerOnlinePlan $plan

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Plan	Required	Specifies an online Content Manager plan object. For the structure and attributes of ContentManagerOnlinePlan, please refer to the DocAve 6 SDK documentation for more information.

Run a Content Manager Export Plan

To run an existing Content Manager export plan, run the command below:

Run-DAContentManagerExportJob [-PlanName] <String> [-WaitForJobComplete] [<CommonParameters>]

The example of the command:

C:\PS>Run-DAContentManagerExportJob -PlanName 'plan1' -WaitForJobComplete

To configure the command parameters, see the table below.

Parameter	Type	Comment
-PlanName	Required	Specifies the name of an existing Content Manager export plan that you want to run.
-WaitForJobComplete	Optional	Specifies whether to wait for the completion of the job. If so, the job process will be displayed and another job cannot be run until this job has completed; if not, the job will run in the background and you can run another job without waiting for the completion of this job.

Run a Content Manager Import Plan

To run an existing Content Manager import plan, run the command below:

Run-DAContentManagerImportJob [-PlanName] <String> [-WaitForJobComplete] [<CommonParameters>]

The example of the command:

C:\PS>Run-DAContentManagerImportJob -PlanName 'plan1' -WaitForJobComplete

To configure the command parameters, see the table below.

Parameter	Type	Comment
-PlanName	Required	Specifies the name of an existing Content Manager import plan that you want to run.
-WaitForJobComplete	Optional	Specifies whether to wait for the completion of the job. If so, the job process will be displayed and another job cannot be run until this job has completed; if not, the job will run in the background and you can run another job without waiting for the completion of this job.

Run a Content Manager Copy Plan

To run an existing Content Manager copy plan, run the command below:

Run-DAContentManagerCopyJob [-PlanName] <String> [-TestRun] [-WaitForJobComplete] [<CommonParameters>]

The example of the command:

C:\PS>Run-DAContentManagerCopyJob -PlanName 'plan1' -TestRun -WaitForJobComplete

To configure the command parameters, see the table below.

Parameter	Type	Comment
-PlanName	Required	Specifies the name of an existing Content Manager copy plan that you want to run.

Parameter	Type	Comment
-TestRun	Optional	Specifies whether to simulate the execution of the selected plan.
-WaitForJobComplete	Optional	Specifies whether to wait for the completion of the job. If so, the job process will be displayed and another job cannot be run until this job has completed; if not, the job will run in the background and you can run another job without waiting for the completion of this job.

Run a Content Manager Move Plan

To run an existing Content Manager move plan, run the command below:

Run-DAContentManagerMoveJob [-PlanName] <String> [-TestRun] [-WaitForJobComplete] [<CommonParameters>]

The example of the command:

C:\PS>Run-DAContentManagerMoveJob -PlanName 'plan1' -TestRun -WaitForJobComplete

To configure the command parameters, see the table below.

Parameter	Type	Comment
-PlanName	Required	Specifies the name of an existing Content Manager move plan that you want to run.
-TestRun	Optional	Specifies whether to simulate the execution of the selected plan.
-WaitForJobComplete	Optional	Specifies whether to wait for the completion of the job. If so, the job process will be displayed and another job cannot be run until this job has completed; if not, the job will run in the background and you can run another job without waiting for the completion of this job.

Delete a Content Manager Online Plan

To delete a content manager online plan, run the command below:

Remove-DAContentManagerOnlinePlan [-Name] <String> [-WhatIf] [-Confirm] [<CommonParameters>]

The example of the command is:

C:\PS>Remove-DAContentManagerOnlinePlan -Name 'plan1'

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Name	Required	Specifies the name of the online Content Manager plan that you want to delete.

Parameter	Type	Comment
-WhatIf	Optional	Describes what would happen without actually executing the command.
-Confirm	Optional	Prompts you for confirmation before executing the command.

Update the Configurations of a Content Manager Plan

To update the configurations of an existing Content Manager plan, run the command below:

Update-DAContentManagerOnlinePlan [-Plan] <ContentManagerOnlinePlan> [<CommonParameters>]

The example of the command is:

C:\PS>$plan = Get-DAContentManagerOnlinePlan -Name 'plan1'
C:\PS>$plan.Name = 'plan2'
C:\PS>Update-DAContentManagerOnlinePlan -Plan $plan

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Plan	Required	Specifies a Content Manager online plan object for updating.

Deployment Manager

Run a Deployment Manager Plan

To run an existing Deployment Manager plan, run the command below:

Run-DADeploymentManagerJob [-PlanName] <String> [-WaitForJobComplete] [<CommonParameters>]

The example of the command is:

C:\PS>Run-DADeploymentManagerJob -PlanName 'plan1' -WaitForJobComplete

To configure the command parameters, see the table below.

Parameter	Type	Comment
-PlanName	Required	Specifies the name of an existing Deployment Manager plan that you want to run.
-WaitForJobComplete	Optional	Specifies whether to wait for the completion of the job. If so, the job process will be displayed and another job cannot be run until this job has completed; if not, the job will run in the background and you can run another job without waiting for the completion of this job.

Replicator

Run a Replicator Export Plan

To run an existing Replicator export plan, run the command below:

Run-DAReplicatorExportJob [-PlanName] <String> -ReplicatorRunLevel <ReplicatorRunLevel> [-ReplicateModifications] [-ReplicateDeletions] [-SpecialReferenceTime <String>] [-WaitForJobComplete] [<CommonParameters>]

The examples of the command are:

C:\PS>Run-DAReplicatorExportJob -PlanName 'plan1' -ReplicatorRunLevel Full -WaitForJobComplete
C:\PS>Run-DAReplicatorExportJob -PlanName 'plan2' -ReplicatorRunLevel Incremental -ReplicateModifications -ReplicateDeletions -SpecialReferenceTime 11m

To configure the command parameters, see the table below.

Parameter	Type	Comment
-PlanName	Required	Specifies the name of an existing Replicator export plan that you want to run.
-ReplicatorRunLevel	Required	Specifies the job level. Valid values are “Full” and “Incremental”.
-ReplicateModifications	Optional	Specifies whether to replicate the modifications. You cannot specify this parameter unless the value of parameter 'ReplicatorRunLevel' is 'Incremental.'
-ReplicateDeletions	Optional	Specifies whether to replicate the deletions. You cannot specify this parameter unless the value of parameter 'ReplicatorRunLevel' is 'Incremental.'
-SpecialReferenceTime	Optional	Specifies the reference time. The job will replicate the contents which were created or modified during the specified period just before the job start time.
-WaitForJobComplete	Optional	Specifies whether to wait for the completion of the job. If so, the job process will be displayed and another job cannot be run until this job has completed; if not, the job will run in the background and you can run another job without waiting for the completion of this job.

Run a Replicator Import Plan

To run an existing Replicator import plan, run the command below:

Run-DAReplicatorImportJob [-PlanName] <String> [-WaitForJobComplete] [<CommonParameters>]

The example of the command is:

C:\PS>Run-DAReplicatorImportJob -PlanName 'plan1' -WaitForJobComplete

To configure the command parameters, see the table below.

Parameter	Type	Comment
-PlanName	Required	Specifies the name of an existing Replicator import plan that you want to run.
-WaitForJobComplete	Optional	Specifies whether to wait for the completion of the job. If so, the job process will be displayed and another job cannot be run until this job has completed; if not, the job will run in the background and you can run another job without waiting for the completion of this job.

Run an Online Replicator Plan Using Management Shell

To run an existing online Replicator plan, run the command below:

Run-DAReplicatorOnlineJob [-PlanName] <String> -ReplicatorRunLevel <ReplicatorRunLevel> [-ReplicateModifications] [-ReplicateDeletions] [-SpecialReferenceTime <String>] [-TestRun] [-WaitForJobComplete] [<CommonParameters>]

The examples of the command are:

C:\PS>Run-DAReplicatorOnlineJob -PlanName 'plan1' -ReplicatorRunLevel Full -WaitForJobComplete -TestRun
C:\PS>Run-DAReplicatorOnlineJob -PlanName 'plan2' -ReplicatorRunLevel Incremental -ReplicateModifications -ReplicateDeletions -TestRun

To configure the command parameters, see the table below.

Parameter	Type	Comment
-PlanName	Required	Specifies the name of an existing Replicator plan that you want to run.
-ReplicatorRunLevel	Required	Specifies the job level. Valid values are “Full” and “Incremental”.
-ReplicateModifications	Optional	Specifies whether to replicate the modifications. You cannot specify this parameter unless the value of parameter 'ReplicatorRunLevel' is 'Incremental'.
-ReplicateDeletions	Optional	Specifies whether to replicate the deletions. You cannot specify this parameter unless the value of parameter 'ReplicatorRunLevel' is 'Incremental'.
-SpecialReferenceTime	Optional	Specifies the reference time. The job will replicate the contents which were created or

Parameter	Type	Comment
		modified during the specified period just before the job start time.
-TestRun	Optional	Specifies whether to simulate the execution of the selected plan.
-WaitForJobComplete	Optional	Specifies whether to wait for the completion of the job. If so, the job process will be displayed and another job cannot be run until this job has completed; if not, the job will run in the background and you can run another job without waiting for the completion of this job.

Storage Optimization

BLOB Provider

Obtain a Stub Database's Configuration

To get configuration details of a stub database, run the command below:

Get-DAStubDBConfiguration [-FarmName] <String> [<CommonParameters>]

The example of the command is:

C:\PS>$StubDBConfig = Get-DAStubDBConfiguration -FarmName 'Farm(SP2010_DB:SHAREPOINT_CONFIG)'
C:\PS>$StubDBConfig.WebApplicationStubDBConfigurations

To configure the command parameters, see the table below.

Parameter	Type	Comment
-FarmName	Required	Specifies a farm name.

Configure a Stub Database

To configure a stub database for a farm, Web application, or content database, run the command below:

Configure-DAStubDB [-Configuration] <StubDBConfiguration> [<CommonParameters>]

The example of the command is:

C:\PS>$StubDBConfig = Get-DAStubDBConfiguration -FarmName 'Farm(SP2010_DB:SHAREPOINT_CONFIG)'
C:\PS>$ContentDBs = $StubDBConfig.GetContentDBLevelStubDBConfigurationByName('WSS_Content_10001')
C:\PS>$ContentDB1 = $ContentDBs[0]
C:\PS>$ContentDB1.ConfigureAsWindowsAuthentication ('DBServer','StubDB1')

C:\PS>Configure-DAStubDB -Configuration $ContentDB1
C:\PS>$ContentDB2 = $StubDBConfig.GetContentDBLevelStubDBConfigurationById ("f17d7594-d09a-49a8-b41a-479a877a51d0")
C:\PS>$password = Read-Host -AsSecureString
******
C:\PS>$ContentDB2.ConfigureAsSQLAuthentication ('DBServer','StubDB2','Username',$password)
C:\PS>Configure-DAStubDB -Configuration $ContentDB2

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Configuration	Required	Specifies the stub database configuration. You can specify the farm level stub database configuration, Web application stub database configuration and content database stub database configuration.

Get Information about all BLOB Provider Binaries

To get information about all BLOB Provider binaries, run the command below:

Get-DABLOBProviderBinariesInfo [[-FarmName] <String>] [<CommonParameters>]

The example of the command is:

C:\PS>Get-DABLOBProviderBinariesInfo -FarmName 'Farm(SP2010_DB:SHAREPOINT_CONFIG)'

To configure the command parameters, see the table below.

Parameter	Type	Comment
-FarmName	Optional	Specifies a farm name. If you do not set any value for the parameter, all of the farms' BLOB Provider binaries' information will be returned.

Get the BLOB Provider Configuration Details

To get the configuration details of the BLOB Provider, run the command below:

Get-DABLOBProviderConfiguration [-FarmName] <String> [<CommonParameters>]

The example of the command is:

C:\PS>$BLOB = Get-DABLOBProviderConfiguration -FarmName'Farm(SP2010_DB:SHAREPOINT_CONFIG)'
C:\PS>$BLOB.WebApplicationLevelRBSConfigurations
C:\PS> $BLOB.ContentDBLevelRBSConfigurations

To configure the command parameters, see the table below.

Parameter	Type	Comment
-FarmName	Required	Specifies a farm name.

Update BLOB Provider Settings

To update the BLOB Provider settings, run the command below:

Save-DABLOBProviderConfiguration [-Configuration] <BLOBProviderConfiguration> [-RunNow] [<CommonParameters>]

The example of the command is:

C:\PS>$BLOB = Get-DABLOBProviderConfiguration -FarmName '
'Farm(SP2010_DB:SHAREPOINT_CONFIG)'

C:\PS>$ContentDB1 = $BLOB.GetContentDBLevelRBSConfigurationByName ('WSS_Content_10001')
C:\PS>$ContentDB1.RBSEnabled = $true
C:\PS>$ContentDB2 = $BLOB.GetContentDBLevelRBSConfigurationById ('f17d7594-d09a-49a8-b41a-479a877a51d0')

C:\PS>$ContentDB2.RBSEnabled = $true
C:\PS>$ContentDB1
C:\PS>$ContentDB2
C:\>Save-DABLOBProviderConfiguration -Configuration $BLOB

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Configuration	Required	Specifies the BLOB Provider configuration instance.
-RunNow	Optional	Determines whether to run now after saving the configuration.

Archiver

Get Archiver Index Data Information

To get information on the Archiver Data that has not been successfully indexed, run the command below:

Get-DAArchiverIndexDataInfo [-FarmName] <String> [-OutPut] <String> [-Overwrite] [<CommonParameters>]

The example of the command is:

C:\PS>Get-DAArchiverIndexDataInfo -FarmName FarmA -OutPut C:\IndexInfo.xls

To configure the command parameters, see the table below.

Parameter	Type	Comment
-FarmName	Required	Specifies the name of a farm where the Archiver

		Data's index information will be retrieved from.
-OutPut	Required	Specifies a path to store the output file.
-Overwrite	Optional	Overwrites the file if it already exists at the specified path.

Get an Archiver Job's Detailed Information for Backup

To get detailed information on an Archiver job for backup, run the command below:

Get-DAArchiverJobDetailForBackup [-ID] <String> [-JobReportDetailStatus <JobReportDetailStatus[]>] [-StartIndex <Int32>] [-Length <Int32>] [<CommonParameters>]

The examples of the command are:

C:\PS>Get-DAArchiverJobDetailForBackup -ID AR20120710135406923230S
C:\PS>Get-DAArchiverJobDetailForBackup -ID AR20120710135406923230S -JobReportDetailStatus Success,Failed -StartIndex 0 -Length 1

To configure the command parameters, see the table below.

Parameter	Type	Comment
-ID	Required	Specifies the job ID.
-JobReportDetailStatus	Optional	Specifies the detail status to filter the detail. Valid values are "Success", "Failed", and "Skipped".
-StartIndex	Optional	Specifies the start number of the details that you want to get.
-Length	Optional	Specifies the total number of details in the result.

Get an Archiver Job's Detailed Information for Deletion

To get detailed information on an Archiver job for deletion, run the command below:

Get-DAArchiverJobDetailForDeletion [-ID] <String> [-JobReportDetailStatus <JobReportDetailStatus[]>] [-StartIndex <Int32>] [-Length <Int32>] [<CommonParameters>]

The examples of the command are:

C:\PS>Get-DAArchiverJobDetailForDeletion -ID AR2012071013585564535 6A0

Or
C:\PS>Get-DAArchiverJobDetailForDeletion -ID AR2012071013585564535 6A0 -JobReportDetailStatus Success,Failed -StartIndex 0 -Length 1

To configure the command parameters, see the table below.

Parameter	Type	Comment
-ID	Required	Specifies the job ID.
-JobReportDetailStatus	Optional	Specifies the detail status to filter the detail. Valid values are "Success", "Failed", and "Skipped".

Parameter	Type	Comment
-StartIndex	Optional	Specifies the start number of the details that you want to get.
-Length	Optional	Specifies the total number of details in the result.

Run Full Text Index Job

To run a full text index job, run the command below:

Run-DAArchiverIndexJob -JobId <String> [<CommonParameters>]

The example of the command:

C:\PS>Run-DAArchiverIndexJob -JobId AI20120710135406923230

To configure the command parameters, see the table below.

Parameter	Type	Comment
-JobId	Required	Specifies an Archiver index job ID.

Archive SharePoint Data

To archive the specified SharePoint data, run the command below:

Run-DAArchiverJob [-SPObject] <SharePointObject> [-WaitForJobComplete] [<CommonParameters>]

The example of the command：

C:\PS>$site1 = New-Object DocAve.API.Objects.SharePointSite ('Farm(SP2010_DB:SHAREPOINT_CONFIG)','http://SharePoint2010:10001/sites/site1')
C:\PS>Run-DAArchiverJob -SPObject $site1

To configure the command parameters, see the table below.

Parameter	Type	Comment
-SPObject	Required	The SharePoint object to be archived.
-WaitForJobComplete	Optional	Specifies whether to wait for the completion of the job. If so, the job process will be displayed and another job cannot be run until this job has completed; if not, the job will run in the background and you can run another job without waiting for the completion of this job.

Set the Archiver Data Index Information

To set the index information of Archiver data that has not been successfully indexed, run the command below:

Set-DAArchiverIndexDataInfo [-Path] <String> [-OutPut] <String> [-Overwrite] [<CommonParameters>]

The example of the command is:

C:\PS>Set-DAArchiverIndexDataInfo -Path C:\IndexInfo.xls -OutPut C:\Result .xls

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Path	Required	Specifies a path to load the file storing the Archiver data's index information.
-OutPut	Required	Specifies a path to store the output file.
-Overwrite	Optional	Overwrites the file if it already exists at the specified path.

Real-Time Storage Manager

Get a Blank Realtime Storage Manager Rule

To get a blank Real-time Storage Manager rule, run the command below:

Get-DARealtimeStorageManagerBlankRule [<CommonParameters>]

The example of the command is:

C:\PS>$rule = Get-DARealtimeStorageManagerBlankRule

Get a Real-Time Storage Manager Rule

To get the configurations of a Real-Time Storage Manager rule, run the command below:

Get-DARealtimeStorageManagerRule [-Name <String>] [<CommonParameters>]

Get-DARealtimeStorageManagerRule [-SPContentDB <SharePointContentDB>] [<CommonParameters>]

The examples of the command are:

C:\PS>Get-DARealtimeStorageManagerRule
C:\PS>Get-DARealtimeStorageManagerRule -Name 'rule2'
C:\PS>$db = New-Object DocAve.API.Objects.SharePointContentDB
'Farm(SP2010_DB:SHAREPOINT_CONFIG)','http://SharePoint2010:10001/', 'wss_content_10001'
C:\PS>Get-DARealtimeStorageManagerRule -SPContentDB

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Name	Optional	Specifies the name of the Real-Time Storage Manager rule whose configurations will be obtained.

Parameter	Type	Comment
-SPContentDB	Optional	Specifies a SharePoint content database. All Real-Time Storage Manager rules that have been applied to this specified database will be obtained.
-SPWebApplication	Optional	Specifies a SharePoint Web application. All Real-Time Storage Manager rules that have been applied to this specified Web application will be obtained.
-SPSite	Optional	Specifies a SharePoint site collection. All Realtime Storage Manager rules that have been applied to this specified site collection will be obtained.

Create a Real-Time Storage Manager Rule

To create a Real-Time Storage Manager rule, run the command below:

New-DARealtimeStorageManagerRule [-Rule] <RealtimeStorageManagerRule> [<CommonParameters>]

The example of the command is:

C:\PS>$rule = Get-DARealtimeStorageManagerBlankRule
C:\PS>$rule. Name = 'rule1'
C:\PS>$rule.SizeThreshold = '768kb'
C:\PS>$rule.LogicalDeviceName = 'logical device1'
C:\PS>$rule.CompressionLevel = 'level3'
C:\PS>$rule.EnableDataEncryption('MyProfile')
C:\PS>New-DARealtimeStorageManagerRule $rule

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Rule (or $rule)	Required	Specifies a real-time storage manager rule object where you can create a new real-time storage manager rule.

Retract a Real-Time Storage Manager Rule

To retract an existing Real-Time Storage Manager rule from the specified SharePoint content database, Web application, or site collection, run the command below:

Retract-DARealtimeStorageManagerRule [-RuleName] <String> [-SPContentDB] <SharePointContentDB> [<CommonParameters>]

The example of the command:

C:\PS>$db = New-Object DocAve.API.Objects.SharePointContentDB 'Farm(SP2010_DB:SHAREPOINT_CONFIG)','http://SharePoint2010:10001/', 'wss_content_10001'
C:\PS>Retract-DARealtimeStorageManagerRule -RuleName 'rule1' -SPContentDB $db

To configure the command parameters, see the table below.

Parameter	Type	Comment
-RuleName	Required	Specifies the name of the Realtime Storage Manager rule that you want to retract from the specified SharePoint content database.
-SPContentDB	Required	Specifies a SharePoint content database where the specified Realtime Storage Manager rule will be retracted.
-SPWebApplication	Optional	Specifies a SharePoint Web application where the specified Realtime Storage Manager rule will be retracted.
-SPSite	Optional	Specifies a SharePoint site collection where the specified Realtime Storage Manager rule will be retracted.

Reuse a Realtime Storage Manager Rule

To reuse an existing Realtime Storage Manager rule, run the command below:

Reuse-DARealtimeStorageManagerRule [-RuleName] <String> [-SPContentDB] <SharePointContentDB> [<CommonParameters>]

The example of the command:

C:\PS>$db = New-Object DocAve.API.Objects.SharePointContentDB 'Farm(SP2010_DB:SHAREPOINT_CONFIG)','http://SharePoint2010:10001/', 'wss_content_10001'
C:\PS>Reuse-DARealtimeStorageManagerRule -RuleName 'rule1' -SPContentDB $db

To configure the command parameters, see the table below.

Parameter	Type	Comment
-RuleName	Required	Specifies the name of the Realtime Storage Manager rule that you want to reuse.
-SPContentDB	Required	Specifies a SharePoint content database where the specified Realtime Storage Manager rule will be applied.
-SPWebApplication	Optional	Specifies a SharePoint Web application where the specified Realtime Storage Manager rule will be applied.
-SPSite	Optional	Specifies a SharePoint site collection where the specified Realtime Storage Manager rule will be applied.

Delete a Real-Time Storage Manager Rule

To delete a Real-Time Storage Manager rule, run the command below:

Remove-DARealtimeStorageManagerRule [-Name] <String> [<CommonParameters>]

The example of the command is:

C:\PS>Remove-DARealtimeStorageManagerRule -Name 'rule1'

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Name	Required	Specifies the name of a Real-Time Storage Manager rule that you want to delete.

Update the Configurations of a Real-Time Storage Manager Rule

To update the configurations of an existing Real-time Storage Manager rule, run the command below:

Update-DARealtimeStorageManagerRule [-Rule] <RealtimeStorageManagerRule> [<CommonParameters>]

The example of the command is:

C:\PS>$rule = Get-DARealtimeStorageManagerRule -Name 'rule1'
C:\PS>$rule.Name = 'rule2'
C:\PS>$rule.CompressionLevel = 'level7'
C:\PS>Update-DARealtimeStorageManagerRule $rule

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Rule (or $rule)	Required	Specifies a Real-Time Storage Manager rule object whose configurations will be updated.

Job Monitor

Obtain a Specified Module's Job

To get a job of a specified module, run the command below:

Get-DAJob [-ID] <String> [<CommonParameters>]

Get-DAJob [-Module] <Module> [-StartTimeRangeBeginning <DateTime>] [-StartTimeRangeEnding <DateTime>] [<CommonParameters>]

The examples of the command are:

C:\PS>Get-DAJob -Id CA20120710105416471063
Or
C:\PS>Get-DAJob -Module Administrator
Or
C:\PS>Get-DAJob -Module Administrator -StartTimeRangeBeginning "7/10/2012 10:10" -StartTimeRangeEnding "7/10/2012 10:58"

To configure the command parameters, see the table below.

Parameter	Type	Comment
-ID	Required	Specifies the job ID.
-Module	Required	Specifies the job's module. Valid values are “GranularBackup”, “GranularRestore”, "PlatformBackup", "PlatformRestore", "PlatformMaintenanceManager", "Retention", "FarmRebuild", "PlatformIndexMigration", "Administrator", "ContentManager", "DeploymentManager", "Replicator", "Archiver", "ArchiverFullTextIndex", "ArchiverRestore", "ArchiverRetention", "ConvertStubToContent", "ScheduledStorageManager", "Connector", "OrphanBLOBRetention", "ReportCenter", "LogManager", "JobPruning", "FileSystemMigration", "SharePoint2007To2010Migration", "SharePoint2007To2013Migration", "SharePoint2010To2013Migration", "ERoomMigration", "LivelinkMigration", "ExchangePublicFolderMigraton", "LotusNotesMigration", "EndUserArchiver", "EndUserRestore", "DataManager", "eDiscovery" and "Vault".
-StartTimeRangeBeginning	Optional	Specifies the beginning of job start time range. The type must be a valid DateTime format that is culture-specific to the administrative language, such as 2/16/2010 12:15:12 for English-US. If you do not specify the time range, the job information will not be filtered according to job start time.
-StartTimeRangeEnding	Optional	Specifies the end of job start time range. The type must be a valid DateTime format that is culture-specific to the administrative language, such as 2/16/2010 12:15:12 for English-US. If you do not specify the time range, it will

Parameter	Type	Comment
		not filter the job information according to job start time.

Obtain a Job's Detailed Information

To get detailed information on the items included in a job (such as item type, source URL, destination URL, etc.), run the command below:

Get-DAJobDetail [-ID] <String> [-JobReportDetailStatus <JobReportDetailStatus[]>] [-StartIndex <Int32>] [-Length <Int32>] [<CommonParameters>]

The examples of the command are:

C:\PS>Get-DAJobDetail -ID CM20111011115232293242
C:\PS>Get-DAJobDetail -ID CM20111011115233293432 -JobReportDetailStatus Success,Failed -StartIndex 0 -Length 1

To configure the command parameters, see the table below.

Parameter	Type	Comment
-ID	Required	Specifies the job ID.
-JobReportDetailStatus	Optional	Specifies the detail status to filter the detail. Valid values are "Success", "Failed", and "Skipped". If the job is a Deployment Manager job, this parameter is ignored.
-StartIndex	Optional	Specifies the start number of the details. If the job is a Deployment Manager job, this parameter is ignored.
-Length	Optional	Specifies the total number of details in the result. If the job is a Deployment Manager job, this parameter is ignored.

Obtain a Job's Summary Information

To get a job summary, run the command below:

Get-DAJobSummary [-ID] <String> [<CommonParameters>]

The example of the command is:

C:\PS>Get-DAJobSummary -ID CA20120710105416471063

To configure the command parameters, see the table below.

Parameter	Type	Comment
-ID	Required	Specifies the job ID.

Download a Job Report

To download a job report, run the command below:

Download-DAJobReport [-ID] <String> [-Format <JobReportFormat>] -Path <String> [-Overwrite] [<CommonParameters>]

The example of the command is:

C:\PS>Download-DAJobReport -ID FB20120627174730343482 -Path 'C:\Report.zip'-Format txt -Overwrite

To configure the command parameters, see the table below.

Parameter	Type	Comment
-ID	Required	Specifies the job ID.
-Format	Optional	Specifies the format of the report. Valid values are “XLS”, “CSV”, and “TXT”. If the job is a Log Manager job, this parameter is ignored.
-Path	Required	Specifies the path of the report file.
-Overwrite	Optional	Allows the cmdlet to overwrite an existing file.

Migration

General SharePoint Migration Commands

Get the Configuration Details of a SharePoint Migration Job

To get the configuration details of the specified SharePoint online migration job, run the command below:

Get-DASPMigrationJobConfigurationDetail [-ID] <String> [-JobReportDetailStatus <JobReportDetailStatus[]>] [-StartIndex <Int32>] [-Length <Int32>] [<CommonParameters>]

The examples of the command are:

C:\PS>Get-DASPMigrationJobConfigurationDetail -ID MD20120601151648823941

C:\PS>Get-DASPMigrationJobConfigurationDetail -ID MD20120601151648823941 -JobReportDetailStatus Success -StartIndex0 -Length 1

To configure the command parameters, see the table below.

Parameter	Type	Comment
-ID	Required	Specifies a Job ID to get the detailed job configurations.

Parameter	Type	Comment
-JobReportDetailStatus	Optional	Specifies a status filter to only display the configuration details of the specified status. The valid values are "Success", "Failed", "Skipped", and "Filtered".
-StartIndex	Optional	Specifies a number to be used when displaying the configuration details of the specified status. After the detailed information is filtered according to the status specified in the "JobReportDetailStatus" parameter, the configuration details will be displayed starting from the detailed information ranked with this number.
-Length	Optional	Specifies the number of the configuration details to be displayed.

Get the Permission Details of a SharePoint Migration Job

To get the detailed information of the objects in a SharePoint online migration job, run the command below:

Get-DASPMigrationJobObjectsDetail [-ID] <String> [-JobReportDetailStatus <JobReportDetailStatus[]>] [-StartIndex <Int32>] [-Length <Int32>] [<CommonParameters>]

The examples of the command are:

C:\PS>Get-DASPMigrationJobPermissionDetail -ID MD20120601151648823941

C:\PS>Get-DASPMigrationJobPermissionDetail -ID MD20120601151648823941-JobReportDetailStatus Success -StartIndex 0 -Length 1

To configure the command parameters, see the table below.

Parameter	Type	Comment
-ID	Required	Specifies a Job ID to get the detailed permission configurations.
-JobReportDetailStatus	Optional	Specifies a status filter to only display the permission details of the specified status. The valid values are "Success", "Failed", "Skipped" and "Filtered".

Parameter	Type	Comment
-StartIndex	Optional	Specifies a number to be used when displaying the permission details of the specified status. After the detailed information is filtered according to the status specified in the "JobReportDetailStatus" parameter, the configuration details will be displayed starting from the detailed information ranked with this number.
-Length	Optional	Specifies the number of the permission details to be displayed.

Get the Object Details of a SharePoint Migration Job

To get the detailed information of the objects in a SharePoint online migration job, run the command below:

Get-DASPMigrationJobObjectsDetail [-ID] <String> [-JobReportDetailStatus <JobReportDetailStatus[]>] [-StartIndex <Int32>] [-Length <Int32>] [<CommonParameters>]

The examples of the command are:

C:\PS>Get-DASPMigrationJobObjectsDetail -ID MD20120601151648823941

C:\PS>Get-DASPMigrationJobObjectsDetail -ID MD20120601151648823941-JobReportDetailStatus Success -StartIndex 0 -Length 1

To configure the command parameters, see the table below.

Parameter	Type	Comment
-ID	Required	Specifies a job ID to get the detailed object information.
-JobReportDetailStatus	Optional	Specifies a status filter to only display the object details of the specified status. The valid values are "Success", "Failed", "Skipped" and "Filtered".

Parameter	Type	Comment
-StartIndex	Optional	Specifies a number to be used when displaying the detailed object information of the specified status. After the detailed information is filtered according to the status specified in the "JobReportDetailStatus" parameter, the object details will be displayed starting from the detailed information ranked with this number.
-Length	Optional	Specifies the number of object details to be displayed.

SharePoint 2007 to 2010 Migration

Obtain a Blank SharePoint 2007 to 2010 Online Migration Plan

To obtain a blank SharePoint 2007 to 2010 online Migration plan, run the command below:

Get-DASP07To10MigrationBlankOnlinePlan [<CommonParameters>]

The example of the command is:

C:\PS>$plan = Get-DASP07To10MigrationBlankOnlinePlan

Obtain a SharePoint 2007 to 2010 Online Migration Plan

To obtain a SharePoint 2007 to 2010 online Migration plan, run the command below:

Get-DASP07To10MigrationOnlinePlan [[-Name] <String[]>] [<CommonParameters>]

The example of the command is:

C:\PS>Get-DASP07To10MigrationOnlinePlan -Name 'plan1'

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Name	Optional	Specifies the name(s) of the existing SharePoint 2007 to 2010 online Migration plan(s) you want to obtain. Wildcards and multiple values are allowed, which should be separated by commas. If no values are entered here, all information of the existing SharePoint 2007 to 2010 online Migration plan(s) will be obtained.

Create a SharePoint 2007 to 2010 Online Migration Plan

To create a new SharePoint 2007 to 2010 online Migration plan, run the command below:

New-DASP07To10MigrationOnlinePlan [-Plan] <SP07To10MigrationOnlinePlan> [<CommonParameters>]

The example of the command is:

C:\PS>$plan = Get-DASP07To10MigrationBlankOnlinePlan C:\PS>$plan.Name = 'plan1'
C:\PS>$plan.Description = 'Description'
C:\PS>$plan.SourceFarmName = 'Farm(SP2007_DB:SHAREPOINT_CONFIG)'
C:\PS>$plan.DestinationFarmName = 'Farm(SP2010_DB:SHAREPOINT_CONFIG)'
C:\PS>$plan.SourceAgentGroup = 'AgentGroup07'
C:\PS>$plan.DestinationAgentGroup = 'AgentGroup10'
C:\PS>$plan.ProfileName = 'Default Profile'
C:\PS>$list07 = New-Object DocAve.API.Objects.Sharepointlist('Farm(SP2007_DB:SHAREPOINT_CONFIG)', 'http://SharePoint2007:10001/sites/site1','sites/site1', 'list1')
C:\PS>$plan.SourceTree.IncludeSPList($list07, $true, $false)
C:\PS>$list10 = New-Object DocAve.API.Objects.Sharepointlist('Farm(SP2007_DB:SHAREPOINT_CONFIG)', 'http://SharePoint2007:10001/sites/site1','sites/site1', 'list1')
C:\PS>$plan.DestinationTree.SelectSPObject($list10)
C:\PS>$plan.Action = 'Merge' C:\PS>$plan.AssociatedPlanGroup.Add('plan group1')
C:\PS>New-DASP07To10MigrationOnlinePlan $plan

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Plan (or $myPlan)	Required	Specifies a SharePoint 2007 to 2010 Migration online plan object.

Run a SharePoint 2007 to 2010 Online Migration Job

To run an existing SharePoint 2007 to 2010 online Migration plan, run the command below:

Run-DASP07To10MigrationOnlineJob [-PlanName] <String> -Options <String> [-IncrementalType <String>] -ConflictResolution <SP07To10MigrationConflictResolution> -JobReportType <String> [-Description <String>] [-TestRun] [-WaitForJobComplete] [<CommonParameters>]

The examples of the command are:

C:\PS>Run-DASP07To10MigrationOnlineJob -PlanName 'plan1' -Options Full -ConflictResolution Skip -JobReportType Detailed -Description description -WaitForJobComplete

C:\PS>Run-DASP07To10MigrationOnlineJob -PlanName 'plan1' -Options Full -ConflictResolution Skip -JobReportType Detailed -Description description -WaitForJobComplete

To configure the command parameters, see the table below.

Parameter	Type	Comment
-PlanName	Required	Specifies the name of an existing SharePoint 2007 to 2010 online Migration plan that you want to run.
-Options	Required	Specifies to run a Full or an Incremental job of the selected Migration plan. Valid values are “Full” and “Incremental”.
-IncrementalType	Optional	Specifies a schedule to be used for an Incremental Migration job. Enters the parameter in the format of *Minute *Hour *Day *Week *Month. The * is any number between 1 and 2147483647 followed by a time unit. You cannot specify this parameter unless the value of parameter 'Options' is 'Incremental.'
-ConflictResolution	Required	Specifies a conflict resolution to handle conflicts occurred during Migration. The valid values are “Skip”, “SkipAndSkip”, “SkipAndOverwrite”, “SkipAndAppend”, “MergeAndSkip”, “MergeAndOverwrite”, and “MergeAndAppend”. For detailed information on each of these values, refer to the note below this table.
-JobReportType	Optional	Specifies the job report level. The valid values are “Simplified” and “Detailed”.
-Description	Optional	Specifies the description of the SharePoint 2007 to 2010 online Migration plan that you want to run.
-TestRun	Optional	Specifies whether to simulate the execution of the selected plan.
-WaitForJobComplete	Optional	Specifies whether to wait for the completion of the job. If so, the job process will be displayed and another job cannot be run until this job has completed; if not, the job will run in the background and you can run another job without waiting for the completion of this job.

***Note**: The following information related to the values in the **-ConflictResolution** parameter.

- **Skip** – Skips the conflicted containers in a migration job when conflicts occur.
- **SkipAndSkip** – Skips the conflicted containers and all lower-level objects in a migration job when conflicts occur.
- **SkipAndOverwrite** – Skips the conflicted containers and overwrites all lower-level objects in a migration job when conflicts occur.
- **SkipAndAppend** – Skips the conflicted containers and appends all lower-level objects in a migration job when conflicts occur.

- **MergeAndSkip** – Merges the conflicted containers and skips all lower-level objects in a migration job when conflicts occur.
- **MergeAndOverwrite** – Merges the conflicted containers and overwrites all lower-level objects in a migration job when conflicts occur.
- **MergeAndAppend** – Merges the conflicted containers and appends all lower-level objects in a migration job when conflicts occur.

Delete a SharePoint 2007 to 2010 Online Migration Plan

To delete a SharePoint 2007 to 2010 online Migration plan, run the command below:

Remove-DASP07To10MigrationOnlinePlan [-Name] <String> [-WhatIf] [-Confirm] [<CommonParameters>]

The example of the command is:

C:\PS>Remove-DASP07To10MigrationOnlinePlan -Name 'plan1'

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Name	Required	Specifies the name of a SharePoint 2007 to 2010 online Migration plan that you want to delete.
-WhatIf	Optional	Describes what would happen if you executed the command.
-Confirm	Optional	Prompts you for confirmation before executing the command.

Upload a SharePoint 2007 to 2010 Online Migration Profile

To upload a SharePoint 2007 to 2010 online Migration profile, run the command below:

Upload-DASP07To10MigrationOnlineProfile [-Path] <String> [-Overwrite] [<CommonParameters>]

The example of the command is:

C:\PS>Upload-DASP07To10MigrationOnlineProfile -Path 'C:\SP07To10MigrationOnlineProfile1.xml' -Overwrite

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Path	Required	Specifies the path of the XML file.

Parameter	Type	Comment
-Overwrite	Optional	Allows the cmdlet to overwrite an existing profile.

Download a SharePoint 2007 to 2010 Online Migration Profile

To download a SharePoint 2007 to 2010 online Migration profile, run the command below:

Download-DASP07To10MigrationOnlineProfile [-Name] <String> [-Path] <String> [-Overwrite] [<CommonParameters>]

The example of the command is:

C:\PS>Download-DASP07To10MigrationOnlineProfile -Name 'SP07To10MigrationOnlineProfile1' -Path 'C:\SP07To10MigrationOnlineProfile1.xml' -Overwrite

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Name	Required	Specifies the profile name.
-Path	Required	Specifies the path to store the downloaded XML file.
-Overwrite	Optional	Allows the cmdlet to overwrite an existing file.

SharePoint 2007 to 2013 Migration

Get-DASP07To13MigrationBlankOnlinePlan

To get a blank SharePoint 2007 to 2013 online migration plan, run the command below:

Get-DASP07To13MigrationBlankOnlinePlan [<CommonParameters>]

The example of the command is:

C:\PS>$plan = Get-DASP07To13MigrationBlankOnlinePlan

C:\PS>$plan

Get the Plan Settings of a SharePoint 2007 to 2013 Online Migration Plan

To get the plan settings of a SharePoint 2007 to 2013 online migration plan, run the command below:

Get-DASP07To13MigrationOnlinePlan [[-Name] <String[]>] [<CommonParameters>]

The example of the command is:

C:\PS>Get-DASP07To13MigrationOnlinePlan -Name 'plan1'

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Name	Optional	Specifies the name(s) of the existing SharePoint 2007 to 2013 online migration plan(s) you want to get information from. Wildcards and multiple values are allowed, which are separated by commas. If no values are entered here, the information of all of the existing SharePoint 2007 to 2013 online migration plan(s) will be obtained.

Create a New SharePoint 2007 to 2013 Online Migration Plan

To create a new SharePoint 2007 to 2013 online migration plan, run the command below:

New-DASP07To13MigrationOnlinePlan [-Plan] <SP07To13MigrationOnlinePlan> [<CommonParameters>]

The example of the command is:

C:\PS>$plan = Get-DASP07To13MigrationBlankOnlinePlan

C:\PS>$plan.Name = 'plan1'

C:\PS>$plan.Description = 'Description'

C:\PS>$plan.SourceFarmName = 'Farm(SP2007_DB:SHAREPOINT_CONFIG)'

C:\PS>$plan.DestinationFarmName = 'Farm(SP2013_DB:SHAREPOINT_CONFIG)'

C:\PS>$plan.SourceAgentGroupName = 'AgentGroup07'

C:\PS>$plan.DestinationAgentGroupName = 'AgentGroup13'

C:\PS>$plan.ProfileName = 'Default Profile'

C:\PS>$list07 = New-Object DocAve.API.Objects.Sharepointlist('Farm(SP2007_DB:SHAREPOINT_CONFIG)', 'http://SharePoint2007:10001/sites/site1','sites/site1', 'list1')

C:\PS>$plan.SourceTree.IncludeSPList($list07, $true, $false)

C:\PS>$list13 = New-Object DocAve.API.Objects.Sharepointlist('Farm(SP2013_DB:SHAREPOINT_CONFIG)', 'http://SharePoint2013:10001/sites/site1','sites/site1', 'list1')

C:\PS>$plan.DestinationTree.SelectSPObject($list13)

C:\PS>$plan.Action = 'Merge'

C:\PS>$plan.AssociatedPlanGroup.Add('plan group1')

C:\PS>$plan.UseSpecifiedMigrationDatabaseToStoreTheJobData = $true

C:\PS>New-DASP07To13MigrationOnlinePlan $plan

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Plan	Required	Specifies a SharePoint 2007 to 2013 Migration online plan object. About the structure and attribute of SP07To13MigrationOnlinePlan, please refer to the SDK documentation for more information.

Delete a SharePoint 2007 to 2013 Online Migration Plan

To delete a SharePoint 2007 to 2013 online migration plan, run the command below:

Remove-DASP07To13MigrationOnlinePlan [-Name] <String> [-WhatIf] [-Confirm] [<CommonParameters>]

The example of the command is:

C:\PS>Remove-DASP07To13MigrationOnlinePlan -Name 'plan1'

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Name	Required	Specifies the name of a SharePoint 2007 to 2013 online migration plan that you want to delete.
-WhatIf	Optional	Describes what would happen if you executed the command without actually executing the command.
-Confirm	Optional	Prompts you for confirmation before executing the command.

Run a SharePoint 2007 to 2013 Online Migration Job

To perform a SharePoint 2007 to 2013 online migration job, run the command below:

Run-DASP07To13MigrationOnlineJob [-PlanName] <String> -Options <String> [-IncrementalType <String>] -ConflictResolution <SP07To13MigrationConflictResolution> -JobReportType <String> [-TestRun] [-WaitForJobComplete] [<CommonParameters>]

The example of the command is:

C:\PS>Run-DASP07To13MigrationOnlineJob -PlanName 'plan1' -Options Full -ConflictResolution Skip -JobReportType Detailed -WaitForJobComplete

To configure the command parameters, see the table below.

Parameter	Type	Comment
-PlanName	Required	Specifies the name of an existing SharePoint 2007 to 2013 online migration plan that you want to run.
-Options	Required	Specifies to run a full or an incremental migration job of the selected migration plan. The valid values are "Full" and "Incremental".
-IncrementalType	Optional	Specifies a time range for an incremental migration job to only migrate the changes in the specified time period. The format is *Minute, *Hour, *Day, *Week, or *Month. Enter any number between 1 and 2147483647 followed by a time unit. This parameter cannot be specified unless the value of the Option parameter is "Incremental". You cannot specify this parameter unless the value of parameter 'Options' is 'Incremental'.
-ConflictResolution	Required	Specifies a conflict resolution to handle conflicts occurred during migration. The valid values are "Skip", "SkipAndSkip", "SkipAndOverwrite", "SkipAndAppend", "MergeAndSkip", "MergeAndOverwrite" and "MergeAndAppend". For detailed information on each of these values, refer to the note below this table.
-JobReportType	Required	Specifies the job report level. The valid values are "Simplified" and "Detailed".

Parameter	Type	Comment
-TestRun	Optional	Specifies whether to simulate the execution of the selected plan.
-WaitForJobComplete	Optional	Specifies whether to wait for the completion of the job. If so, the job process will be displayed and another job can be run until this job has completed; if not, the job will run in the background and you can run another job without waiting for the completion of this job.

***Note：** The following information related to the values in the **-ConflictResolution** parameter.

- **Skip** – Skips the conflicted containers in a migration job when conflicts occur.
- **SkipAndSkip** – Skips the conflicted containers and all lower-level objects in a migration job when conflicts occur.
- **SkipAndOverwrite** – Skips the conflicted containers and overwrites all lower-level objects in a migration job when conflicts occur.
- **SkipAndAppend** – Skips the conflicted containers and appends all lower-level objects in a migration job when conflicts occur.
- **MergeAndSkip** – Merges the conflicted containers and skips all lower-level objects in a migration job when conflicts occur.
- **MergeAndOverwrite** – Merges the conflicted containers and overwrites all lower-level objects in a migration job when conflicts occur.
- **MergeAndAppend** – Merges the conflicted containers and appends all lower-level objects in a migration job when conflicts occur.

Update a SharePoint 2007 to 2013 Online Migration Plan

To update a SharePoint 2007 to 2013 online migration plan, run the command below:

Update-DASP07To13MigrationOnlinePlan [-Plan] <SP07To13MigrationOnlinePlan> [<CommonParameters>]

The example of the command is:

C:\PS>$plan = Get-DASP07To13MigrationOnlinePlan -Name 'plan1'

C:\PS>$plan.Name = 'plan2'

C:\PS>$plan.Description = 'New Description'

C:\PS>$plan.SourceFarmName = 'Farm(SP2007_DB:SHAREPOINT_CONFIG)'

C:\PS>$plan.DestinationFarmName = 'Farm(SP2013_DB:SHAREPOINT_CONFIG)'

C:\PS>$plan.SourceAgentGroup = 'AgentGroup07'

C:\PS>$plan.DestinationAgentGroup = 'AgentGroup13'

C:\PS>$plan.ProfileName = 'Default Profile'

C:\PS>$list07 = New-Object DocAve.API.Objects.Sharepointlist('Farm(SP2007_DB:SHAREPOINT_CONFIG)', 'http://SharePoint2007:10001/sites/site1','sites/site1', 'list1')

C:\PS>$plan.SourceTree.IncludeSPList($list07, $true, $false)

C:\PS>$list13 = New-Object DocAve.API.Objects.Sharepointlist('Farm(SP2013_DB:SHAREPOINT_CONFIG)', 'http://SharePoint2013:10001/sites/site1','sites/site1', 'list1')

C:\PS>$plan.DestinationTree.SelectSPObject($list13)

C:\PS>$plan.Action = 'Merge'

C:\PS>$plan.AssociatedPlanGroup.Add('plan group1')

C:\PS>$plan.UseSpecifiedMigrationDatabaseToStoreTheJobData = $true

C:\PS>Update-DASP07To13MigrationOnlinePlan -Plan $plan

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Plan	Required	Specifies a plan name to get an existing SharePoint 2007 to 2013 online migration plan for editing.

Upload a SharePoint 2007 to 2013 Online Migration Profile

To upload a SharePoint 2007 to 2013 online migration profile, run the command below:

Upload-DASP07To13MigrationOnlineProfile [-Path] <String> [-Overwrite] [<CommonParameters>]

The example of the command is:

C:\PS>Upload-DASP07To13MigrationOnlineProfile -Path 'C:\SP07To13MigrationOnlineProfile1.xml' -Overwrite

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Path	Required	Specifies the path of the XML file.
-Overwrite	Optional	Allows the cmdlet to overwrite an existing profile.

Download-DASP07To13MigrationOnlineProfile

To download a SharePoint 2007 to 2010 online migration plan, run the command below:

Download-DASP07To13MigrationOnlineProfile [-Name] <String> [-Path] <String> [-Overwrite] [<CommonParameters>]

The example of the command is:

C:\PS>Download-DASP07To13MigrationOnlineProfile -Name 'SP07To13MigrationOnlineProfile1' -Path 'C:\SP07To13MigrationOnlineProfile1.xml' -Overwrite

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Name	Required	Specifies the profile name.
-Path	Required	Specifies the path to store the downloaded XML file.
-Overwrite	Optional	Allows the cmdlet to overwrite an existing file.

SharePoint 2010 to 2013 Migration

Get a Blank SharePoint 2010 to 2013 Online Migration Plan

To get a blank SharePoint 2010 to 2013 online migration plan, run the command below:

Get-DASP10To13MigrationBlankOnlinePlan [<CommonParameters>]

The example of the command is:

C:\PS>$plan = Get-DASP10To13MigrationBlankOnlinePlan

Get the Plan Settings of a SharePoint 2010 to 2013 Online Migration Plan

To get the plan settings of a SharePoint 2010 to 2013 online migration plan, run the command below:

The Get-DASP10To13MigrationOnlinePlan cmdlet gets the configurations of the specified SharePoint 2010 to 2013 online migration plan(s).

The example of the command is:

C:\PS>Get-DASP10To13MigrationOnlinePlan -Name 'plan1'

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Name	Optional	Specifies the name(s) of the existing SharePoint 2010 to 2013 online migration plan(s) you want to get information from. Wildcards and multiple values are allowed, which are separated by commas. If no values are entered here, the information of all of the existing SharePoint 2010 to 2013 online migration plan(s) will be obtained.

Create a SharePoint 2010 to 2013 Online Migration Plan

To create a SharePoint 2010 to 2013 online migration plan, run the command below:

New-DASP10To13MigrationOnlinePlan [-Plan] <SP10To13MigrationOnlinePlan> [<CommonParameters>]

The example of the command is:

C:\PS>$plan = Get-DASP10To13MigrationBlankOnlinePlan

C:\PS>$plan.Name = 'plan1'

C:\PS>$plan.Description = 'Description'

C:\PS>$plan.SourceFarmName = 'Farm(SP2010_DB:SHAREPOINT_CONFIG)'

C:\PS>$plan.DestinationFarmName = 'Farm(SP2013_DB:SHAREPOINT_CONFIG)'

C:\PS>$plan.SourceAgentGroupName = 'AgentGroup10'

C:\PS>$plan.DestinationAgentGroupName = 'AgentGroup13'

C:\PS>$plan.ProfileName = 'Default Profile'

C:\PS>$list10 = New-Object DocAve.API.Objects.Sharepointlist('Farm(SP2010_DB:SHAREPOINT_CONFIG)', 'http://SharePoint2010:10001/sites/site1','sites/site1', 'list1')

C:\PS>$plan.SourceTree.IncludeSPList($list10, $true, $false)

C:\PS>$list13 = New-Object DocAve.API.Objects.Sharepointlist('Farm(SP2013_DB:SHAREPOINT_CONFIG)', 'http://SharePoint2013:10001/sites/site1','sites/site1', 'list1')

C:\PS>$plan.DestinationTree.SelectSPObject($list13)

C:\PS>$plan.Action = 'Merge'

C:\PS>$plan.AssociatedPlanGroup.Add('plan group1')

C:\PS>$plan.UseSpecifiedMigrationDatabaseToStoreTheJobData = $true

C:\PS>New-DASP10To13MigrationOnlinePlan $plan

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Plan	Required	Specifies a SharePoint 2010 to 2013 Migration online plan object. About the structure and attribute of SP10To13MigrationOnlinePlan, please refer to the SDK documentation for more information.

Delete a SharePoint 2010 to 2013 Online Migration Plan

To delete a SharePoint 2010 to 2013 online migration plan, run the command below:

Remove-DASP10To13MigrationOnlinePlan [-Name] <String> [-WhatIf] [-Confirm] [<CommonParameters>]

The example of the command is:

C:\PS>Remove-DASP10To13MigrationOnlinePlan -Name 'plan1'

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Name	Required	Specifies the name of a SharePoint 2010 to 2013 online migration plan that you want to delete.
-WhatIf	Optional	Describes what would happen if you executed the command without actually executing the command.
-Confirm	Optional	Prompts you for confirmation before executing the command.

Run a SharePoint 2010 to 2013 Online Migration Job

To run a SharePoint 2010 to 2013 online migration job, run the command below:

Run-DASP10To13MigrationOnlineJob [-PlanName] <String> -Options <String> [-IncrementalType <String>] -ConflictResolution <SP10To13MigrationConflictResolution> -JobReportType <String> [-TestRun] [-WaitForJobComplete] [<CommonParameters>]

The example of the command is:

C:\PS>Run-DASP10To13MigrationOnlineJob -PlanName 'plan1' -Options Full -ConflictResolution Skip -JobReportType Detailed -WaitForJobComplete

To configure the command parameters, see the table below.

Parameter	Type	Comment
-PlanName	Required	Specifies the name of an existing SharePoint 2010 to 2013 online migration plan that you want to run.
-Options	Required	Specifies to run a full or an incremental migration job of the selected migration plan. The valid values are "Full" and "Incremental".
-IncrementalType	Optional	Specifies a time range for an incremental migration job to only migrate the changes in the specified time period. The format is *Minute, *Hour, *Day, *Week, or *Month. Enter any number between 1 and 2147483647 followed by a time unit. This parameter cannot be specified unless the value of the Option parameter is "Incremental". You cannot specify this parameter unless the value of parameter 'Options' is 'Incremental'.
-ConflictResolution	Required	Specifies a conflict resolution to handle conflicts occurred during migration. The valid values are "Skip", "SkipAndSkip", "SkipAndOverwrite", "SkipAndAppend", "MergeAndSkip", "MergeAndOverwrite" and "MergeAndAppend". For detailed information on each of these values, refer to the note below this table.
-JobReportType	Required	Specifies the job report level. The valid values are "Simplified" and "Detailed".

Parameter	Type	Comment
-TestRun	Optional	Specifies whether to simulate the execution of the selected plan.
-WaitForJobComplete	Optional	Specifies whether to wait for the completion of the job. If so, the job process will be displayed and another job can be run until this job has completed; if not, the job will run in the background and you can run another job without waiting for the completion of this job.

***Note:** The following information related to the values in the **-ConflictResolution** parameter.

- **Skip** – Skips the conflicted containers in a migration job when conflicts occur.
- **SkipAndSkip** – Skips the conflicted containers and all lower-level objects in a migration job when conflicts occur.
- **SkipAndOverwrite** – Skips the conflicted containers and overwrites all lower-level objects in a migration job when conflicts occur.
- **SkipAndAppend** – Skips the conflicted containers and appends all lower-level objects in a migration job when conflicts occur.
- **MergeAndSkip** – Merges the conflicted containers and skips all lower-level objects in a migration job when conflicts occur.
- **MergeAndOverwrite** – Merges the conflicted containers and overwrites all lower-level objects in a migration job when conflicts occur.
- **MergeAndAppend** – Merges the conflicted containers and appends all lower-level objects in a migration job when conflicts occur.

Update a SharePoint 2010 to 2013 Online Migration Plan

To update a SharePoint 2010 to 2013 online migration plan, run the command below:

Update-DASP10To13MigrationOnlinePlan [-Plan] <SP10To13MigrationOnlinePlan> [<CommonParameters>]

The example of the command is:

C:\PS>$plan = Get-DASP10To13MigrationOnlinePlan -Name 'plan1'

C:\PS>$plan.Name = 'plan2'

C:\PS>$plan.Description = 'New Description'

C:\PS>$plan.SourceFarmName = 'Farm(SP2010_DB:SHAREPOINT_CONFIG)'

C:\PS>$plan.DestinationFarmName = 'Farm(SP2013_DB:SHAREPOINT_CONFIG)'

C:\PS>$plan.SourceAgentGroup = 'AgentGroup10'

C:\PS>$plan.DestinationAgentGroup = 'AgentGroup13'

C:\PS>$plan.ProfileName = 'Default Profile'

C:\PS>$list10 = New-Object DocAve.API.Objects.Sharepointlist('Farm(SP2010_DB:SHAREPOINT_CONFIG)', 'http://SharePoint2010:10001/sites/site1','sites/site1', 'list1')

C:\PS>$plan.SourceTree.IncludeSPList($list10, $true, $false)

C:\PS>$list13 = New-Object DocAve.API.Objects.Sharepointlist('Farm(SP2013_DB:SHAREPOINT_CONFIG)', 'http://SharePoint2013:10001/sites/site1','sites/site1', 'list1')

C:\PS>$plan.DestinationTree.SelectSPObject($list13)

C:\PS>$plan.Action = 'Merge'

C:\PS>$plan.AssociatedPlanGroup.Add('plan group1')

C:\PS>$plan.UseSpecifiedMigrationDatabaseToStoreTheJobData = $true

C:\PS>Update-DASP10To13MigrationOnlinePlan -Plan $plan

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Plan	Required	Specifies a plan name to get an existing SharePoint 2010 to 2013 online migration plan for editing.

Upload a SharePoint 2010 to 2013 Online Migration Profile

To upload a SharePoint 2010 to 2013 online migration profile, run the command below:

Upload-DASP10To13MigrationOnlineProfile [-Path] <String> [-Overwrite] [<CommonParameters>]

The example of the command is:

C:\PS>Upload-DASP10To13MigrationOnlineProfile -Path 'C:\SP10To13MigrationOnlineProfile1.xml' -Overwrite

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Path	Required	Specifies the path of the XML file.
-Overwrite	Optional	Allows the cmdlet to overwrite an existing profile.

Download a SharePoint 2010 to 2013 Online Migration Profile

To download a SharePoint online migration plan, run the command below:

Download-DASP10To13MigrationOnlineProfile [-Name] <String> [-Path] <String> [-Overwrite] [<CommonParameters>]

The example of the command is:

C:\PS>Download-DASP10To13MigrationOnlineProfile -Name 'SP10To13MigrationOnlineProfile1' -Path 'C:\SP10To13MigrationOnlineProfile1.xml' -Overwrite

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Name	Required	Specifies the profile name.
-Path	Required	Specifies the path to store the downloaded XML file.
-Overwrite	Optional	Allows the cmdlet to overwrite an existing file.

File System Migration

Generate the Metadata File(s) for the Migrated Contents

To generate the metadata file(s) for the migrated content, run the command below:

Generate-DAFileSystemMigrationMetadataFile [-PlanName] <String> -Option <String> [<CommonParameters>]

The example of the command is:

C:\PS>$jobId = Generate-DAFileSystemMigrationMetadataFile -PlanName 'plan 1' -Option Full

To configure the command parameters, see the table below.

Parameter	Type	Comment
-PlanName	Required	Specifies the name of the plan used to generate the metadata configuration file.

Parameter	Type	Comment
-Option	Required	Specifies the generation type for the metadata file. Valid values are "Full" and "Incremental".

Get a Blank File System Online Migration Plan

To get a blank file system online migration plan, run the command below:

Get-DAFileSystemMigrationBlankOnlinePlan [<CommonParameters>]

The example of the command is:

C:\PS>$plan = Get-DAFileSystemMigrationBlankOnlinePlan

Get the Plan Settings of a File System Online Migration Plan

To get the plan settings of a file system online migration plan, run the command below:

Get-DAFileSystemMigrationOnlinePlan [[-Name] <String[]>] [<CommonParameters>]

The example of the command is:

C:\PS>Get-DAFileSystemMigrationOnlinePlan -Name 'plan 1'

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Name	Optional	Specifies the name(s) of the existing file system online migration plan(s) that you want to get information from. Wildcards and multiple values are allowed, which are separated by commas. If no values are entered here, the information of all the existing file system online migration plan(s) will be obtained.

Create a New File System Online Migration Plan

To create a new file system online migration plan, run the command below:

New-DAFileSystemMigrationOnlinePlan [-Plan] <FileSystemMigrationOnlinePlan> [<CommonParameters>]

The example of the command is:

C:\PS>$plan = Get-DAFileSystemMigrationBlankOnlinePlan

C:\PS>$plan.Name = 'plan 1'

C:\PS>$plan.ProfileName = 'default profile'

C:\PS>$plan.SourceAgentName = 'sp2010'

C:\PS>$folder = New-Object DocAve.API.Migration.FileSystemMigration.FileSystemFolder 'connection 2', 'folder1\folder2'

C:\PS>$plan.SourceTree.Include($folder)

C:\PS>$list = New-Object DocAve.API.Objects.SharePointList 'Farm(SQL:SHAREPOINT_CONFIG)', 'http://sp2010', '', 'shared documents'

C:\PS>$plan.DestinationTree.SelectSPObject($list)

C:\PS>New-DAFileSystemMigrationOnlinePlan $plan

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Plan	Required	Specifies a file system migration online plan object. About the structure and attributes of FileSystemMigrationOnlinePlan, please refer to the SDK documentation for more information.

Delete a File System Online Migration Plan

To delete a file system online migration plan, run the command below:

Remove-DAFileSystemMigrationOnlinePlan [-Name] <String> [-WhatIf] [-Confirm] [<CommonParameters>]

The example of the command is:

C:\PS>Remove-DAFileSystemMigrationOnlinePlan -Name 'plan1'

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Name	Required	Specifies the name of a file system online migration plan that you want to delete.
-WhatIf	Optional	Describes what would happen if you executed the command without actually executing the command.
-Confirm	Optional	Prompts you for confirmation before executing the command.

Run a File System Online Migration Job

To perform a file system online migration job, run the command below:

Run-DAFileSystemMigrationOnlineJob [-PlanName] <String> -Options <String> -ConflictResolution <FileSystemMigrationConflictResolution> [-TestRun] [-WaitForJobComplete] [<CommonParameters>]

The example of the command is:

C:\PS>Run-DAFileSystemMigrationOnlineJob -PlanName 'plan 1' -Options Full -ConflictResolution MergeAndAppend -WaitForJobComplete

To configure the command parameters, see the table below.

c	Type	Comment
-PlanName	Required	Specifies the name of an existing file system online migration plan that you want to run.
-Options	Required	Specifies to run a full or an incremental migration job of the selected file system online migration plan. Valid values are "Full" and "Incremental".
-ConflictResolution	Required	Specifies a conflict resolution to handle conflicts occurred during migration. The valid values are "Skip", "SkipAndSkip", "SkipAndOverwrite", "SkipAndAppend", "MergeAndSkip", "SkipAndNewVersion", "MergeAndOverwrite", "MergeAndAppend" and "MergeAndNewVersion". For detailed information on each of these values, refer to the note below this table.
-TestRun	Optional	Specifies whether to simulate the execution of the selected plan.
-WaitForJobComplete	Optional	Specifies whether to wait for the completion of the job. If so, the job process will be displayed and another job can be run until this job has completed; if not, the job will run in the background and you can run another job without waiting for the completion of this job.

***Note**: The following information related to the values in the **-ConflictResolution** parameter.

- **Skip** – Skips the conflicted containers in a migration job when conflicts occur.
- **SkipAndSkip** – Skips the conflicted containers and all lower-level objects in a migration job when conflicts occur.
- **SkipAndOverwrite** – Skips the conflicted containers and overwrites all lower-level objects in a migration job when conflicts occur.
- **SkipAndAppend** – Skips the conflicted containers and appends all lower-level objects in a migration job when conflicts occur.

- **MergeAndSkip** – Merges the conflicted containers and skips all lower-level objects in a migration job when conflicts occur.
- **MergeAndOverwrite** – Merges the conflicted containers and overwrites all lower-level objects in a migration job when conflicts occur.
- **MergeAndAppend** – Merges the conflicted containers and appends all lower-level objects in a migration job when conflicts occur.

Update a File System Online Migration Plan

To update a file system online migration plan, run the command below:

Update-DAFileSystemMigrationOnlinePlan [-Plan] <FileSystemMigrationOnlinePlan> [<CommonParameters>]

The example of the command is:

C:\PS>$plan = Get-DAFileSystemMigrationOnlinePlan -Name 'plan 1'

C:\PS>$plan.Description = 'updated from cmdlet'

C:\PS>$file = New-Object DocAve.API.Migration.FileSystemMigration.FileSystemFile 'connection 1', 'folder1\folder2\test.txt'

C:\PS>$plan.SourceTree.Include($file)

C:\PS>Update-DAFileSystemMigrationOnlinePlan $plan

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Plan	Required	Specifies a file system online migration plan for editing.

eRoom Migration

Get a Blank eRoom Online Migration Plan

To get a blank eRoom online migration plan, run the command below:

Get-DAERoomMigrationBlankOnlinePlan [<CommonParameters>]

The example of the command is:

C:\PS>$plan = Get-DAERoomMigrationBlankOnlinePlan

C:\PS>$plan

Get the Plan Settings of an eRoom Online Migration Plan

To get an eRoom online migration plan, run the command below:

Get-DAERoomMigrationOnlinePlan [[-Name] <String[]>] [<CommonParameters>]

The example of the command is:

C:\PS>Get-DAERoomMigrationOnlinePlan -Name 'plan01'

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Name	Optional	Specifies the name(s) of the existing eRoom online migration plan(s) that you want to get information from. Wildcards and multiple values are allowed, which are separated by commas. If no values are entered here, The information of all of the existing eRoom online migration plan(s) will be obtained.

Create a New eRoom Online Migration Plan

To create an eRoom online migration plan, run the command below:

New-DAERoomMigrationOnlinePlan [-Plan] <ERoomMigrationOnlinePlan> [<CommonParameters>]

The example of the command is:

C:\PS>$plan = Get-DAERoomMigrationBlankOnlinePlan

C:\PS>$plan.Name = 'plan1'

C:\PS>$plan.Description = 'this is an eRoom plan'

C:\PS>$plan.SourceAgentName = 'agent-eroom'

C:\PS>$item = New-Object
DocAve.API.Migration.ERoomMigration.ERoomItem('Community01','Facility01','Room01','0_9f7')

C:\PS>$room = New-Object
DocAve.API.Migration.ERoomMigration.ERoomRoom('Community01','Facility01','Room02')

C:\PS>$facility = New-Object
DocAve.API.Migration.ERoomMigration.ERoomFacility('Community01','Facility02')

C:\PS>$community = New-Object
DocAve.API.Migration.ERoomMigration.ERoomCommunity('Community02)

C:\PS>$plan.SourceTree.Include($item)

C:\PS>$plan.SourceTree.IncludeHomeFolder($room)

C:\PS>$plan.SourceTree.Include($facility)

C:\PS>$plan.SourceTree.Exclude($community)

C:\PS>$webapp10 = New-Object DocAve.API.Objects.SharePointWebApplication('Farm(SP2010_DB:SHAREPOINT_CONFIG)','http://SharePoint2010:10001/')

C:\PS>$plan.DestinationTree.SelectSPObject($webapp10)

C:\PS>$plan.ProfileName = 'Default Profile'

C:\PS>$plan.AssociatedPlanGroup.Add('plan01')

C:\PS>New-DAERoomMigrationOnlinePlan -Plan $plan

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Plan	Required	Specifies an eRoom online migration plan object. About the structure and attribute of ERoomMigrationOnlinePlan, please refer to the SDK documentation for more information.

Delete an eRoom Online Migration Plan

To delete an eRoom online migration plan, run the command below:

Remove-DAERoomMigrationOnlinePlan [-Name] <String> [-WhatIf] [-Confirm] [<CommonParameters>]

The example of the command is:

C:\PS>Remove-DAERoomMigrationOnlinePlan -Name 'plan1'

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Name	Required	Specifies the name of an eRoom online migration plan that you want to delete.
-WhatIf	Optional	Describes what would happen if you executed the command without actually executing the command.
-Confirm	Optional	Prompts you for confirmation before executing the command.

Run an eRoom Online Migration Job

To perform an eRoom online migration job, run the command below:

Run-DAERoomMigrationOnlineJob [-PlanName] <String> -Options <String> -ConflictResolution <ERoomMigrationConflictResolution> [-TestRun] [-WaitForJobComplete] [<CommonParameters>]

The example of the command is:

C:\PS>Run-DAERoomMigrationOnlineJob -PlanName 'plan1' -Options Full -ConflictResolution Skip

To configure the command parameters, see the table below.

Parameter	Type	Comment
-PlanName	Required	Specifies the name of an existing eRoom online migration plan that you want to run.
-Options	Required	Specifies to run a full or an incremental migration job of the selected migration plan. Valid values are "Full" and "Incremental".
-ConflictResolution	Optional	Specifies a conflict resolution to handle conflicts occurred during migration. The valid values are "Skip", "SkipAndSkip", "SkipAndOverwrite", "SkipAndAppend", "MergeAndSkip", "MergeAndOverwrite" and "MergeAndAppend". For detailed information on each of these values, refer to the note below this table.
-TestRun	Optional	Specifies whether to simulate the execution of the selected plan.
-WaitForJobComplete	Optional	Specifies whether to wait for the completion of the job. If so, the job process will be displayed and another job can be run until this job has completed; if not, the job will run in the background and you can run another job without waiting for the completion of this job.

***Note**: The following information related to the values in the **-ConflictResolution** parameter.

- **Skip** – Skips the conflicted containers in a migration job when conflicts occur.
- **SkipAndSkip** – Skips the conflicted containers and all lower-level objects in a migration job when conflicts occur.
- **SkipAndOverwrite** – Skips the conflicted containers and overwrites all lower-level objects in a migration job when conflicts occur.
- **SkipAndAppend** – Skips the conflicted containers and appends all lower-level objects in a migration job when conflicts occur.
- **MergeAndSkip** – Merges the conflicted containers and skips all lower-level objects in a migration job when conflicts occur.
- **MergeAndOverwrite** – Merges the conflicted containers and overwrites all lower-level objects in a migration job when conflicts occur.
- **MergeAndAppend** – Merges the conflicted containers and appends all lower-level objects in a migration job when conflicts occur.

Update an eRoom Online Migration Plan

To update an eRoom online migration plan, run the command below:

Update-DAERoomMigrationOnlinePlan [-Plan] <ERoomMigrationOnlinePlan> [<CommonParameters>]

The example of the command is:

C:\PS>$plan = Get-DAERoomMigrationOnlinePlan -Name 'plan1'

C:\PS>$plan.Name = 'plan2'

C:\PS>$plan.Description = 'this is an eRoom plan'

C:\PS>$plan.SourceAgentName = 'agent-eroom'

C:\PS>$item = New-Object DocAve.API.Migration.ERoomMigration.ERoomItem('Community01','Facility01','Room01','0_9f7')

C:\PS>$room = New-Object DocAve.API.Migration.ERoomMigration.ERoomRoom('Community01','Facility01','Room02')

C:\PS>$facility = New-Object DocAve.API.Migration.ERoomMigration.ERoomFacility('Community01','Facility02')

C:\PS>$community = New-Object DocAve.API.Migration.ERoomMigration.ERoomCommunity('Community02)

C:\PS>$plan.SourceTree.Include($item)

C:\PS>$plan.SourceTree.IncludeHomeFolder($room)

C:\PS>$plan.SourceTree.Include($facility)

C:\PS>$plan.SourceTree.Exclude($community)

C:\PS>$webapp10 = New-Object DocAve.API.Objects.SharePointWebApplication('Farm(SP2010_DB:SHAREPOINT_CONFIG)','http://SharePoint2010:10001/')

C:\PS>$plan.DestinationTree.SelectSPObject($webapp10)

C:\PS>$plan.ProfileName = 'Default Profile'

C:\PS>$plan.AssociatedPlanGroup.Add('plan01')

C:\PS>Update-DAERoomMigrationOnlinePlan -Plan $plan

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Plan	Required	Specifies an existing eRoom online migration plan for editing.

Lotus Notes Migration

Get a Blank Lotus Notes Online Migration Plan

To get a blank Lotus Notes online migration plan, run the command below:

Get-DALotusNotesMigrationBlankOnlinePlan [<CommonParameters>]

The example of the command is:

C:\PS>$plan = Get-DALotusNotesMigrationBlankOnlinePlan

C:\PS>$plan

Get the Plan Settings of a Lotus Notes Online Migration Plan

To get the plan settings of a Lotus Notes online migration plan, run the command below:

Get-DALotusNotesMigrationOnlinePlan [[-Name] <String[]>] [<CommonParameters>]

The example of the command is:

C:\PS>Get-DALotusNotesMigrationOnlinePlan -Name 'plan1'

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Name	Optional	Specifies the name(s) of the existing Lotus Notes online migration plan(s) that you want to get information from. Wildcards and multiple values are supported and the multiple values are separated by commas. If no values are entered here, the information of all of the existing Lotus Notes online migration plan(s) will be obtained.

Creates a New Lotus Notes Online Migration Plan

To create a new Lotus Notes online migration plan, run the command below:

New-DALotusNotesMigrationOnlinePlan [-Plan] <LotusNotesMigrationOnlinePlan> [<CommonParameters>

The example of the command is:

C:\PS>$plan = Get-DALotusNotesMigrationBlankOnlinePlan

C:\PS>$plan.Name = 'plan1'

C:\PS>$plan.Description = 'Description'

C:\PS>$plan.ProfileName = 'Default Profile'

C:\PS>$site1 = New-Object DocAve.API.Objects.SharePointSite('Farm(SP2010_DB:SHAREPOINT_CONFIG)','http://SharePoint2010:10001/sites/site1')

C:\PS>$plan.DestinationTree.SelectSPObject($site1)

C:\PS>$plan.SetSourceInformation('Connection-LotusNotes', 'Local')

C:\PS>$NotesObject = New-Object DocAve.API.Migration.LotusNotesMigration.LotusNotesDatabase("dommon.nsf")

C:\PS>$plan.SourceTree.Include($NotesObject)

C:\PS>New-DALotusNotesMigrationOnlinePlan $plan

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Plan	Required	Specifies a Lotus Notes online Migration plan object. About the structure and attribute of LotusNotesMigrationOnlinePlan, please refer to the SDK documentation for more information.

Delete a Lotus Notes Online Migration Plan

To delete a Lotus Notes online migration plan, run the command below:

Remove-DALotusNotesMigrationOnlinePlan [-Name] <String> [-WhatIf] [-Confirm] [<CommonParameters>]

The example of the command is:

C:\PS>Remove-DALotusNotesMigrationOnlinePlan -Name 'plan1'

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Name	Required	Specifies the name of a Lotus Notes online migration plan that you want to delete.
-WhatIf	Optional	Describes what would happen if you executed the command without actually executing the command.
-Confirm	Optional	Prompts you for confirmation before executing the command.

Run a Lotus Notes Online Migration Job

To perform a Lotus Notes online migration plan, run the command below:

Run-DALotusNotesMigrationOnlineJob [-PlanName] <String> -Options <String> -ConflictResolution <LotusNotesMigrationConflictResolution> [-TestRun] [-WaitForJobComplete] [<CommonParameters>]

The example of the command is:

C:\PS>Run-DALotusNotesMigrationOnlineJob -PlanName 'plan1' -Options Full -ConflictResolution Skip -WaitForJobComplete

To configure the command parameters, see the table below.

Parameter	Type	Comment
-PlanName	Required	Specifies the name of an existing Lotus Notes online migration plan that you want to run.
-Options	Required	Specifies to run a full or an incremental migration job of the selected migration plan. Valid values are "Full" and "Incremental".
-ConflictResolution	Required	Specifies a conflict resolution to handle conflicts occurred during migration. The valid values are "Skip", "SkipAndSkip", "SkipAndOverwrite", "SkipAndAppend", "MergeAndSkip", "MergeAndOverwrite" and "MergeAndAppend". For detailed information on each of these values, refer to the note below this table.
-TestRun	Optional	Specifies whether to simulate the execution of the selected plan.
-WaitForJobComplete	Optional	Specifies whether to wait for the completion of the job. If so, the job process will be displayed and another job can be run until this job has completed; if not, the job will run in the background and you can run another job without waiting for the completion of this job.

***Note**: The following information related to the values in the **-ConflictResolution** parameter.

- **Skip** – Skips the conflicted containers in a migration job when conflicts occur.
- **SkipAndSkip** – Skips the conflicted containers and all lower-level objects in a migration job when conflicts occur.
- **SkipAndOverwrite** – Skips the conflicted containers and overwrites all lower-level objects in a migration job when conflicts occur.
- **SkipAndAppend** – Skips the conflicted containers and appends all lower-level objects in a migration job when conflicts occur.
- **MergeAndSkip** – Merges the conflicted containers and skips all lower-level objects in a migration job when conflicts occur.

- **MergeAndOverwrite** – Merges the conflicted containers and overwrites all lower-level objects in a migration job when conflicts occur.
- **MergeAndAppend** – Merges the conflicted containers and appends all lower-level objects in a migration job when conflicts occur.

Update a Lotus Notes Online Migration Plan

To update a Lotus Notes online migration plan, run the command below:

Update-DALotusNotesMigrationOnlinePlan [-Plan] <LotusNotesMigrationOnlinePlan> [<CommonParameters>]

The example of the command is:

C:\PS>$plan = Get-DALotusNotesMigrationOnlinePlan -Name 'plan1'

C:\PS>$plan.Name = 'plan2'

C:\PS>$plan.Description = 'Description'

C:\PS>$plan.ProfileName = 'Default Profile'

C:\PS>$site1 = New-Object DocAve.API.Objects.SharePointSite('Farm(SP2010_DB:SHAREPOINT_CONFIG)','http://SharePoint2010:10001/sites/site1')

C:\PS>$plan.DestinationTree.SelectSPObject($site1)

C:\PS>$plan.SetSourceInformation('Connection-LotusNotes', 'Local')

C:\PS>$NotesObject = New-Object DocAve.API.Migration.LotusNotesMigration.LotusNotesDatabase("dommon.nsf")

C:\PS>$plan.SourceTree.Include($NotesObject)

C:\PS>Update-DALotusNotesMigrationOnlinePlan -Plan $plan

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Plan	Required	Specifies the plan name to get an existing Lotus Notes online migration plan for editing.

Data Protection

Granular Backup and Restore

Obtain a Blank Granular Backup Blank Plan

To get a blank Granular Backup blank plan, run the command below:

Get-DAGranularBackupBlankPlan [<CommonParameters>]

The example of the command is:

C:\PS>Get-DAGranularBackupBlankPlan

Obtain a Granular Backup Plan's Configuration Details

To get the configurations of an existing Granular Backup plan, run the command below:

Get-DAGranularBackupPlan [[-Name] <String[]>] [<CommonParameters>]

The example of the command is:

C:\PS>Get-DAGranularBackupPlan -Name 'plan1'

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Name	Optional	Specifies one or more plans' names. You can type multiple plan names (separated by commas). Wildcards are supported.

Create a Granular Backup Plan

To create a Granular Backup plan, run the command below:

New-DAGranularBackupPlan [-Plan] <GranularBackupPlan> [<CommonParameters>]

The example of the command is:

C:\PS>$plan = Get-DAGranularBackupBlankPlan
C:\PS>$plan.Name = 'plan1'
C:\PS>$plan.Description = 'description'
C:\PS>$plan.StoragePolicyName = 'storage policy1'
C:\PS>$plan.AgentGroupName =
"DEFAULT_AGENT_GROUP_FOR_Farm(SP2010_DB:SHAREPOINT_CONFIG)"
C:\PS>
C:\PS>$site1 = New-Object DocAve.API.Objects.SharePointSite
('Farm(SP2010_DB:SHAREPOINT_CONFIG)','http://SharePoint2010:10001/sites/site1')
C:\PS>$plan.Tree.IncludeSPObject($site1)

C:\PS>
C:\PS>New-DAGranularBackupPlan -Plan $plan

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Plan	Required	Specifies a Granular Backup plan object.

Run a Granular Backup Plan

To run an existing Granular Backup plan, run the command below:

Run-DAGranularBackupJob [-PlanName] <String> -BackupJobType <RunGranularBackupPlanType> -JobReportType <String> [-TestRun] [-WaitForJobComplete] [<CommonParameters>]

The example of the command is：

C:\PS>Run-DAGranularBackupJob -PlanName 'plan1' -BackupJobType full -JobReportType Detailed -WaitForJobComplete

To configure the command parameters, see the table below.

Parameter	Type	Comment
-PlanName	Required	Specifies the name of an existing Granular Backup plan that you want to run.
-BackupJobType	Required	Specifies the type of backup. Valid values are “Full”, “Incremental”, and “Differential”.
-JobReportType	-JobReportType	Specifies the job report level. Valid values are “Simplified” and “Detailed”.
-TestRun	Optional	Specifies whether to simulate the execution of the selected plan.
-WaitForJobComplete	Optional	Specifies whether to wait for the completion of the job. If so, the job process will be displayed and another job cannot be run until this job has completed; if not, the job will run in the background and you can run another job without waiting for the completion of this job.

Delete Granular Backup Plan

To delete a Granular Backup plan, run the command below:

Remove-DAGranularBackupPlan [-Name] <String> [-WhatIf] [-Confirm] [<CommonParameters>]

The example of the command is:

C:\PS>Remove-DAGranularBackupPlan -Name 'plan1'

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Name	Required	Specifies the name of plan to be removed.
-WhatIf	Optional	Describes what would happen if you executed the command.
-Confirm	Optional	Prompts you for confirmation before executing the command.

Update the Configurations of a Granular Backup Plan

To update the configurations of an existing Granular Backup plan, run the command below:

Update-DAGranularBackupPlan [-Plan] <GranularBackupPlan> [<CommonParameters>]

The example of the command is：

C:\PS>$plan = Get-DAGranularBackupPlan -Name 'plan1'
C:\PS>$plan.Name = 'plan2'
C:\PS>Update-DAGranularBackupPlan -Plan $plan

To configure the command parameters, see the table below.

Parameter	Type	Comment
-Plan	Required	Specifies a Granular Backup plan object.

Platform Backup and Restore

Run a Platform Backup Plan

To run an existing Platform Backup plan, run the command below:

Run-DAPlatformBackupJob [-PlanName] <String> -BackupType <PRBackupType> [-CopyDataFromSnapshotToMediaServer] [-DeferCopyingDataFromPersistentSnapshot] [-DeferGeneratingInstaMountMapping] -RestoreGranularityLevel <PlatformRestoreGranularityLevel> [-DeferGenerationGranularRestoreIndex] [-WaitForJobComplete] [<CommonParameters>]

The example of the command is：

C:\PS>Run-DAPlatformBackupJob -PlanName 'plan1' -BackupType Full -RestoreGranularityLevel ItemLevel

To configure the command parameters, see the table below.

Parameter	Type	Comment
-PlanName	Required	Specifies the name of an existing Platform Backup plan that you want to run.
-BackupType	Required	Specifies the type of backup. Valid values are “Full”, “Incremental”, and “Differential”.

Parameter	Type	Comment
-CopyDataFromSnapshotToMediaServer	Optional	Specifies whether to copy data from snapshot to media server.
-DeferCopyingDataFromPersistentSnapshot	Optional	Specifies whether to defer copy data from persistent snapshot.
-DeferGeneratingInstaMountMapping	Optional	Specifies whether to defer generate InstaMount mapping.
-RestoreGranularityLevel	Required	Specifies the restore granularity level. Valid values are “SiteCollectionLevel”, “SiteLevel”, “FolderLevel”, “ItemLevel”, “ItemVersionLevel”, and “None”.
-DeferGenerationGranularRestoreIndex	Optional	Specifies whether to defer generate granular restore index.
-WaitForJobComplete	Optional	Specifies whether to wait for the completion of the job. If so, the job process will be displayed and another job cannot be run until this job has completed; if not, the job will run in the background and you can run another job without waiting for the completion of this job.

Notices and Copyright Information

Notice

The materials contained in this publication are owned or provided by AvePoint, Inc. and are the property of AvePoint or its licensors, and are protected by copyright, trademark and other intellectual property laws. No trademark or copyright notice in this publication may be removed or altered in any way.

Copyright

Copyright © 2012-2013 AvePoint, Inc. All rights reserved. All materials contained in this publication are protected by United States copyright law and no part of this publication may be reproduced, modified, displayed, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior written consent of AvePoint, 3 Second Street, Jersey City, NJ 07311, USA or, in the case of materials in this publication owned by third parties, without such third party's consent.

Trademarks

AvePoint®, DocAve®, the AvePoint logo, and the AvePoint Pyramid logo are registered trademarks of AvePoint, Inc. with the United States Patent and Trademark Office. These registered trademarks, along with all other trademarks of AvePoint used in this publication are the exclusive property of AvePoint and may not be used without prior written consent.

Microsoft, MS-DOS, Internet Explorer, Microsoft Office SharePoint Servers 2007/2010/2013, SharePoint Portal Server 2003, Windows SharePoint Services, Windows SQL server, and Windows are either registered trademarks or trademarks of Microsoft Corporation.

Adobe Acrobat and Acrobat Reader are trademarks of Adobe Systems, Inc.

All other trademarks contained in this publication are the property of their respective owners and may not be used such party's consent.

Changes

The material in this publication is for information purposes only and is subject to change without notice. While reasonable efforts have been made in the preparation of this publication to ensure its accuracy, AvePoint makes no representation or warranty, expressed or implied, as to its completeness, accuracy, or suitability, and assumes no liability resulting from errors or omissions in this publication or from the use of the information contained herein. AvePoint reserves the right to make changes in the Graphical User Interface of the AvePoint software without reservation and without notification to its users.

AvePoint, Inc.
3 Second Street
Jersey City, NJ 07311
USA